A Survey of Large Language Models for Text Classification: What, Why, When, Where, and How

Zhiqiang Wang[1], Yanbin Lin[1], Jiajun Shen[1], and Xingquan Zhu[1]

[1]Florida Atlantic University

May 01, 2025

Large Language Models for Text Classification: What, Why, When, Where, and How

Zhiqiang Wang, Yanbin Lin, Jiajun Shen, Xingquan Zhu[+]

Florida Atlantic University
[+]Corresponding Author

***Abstract*—In an age where unstructured text data is growing rapidly, effective methods for text classification(TC) have become critical. Large Language Models (LLMs), such as the revolutionary GPT-4, have taken the lead in tackling this challenge, showing remarkable abilities in handling complex language tasks. This paper presents the first thorough survey focused on LLMs for TC, a key application for managing and understanding the vast amounts of digital text we encounter today. We examine how well LLMs meet the needs of TC, explore their strengths and weaknesses, and discuss practical situations where LLMs perform best. We systematically address important questions about the effectiveness of LLMs in TC, explaining 'What' these models are in this context, 'Why' they are well-suited for these tasks, 'When' they should be used, 'Where' they have the most impact, and 'How' to use them effectively. By looking at both the benefits and challenges of using LLMs for TC, this survey aims to provide a clear guide for researchers and professionals, encouraging better use and ongoing improvements in text analysis.**

***Index Terms*—Large language models, text classification, data annotation, RAG, fine-tuning, GPT-4, GPT-4o, Llama**

I. Introduction

In recent years, the rapid development of Large Language Models (LLMs) has demonstrated remarkable capabilities in assisting humans with a variety of text-related tasks, such as code generation, content rephrasing, summarization, and text annotation [1]–[3]. A noteworthy milestone in this domain is the launch of ChatGPT, *i.e.* Generative Pre-trained Transformer. In March 2023, the launch of the ChatGPT-4 [4], an advanced AI chatbot built on LLMs, has captured significant attention from public, due to its resemblance to human chatting and answering, and the ability to outperform humans in several areas [5]. Following the GPT, other LLMs such as LLaMA [6] or PaLM [7], have quickly emerged, making large language models a well sought-after approaches to solve many learning tasks, especially for text related applications.

Parallel to the rise of AI advancements, the digital age has seen an exponential increase in data production and consumption globally. Approximately 80% of all generated data is unstructured [], predominantly in textual form, growing at a rate much faster than structured data repositories. Text classification (TC) technologies are crucial as they systematically categorize and manage this vast influx of text in numerous fields, including media, academia, and customer service [8].

The motivation behind this research is rooted in the observation that LLMs have shown a potential solution for TC challenges, frequently outperforming locally trained models. This realization prompts several critical inquiries:

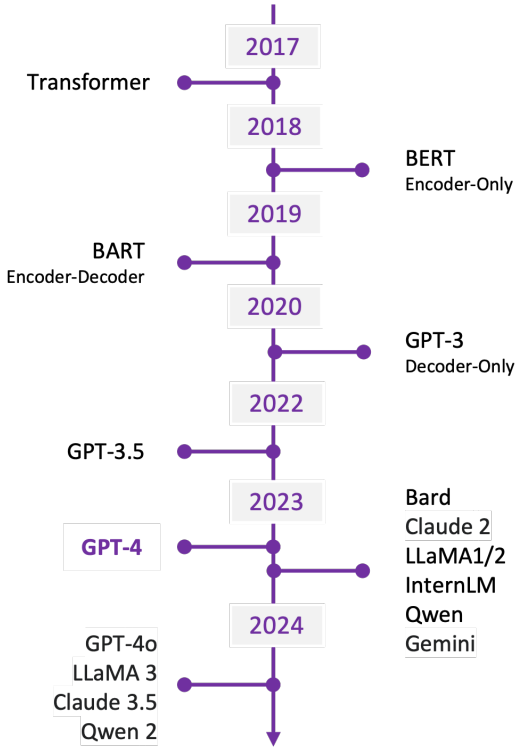

Fig. 1. The figure presents key milestones in the development of modern LLMs as of July 2024, highlighting significant and popular models such as GPT-4, Claude 2, LLaMA, and Qwen 2.

- **What** is LLMs for TC?
- **Why** are these tasks well-suited to LLMs, and what are the primary challenges in TC that LLMs can effectively address?
- **When** should LLMs be preferred over other models for TC tasks?
- **Where** can LLMs be most beneficially employed in TC scenarios?
- **How** can one leverage LLMs for TC tasks, including direct usage or more sophisticated fine-tuning methods?

II. WHAT

A. What are LLMs

LLMs refer to large-sized pre-trained language models [31] that are sophisticated artificial neural networks (NNs) derived from the Transformer architecture [32], an approach known for its ability to handle sequential data with high efficiency and effectiveness. While there are several architectural configurations, such as decoder-only and encoder-only models, as of July 2024, the most dominant and successful examples are predominantly decoder-only models [31]. This trend was significantly influenced by the success of the OpenAI GPT family, which demonstrated robust capabilities in generating coherent and contextually relevant text dynamically. Fig. 1 shows the timeline of key advancements in LLMs, starting with the introduction of the Transformer in 2017. This was followed by various iterations and improvements leading up to the landmark model of GPT-4 in 2023. The figure continues to showcase subsequent prominent LLMs, emphasizing the rapid evolution in the field.

The journey of an LLM begins with pre-training [33], a phase where the model is exposed to vast amounts of raw text corpus, including articles, documents, and various other forms of textual data. This extensive pre-training enables the model to learn the complexities and nuances of language, capturing semantic patterns and contextual relationships between words and phrases.

Following this, the model undergoes fine-tuning on supervised datasets configured to specific tasks or domains. This fine-tuning process helps the LLM generate more accurate and contextually appropriate responses for targeted tasks, such as language translation, question-answering, or text generation. The specialization achieved through fine-tuning allows the model to adapt to various applications with increased precision and reliability.

During the inference stage, which occurs when interacting with an LLM, a user inputs a prompt or query. The model then generates responses based on its pre-training and fine-tuning. This response generation process capitalizes on the model's learned knowledge and capabilities, thereby providing users with relevant information.

B. What is text classification

TC is a fundamental task in natural language processing (NLP) that involves categorizing text into predefined classes or categories. The scope of TC spans a wide range of applications, including spam detection, sentiment analysis, topic labeling, and intent recognition. There are various types of TC, such as binary classification (e.g., spam vs. not spam), multi-class classification (e.g., categorizing news articles into topics like sports, politics, and technology), and multi-label classification (e.g., a single text may belong to multiple categories simultaneously) [34].

Traditionally, machine learning techniques like Naive Bayes (NB) [35], Support Vector Machines (SVM) [36], and Decision Trees have been employed for TC. These methods typically require manual feature extraction and engineering, such as transforming text data into numerical features. With the advent of deep learning, more sophisticated models like Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs) [37], and Transformer-based architectures [38] have significantly enhanced TC performance. These deep learning models can automatically learn hierarchical feature representations from raw text data, leading to more accurate and robust classification outcomes.

The TC process generally follows a structured workflow, beginning with data collection to gather a labeled dataset that reflects the classification task at hand. Next, data preprocessing is undertaken, which includes standardizing the text by removing stopwords and punctuation and converting text to lowercase. Tokenization, stemming, and lemmatization may also be applied to reduce words to their base forms. For feature extraction, traditional methods utilize bag-of-words, TF-IDF, or word embeddings, while deep learning models leverage advanced techniques like Word2Vec [39], GloVe, or contextualized embeddings from models such as BERT. After preprocessing, the classification model is trained on the processed data to distinguish between various categories. After training, the model is evaluated with metrics such as accuracy and F1-score to assess its performance.

C. Text classification Challenges

TC is a critical task that has been extensively explored in both academia and industry, leveraging a variety of methodologies ranging from traditional machine learning techniques to more intricate ensemble learning [40] and deep learning strategies [41]. However, despite these advancements, several challenges persist at various stages of the TC pipeline.

Key challenges originate from the initial stages of data collection and preparation. Often, specific domains, like clinical text [42], suffer from data scarcity where only a limited number of samples are available, complicating the task of building robust classifiers. Data labeling further intensifies resource demands, typically requiring significant human effort to annotate data accurately [43]. Additionally, data cleaning and feature extraction are imperative yet complex steps involving sophisticated techniques to transform raw data into suitable formats, which are crucial for enhancing model performance [44] but require additional skills for human beings.

Moreover, certain scenarios such as short TC introduce unique difficulties, including data sparsity, immediacy, non-standard language usage, noise, and imbalanced class distribution [45], making conventional methods less effective. Furthermore, multi-language content, especially when mixed within a single document, poses a significant hurdle for traditional classifiers [46]–[48]. Building effective models for such scenarios typically demands expertise and considerable effort to manage these intricacies.

Table I provides a comprehensive summary of the challenges in TC and evaluates the effectiveness of both DL models and LLMs in addressing these challenges. The table also highlights relevant LLM-related research and applications that

TABLE I
A SUMMARY OF CHALLENGES IN TEXT CLASSIFICATION AND LLMS' CAPACITY TO ADDRESS SUCH CHALLENGES. THE NUMBER OF PLUS SIGNS INDICATES THE FITNESS OF LLMS IN ADDRESSING CHALLENGES: FROM LOW (+), MEDIUM (++), TO HIGH (+++).

Challenges	DL	LLMs	TC in LLMs' Related Researches
Limited(labeled) Data	+	+++	Enhanced model pre-training allows LLMs to perform well in few-shot learning scenarios. [9]–[15]
Imbalanced Data	+	++	LLMs exhibit improved robustness in handling class imbalance through sophisticated representation learning. [16], [17]
Data Pre-process	++	+++	LLMs inherently capture and process complex patterns in unstructured text, simplifying preprocessing requirements. [17]–[20]
Multi-language	+	+++	LLMs' extensive training on diverse multilingual datasets offers superior cross-linguistic understanding. [21], [21]–[25]
Explainability	-	+++	LLMs are able to provide reasoning behind the categorized class. [26]–[30]

pertain to these specific challenges. The number of plus signs indicates the degree to which LLMs are adept at addressing each challenge, ranging from low (+), medium (++), to high (+++).

D. LLMs for text classification

LLMs offer revolutionary capabilities that address many traditional challenges in TC. These models, distinguished by their vast knowledge bases acquired during pre-training, can act as zero-shot classifiers [49], [50], eliminating the need for substantial data gathering and labeling efforts. By directly applying LLMs to raw text data, organizations can bypass the labor-intensive data labeling process, substantially reducing the time and cost involved [51].

Beyond simplifying data preparation, LLMs inherently streamline the feature extraction process. They are designed to extract and utilize contextual features from text without the explicit need for manual feature engineering steps [52]. This capability is particularly beneficial in scenarios like short TC where textual context is minimal and prone to noise as LLMs can interpret such texts effectively due to their contextual understanding and generative abilities [32], [53].

Furthermore, LLMs democratize TC, making it accessible to non-specialists. This is of immense value in regions lacking a deep pool of machine learning expertise, as it allows for more widespread use and integration of sophisticated TC solutions across various applications. The ability of LLMs to understand and classify multilingual content also stands out as a crucial advantage [54], [55], enabling more inclusive and globally applicable TC systems.

III. WHY

Understanding why LLMs are well-suited for TC tasks is essential for capitalizing on their capabilities and integrating them effectively into practical applications. This section delves into the intrinsic advantages of LLMs, highlighting their efficiency, flexibility, built-in capabilities, and ability to address specific challenges in TC.

Table II, adapted from [87], provides a high-level overview of popular language models ranging from BERT to the most recent advancements, such as GPT-4 and LLaMA. A significant trend evident from the table is the exponential increase in training data size and diversification of training sources over time. For instance, LLaMA 3 was pre-trained on a corpus of approximately 15 trillion multilingual tokens, a considerable leap from the 1.8 trillion tokens used for LLaMA 2 [74], developed merely a year earlier. Additionally, while the predominant size of LLMs now ranges from 13 billion to 70 billion parameters, cutting-edge models have achieved unprecedented scales, such as LLaMA 3 with 405 billion parameters and Claude 3 with a remarkable 2 trillion parameters in 2024 [74], [83], [88], compared to BERT's 110 million parameters in 2018 [52]. This unprecedented growth in both the size and quality of pre-training data and model parameters enables LLMs to encapsulate a vast spectrum of linguistic patterns and contextual knowledge. As a result, these advancements significantly enhance their capability to tackle complex NLP tasks, including TC, with greater accuracy and flexibility.

A. Robust Pre-Training and Extensive Knowledge Base

LLMs are subjected to extensive unsupervised pre-training on vast and diverse text corpora, which equips them with a broad understanding of natural language; for example, Llama 3 and GPT-3 are pre-trained on approximately 15 trillion [74], [88], and 300 billion tokens [53], respectively, sources ranging from Wikipedia, books, articles, and code, etc. This exhaustive preparation is not limited to high-resource languages but in-

TABLE II
HIGH-LEVEL OVERVIEW OF POPULAR LANGUAGE MODELS

Type	Model Name	#Parameters	Release	Base Models	Open Source	#Tokens	Training dataset
Encoder-Only	BERT [52]	110M, 340M	2018	-	✓	137B	BooksCorpus, English Wikipedia
	RoBERTa [56]	355M	2019	-	✓	2.2T	BooksCorpus, English Wikipedia, CC-NEWS, STORIES (a subset of Common Crawl), Reddit
	ALBERT [57]	12M, 18M, 60M, 235M	2019	-	✓	137B	BooksCorpus, English Wikipedia
	DeBERTa [58]	134M, 900M, 1.5B	2020	-	✓	-	BooksCorpus, English Wikipedia, STORIES, Reddit content
	XLNet [59]	110M, 340M	2019	-	✓	32.89B	BooksCorpus, English Wikipedia, Giga5, Common Crawl, ClueWeb 2012-B
Decoder-only	GPT-1 [60]	120M	2018	-	✓	1.3B	BooksCorpus
	GPT-2 [61]	1.5B	2019	-	✓	10B	Reddit outbound
Encoder-Decoder	T5 (Base) [62]	223M	2019	-	✓	156B	Common Crawl
	MT5 (Base) [63]	300M	2020	-	✓	-	New Common Crawl-based dataset in 101 languages (m Common Crawl)
	BART (Base) [64]	139M	2019	-	✓	-	Corrupting text
GPT Family	GPT-3 [53]	125M, 350M, 760M, 1.3B, 2.7B, 6.7B, 13B, 175B	2020		×	300B	Common Crawl (filtered), WebText2, Books1, Books2, Wikipedia
	CODEX [65]	12B	2021	GPT	✓	-	Public GitHub software repositories
	WebGPT [66]	760M, 13B, 175B	2021	GPT-3	×	-	ELI5
	GPT-4 [67]	1.76T	2023	-	×	13T	-
	GPT-4o [68]	-	2024	-	×	-	-
LLaMA Family	LLaMA1 [6]	7B, 13B, 33B, 65B	2023	-	✓	1T, 1.4T	Online sources
	LLaMA2 [69]	7B, 13B, 34B, 70B	2023	-	✓	2T	Online sources
	Alpaca [70]	7B	2023	LLaMA1	✓	-	GPT-3.5
	Vicuna-13B [71]	13B	2023	LLaMA1	✓	-	GPT-3.5
	Mistral-7B [72]	7.3B	2023		✓	-	-
	Code Llama [73]	34	2023	LLaMA2	✓	500B	Publicly available code
	LLaMA3 [74]	8B, 70B, 405B	2024	-	✓	15T	Online sources
PaLM Family	PaLM [7]	8B, 62B, 540B	2022	-	×	780B	Web documents, books, Wikipedia, conversations, GitHub code
	U-PaLM [75]	8B, 62B, 540B	2022	-	×	1.3B	Web documents, books, Wikipedia, conversations, GitHub code
	PaLM-2 [76]	340B	2023	-	✓	3.6T	Web documents, books, code, mathematics, conversational data
	Med-PaLM [77]	540B	2022	PaLM	×	780B	HealthSearchQA, MedicationQA, LiveQA
	Med-PaLM 2 [76]	-	2023	PaLM 2	×	-	MedQA, MedMCQA, HealthSearchQA, LiveQA, MedicationQA
Other LLMs	FLAN [78]	137B	2021	LaMDA-PT	✓	-	Web documents, code, dialog data, Wikipedia
	LaMDA [79]	137B	2022	-	×	168B	public dialog data and web documents
	Mixtral-8x7B [80]	46.7B	2023	-	✓	-	Instruction dataset
	Qwen [81]	1.8B, 7B, 14B, 72B	2023	-	✓	3T	Web documents, encyclopedia, books, codes
	Qwen 2 [82]	0.5B, 15B, 7B, 72B	2024	-	✓	7T	Web documents, encyclopedia, books, codes
	Claude 3 [83]	2T	2024	-	×	40T	-
	Gemini 1.5 [84]	-	2024	-	×	-	Web documents, books, and code, image data, audio data, video data
	DeepSeek-Coder [85]	1.3B, 6.7B, 33B	2024	-	✓	2T	GitHub's Markdown and StackExchange
	DocLLM [86]	1B,7B	2024	-	×	2T	IIT-CDIP Test Collection 1.0, DocBank

cludes materials from a wide array of domains and languages, thus fostering a deep and wide-ranging lexical and contextual knowledge base.

Such comprehensive pre-training enables LLMs to adeptly process various types of text, including those inputs that are unseen. Unlike a traditional database, the knowledge in LLMs is distributed across the NN parameters. Their general "knowledge base"—the NN weights—can be further improved to handle targeted tasks by supervised fine-tuning and reinforcement learning with human feedback (RLHF) [89]. In this context, a larger-sized LLM typically encapsulates more knowledge.

B. Flexibility Across Contexts and Languages

Due to their training in multilingual datasets, LLMs can effortlessly handle text in numerous languages and dialects. This multilingual capability allows for the application of a single model across different linguistic settings without developing separate language-specific models. For instance, ChatGPT-4 supports more than 80 languages, and some languages show only a slight gap compared to English in some evaluations [90], demonstrating robust multilingual abilities. Additionally, Llama 3 will be trained on datasets with high-quality non-English data to better support a diverse range of multilingual use cases [88].

Additionally, LLMs are highly flexible in adapting to various styles and registers of language, such as the informal and dynamic text often found in social media. This adaptability is crucial for maintaining high accuracy in environments where colloquial language and slang are prevalent. For example, models like Anthropic Claude [91] and ChatGPT not only handle everyday conversational text but also support specialized formatting and coding languages such as Markdown, LaTeX, HTML, and CSS. This demonstrates their ability to interpret and generate a wide range of text types, making them invaluable for both casual and technical applications.

C. Efficiency in Learning and Deployment

LLMs' ability to perform zero-shot and few-shot learning tasks is a pivotal advantage. They can readily apply their extensive pre-trained knowledge to new TC tasks without needing much task-specific data or extensive retraining. This capability is particularly valuable in scenarios where swift adaptability to rapidly evolving topics or limited data availability defines success.

The reduced need for additional training data, coupled with the capability to adjust to new tasks quickly, dramatically decreases both the time and resources required for model deployment, presenting a stark contrast to traditional models that often necessitate extensive and time-consuming training [92].

D. Inherent Support for Text Classification

LLMs incorporate sophisticated architectural features that are inherently advantageous for TC tasks. The core of this capability lies in the attention mechanisms that enable these models to process and prioritize different parts of the input text based on their relevance to the specific task. This allows the model to appreciate and utilize contextual relationships within the text, a critical aspect of deciphering meaning and intent in natural language.

Beyond mere classification, the architectural nuances of LLMs equip them to delve deeper into the underlying semantics of the text, thereby not just categorizing content but also unraveling the logic behind these categorizations. Such an understanding promotes greater transparency and explainability in model outputs, offering users answers and comprehensible insight into how those answers were derived. This ability is increasingly vital in applications demanding high trust and accountability, such as in legal or regulatory environments.

The superior pre-training, multilingual capabilities, operational efficiency, and inherent architectural advantages position LLMs uniquely effective for TC tasks. Continuing advances in model design and training methodologies will likely enhance their performance and applicability further, reinforcing the role of LLMs at the forefront of text-based AI applications. This ongoing evolution promises to expand their utility further, solidifying their status as an essential tool in modern AI arsenals.

IV. When

Determining when to use LLMs over other models for TC can significantly impact performance, resource utilization, and the overall success of a project. This section delves into specific scenarios where opting for LLMs presents clear advantages, guiding practitioners to make informed decisions based on unique requirements and constraints.

A. Under Resource-Limited Scenarios

Having been trained on extensive and diverse datasets, LLMs, such as GPT-4, provide robust performance across various tasks without needing a lot of domain-specific data or extensive training. This attribute makes them particularly valuable when collecting large, annotated datasets is impractical or expensive.

LLMs reduce the cost of extensive data preprocessing and feature engineering, which are important in traditional models but can be resource-consuming. This simplification accelerates the development cycle and lowers the entry barrier for organizations without deep technical resources.

Since LLMs come pre-trained and understand a broad array of tasks and languages, they considerably reduce the dependency on specialized machine learning expertise. Organizations can utilize LLMs effectively with a more generalist skill set.

Without requiring specific training on the data, LLMs can achieve over 95% accuracy in labeling unlabeled datasets, such as IMDb and UMLS, as demonstrated in [15]. This significantly reduces the need for manual labeling, which would otherwise demand substantial human resources.

TABLE III
TYPES OF TEXT CLASSIFICATION TASKS, APPLICATIONS, AND STRENGTHS AND WEAKNESSES OF USING LLMS FOR SUCH TC TASKS.

Industry	TC tasks
Healthcare	Medical diagnosis, health analysis, research classification [15], [18], [19], [27], [93], [94]
Environmental Science	Forest cover type classification, agricultural text classification [25], [95]
Finance	Financial sentiment/emotion analysis, document classification, advisor [9], [12], [13], [96]–[99]
Education	Educational Assessment Analysis, material categorization [14], [29], [93], [100], [101]
Technology & Cybersecurity	Cyber threat detection, vulnerability identification, emotion/sentiment classification [16], [17], [17], [22], [26], [102]–[105]
Biological Sciences	Biological Entity recognition, molecular classification, literature sorting [10], [28], [30], [106]
Legal	Legal document classification [17], [20], [107]–[109]
Media & Entertainment	Reviews, sentiment analysis [10], [15], [21], [97]
Others	Benchmark, general TC tasks [23], [49], [93], [99], [105], [110]

B. In Complex Linguistic Scenarios

TC tasks that involve multiple languages present a unique set of challenges. The intrinsic ability of LLMs to process and understand text across multiple languages makes them ideal for global applications, where data might span several linguistic frameworks.

Traditional models often require extensive fine-tuning to handle texts with intricate semantics or unusual syntactical features. LLMs, in contrast, leverage their broad training foundation to manage such complexities better without additional customization.

While multilingual LLMs(MLLMs) can understand multiple languages, they could have hallucination phenomena, inaccuracies, inconsistency, outdated knowledge across different languages, as well as moral and privacy risk issues [111]. MLLMs generally perform better with English, often a result of imbalanced training datasets that are mainly from English corpora. In this case, studies indicate that those MLLMs can usually perform better after the task is translated into English using professional translation tools, such as Google Translate. Additionally, the degree of syntactic similarity to English also significantly impacts the quality of these translations [112].

C. In Dynamic and Evolving Domains

Sectors like social media and news are characterized by constantly evolving language patterns, including the emergence of new slang, terminology, and usage trends. LLMs, with their continuous training and updating mechanisms, are adept at understanding and adapting to these changes. This adaptability ensures that models remain relevant and accurate over time, making LLMs particularly suitable for applications in dynamic content areas where staying current is crucial.

The ability to process and classify streams of fresh information without the need for frequent updates is where LLMs excel, making them suitable for dynamic content environments.

D. When Explanation is as Important as Classification

In sensitive fields such as healthcare or finance, the ability to explain decisions is almost as critical as making accurate classifications. LLMs can be prompted not only to classify but also to generate understandable narratives about their reasoning processes, aiding in compliance and transparency. Generating explanations alongside classifications helps build trust with end-users, who might rely on model outputs for critical decision-making.

LLaMA2 was trained on a dataset, IMHI, to build MentaLLaMA [27] to analyze mental health issues based on social media. MentaLLaMA can not only detect issues, such as depression detection and stress detection, correctly but also generate human-level explanations, which are significant in healthcare.

V. WHERE

Understanding the practical applications of LLMs in TC can significantly guide stakeholders in leveraging their capabilities across various industries. LLMs have demonstrated remarkable efficacy in automating and enhancing TC tasks due to their extensive training in diverse data sources and contexts. Table III illustrates various industries where LLMs can be pivotal, providing specific tasks and their corresponding application areas.

A. Healthcare

In the healthcare sector, LLMs have been employed to automate complex tasks such as medical diagnosis, health data analysis, and research classification. Models can parse through vast amounts of medical literature, electronic health records, and patient-generated data to assist healthcare professionals in making informed decisions. For instance, by classifying patient symptoms and predicting potential diagnoses along with

detailed explanations, LLMs can notably improve efficiency and accuracy in clinical settings [15], [18], [19], [27], [93], [94].

B. Environmental Science

TC in environmental science benefits from LLMs through tasks such as forest cover type classification and agricultural TC. These tasks involve analyzing textual data from diverse environmental reports and scientific studies to facilitate better decision-making in conservation and agricultural management [25], [95].

C. Finance

In the financial industry, LLMs are indispensable for sentiment and emotion analysis, financial document classification, and advisory services. By efficiently classifying financial news, reports, and sentiment from social media, LLMs can help in making informed investment decisions and risk assessments [9], [12], [13], [96]–[99].

D. Education

Educational institutions leverage LLMs for educational assessment analysis and material categorization. LLMs can analyze students' textual responses to classify their proficiency levels, recommend personalized learning materials, and assess educational outcomes more effectively [14], [29], [93], [100], [101].

E. Technology and Cybersecurity

In the realm of technology and cybersecurity, LLMs are employed for tasks like cyber threat detection, such as phishing and spam email detection, vulnerability identification, and sentiment classification. They can efficiently parse through security logs, threat reports, and even user-generated content to identify potential security threats and classify them based on severity [16], [17], [22], [26], [102]–[105].

F. Biological Sciences

For biological sciences, tasks such as biological entity recognition, molecular classification, and literature sorting are paramount. LLMs enable researchers to quickly classify and organize large volumes of biological data and literature, thereby facilitating more efficient research processes and discoveries [10], [28], [106].

G. Legal

In the legal industry, LLMs enhance the automation of legal document classification. By classifying and organizing vast amounts of legal documents, case files, and statutes, LLMs streamline the workflow of legal practitioners, aiding in faster retrieval and analysis of relevant legal information [17], [20], [107]–[109].

H. Media and Entertainment

In media and entertainment, LLMs are used for topic categorization and sentiment analysis of reviews and audience feedback. They automate the classification of news, user reviews, social media comments, and other forms of audience engagement, aiding companies in understanding consumer interested topics and sentiment and tailoring their content accordingly [10], [15], [21], [97].

Across these diverse domains, the inherent strengths of LLMs, such as their ability to learn and adapt to varied contexts, make them exceptionally suited for enhancing TC tasks, while their limitations, including computational demands and potential biases, necessitate cautious and informed deployment.

VI. How

The previous section has shown the impressive capabilities of LLMs in dealing with TC. However, it is essential to understand the various methodologies that can be employed to enhance their performance instead of providing the unstructured text to LLMs directly and asking them to provide the results, usually named zero-shot. This section explores how different approaches—such as Few-Shot, Chain of Thought(CoT), Retrieval-Augmented Generation(RAG), and fine-tuning—can significantly improve the effectiveness of LLMs in TC tasks. Table IV shows these methodologies and their corresponding related research. The *Others* row includes employing iterative prompting, reinforcement learning, ensemble learning, etc., to improve the TC tasks.

TABLE IV
A summary of methods employed by LLMs in text classification

Methods	Research
Prompt Engineering	[9], [12], [23], [24], [28], [49], [77], [93], [95], [96], [99], [100], [102]–[106], [113]–[115]
RAG	[9], [19], [94], [98]
Instruction Tuning	[12], [27], [77], [98]
Fine-tuning	[9], [10], [12], [12]–[15], [15]–[17], [19], [21], [22], [26], [28], [49], [77], [97], [103], [104], [107], [110], [113]
Others	[18], [19], [110], [116]

A. Zero-shot

Zero-shot learning(ZSL) for TC is a paradigm where a model that only provides natural language task descriptions without any detailed examples can classify text into the desired categories [53]. Unlike traditional supervised learning methods that require a large corpus of labeled data for each new task [34], or traditional ZSL may require additional descriptions for the unseen labels [117], LLMs are able to understand the context and predict labels for unseen tasks directly. The top left sub-figure in Fig. 2 shows a ZSL example for sentiment classification.

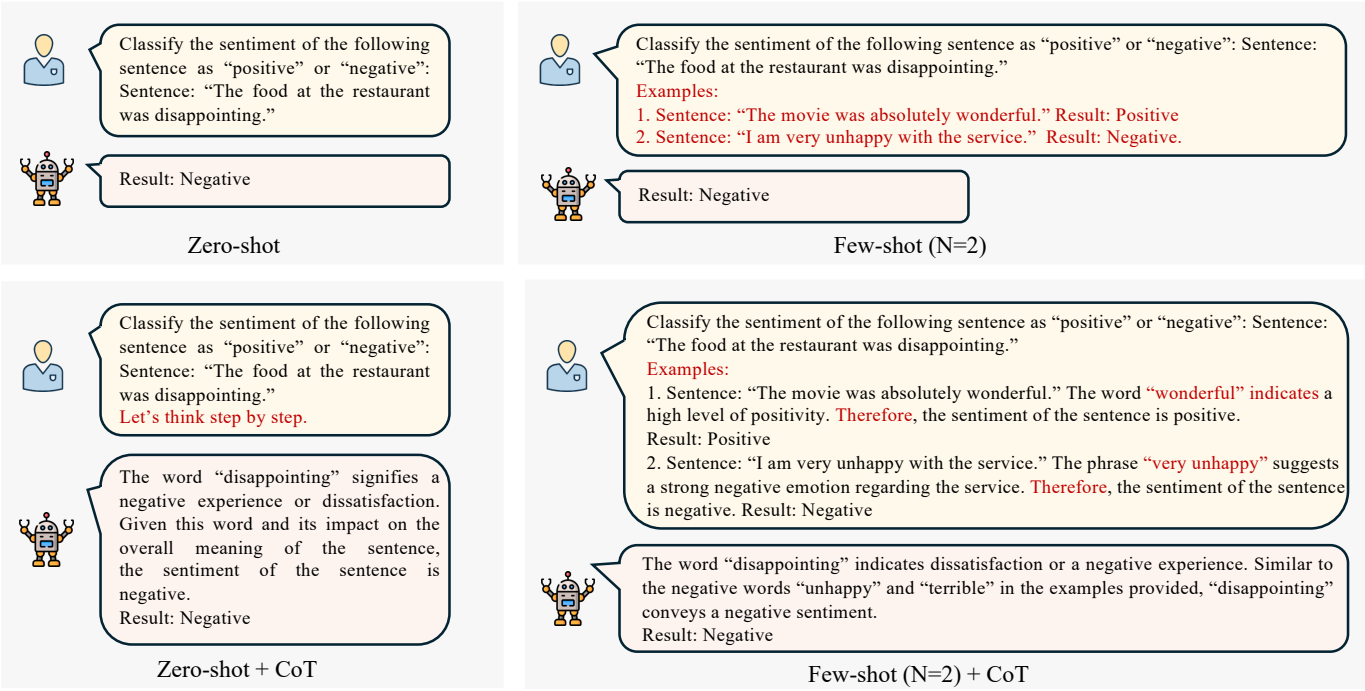

Fig. 2. The differences of Zero-shot, Zero-shot + CoT, Few-shot (N=2), Few-shot (N=2) + CoT

In the context of zero-shot TC, the task involves providing a textual input and having the model output the most relevant label from a predefined set of categories. For example, consider a social media post annotated with stances expressed towards Donald Trump and Hillary Clinton during the 2016 election [118]. An LLM could classify the post into categories such as "oppose," "support," or "neutral" without prior explicit training on a dataset specifically labeled with these categories [49].

Despite its advantages, ZSL has several limitations: it often struggles with ambiguous or nuanced tasks, lacks the context provided by labeled examples [53], [119], and can misinterpret the expected output format [113]. Moreover, it may generalize poorly on domain-specific tasks that require specialized knowledge not covered in pre-training, leading to potential misclassifications. Providing clear examples or fine-tuning the model can help mitigate these issues.

B. Few-shot

Few-shot in the context of LLMs refers to the ability of these models to perform tasks after being given only a few examples of the desired task within the inference input, which are presented purely as text, without requiring gradient updates or fine-tuning [53], [120]. This method leverages the extensive knowledge encoded in the models during their pre-training phase on diverse large-scale corpora. The top right sub-figure in Fig. 2 shows an example of few-shot-learning.

Few-shot prompting strategy has succeeded in various tasks, including sentence completion, clinical information extraction, reasoning tasks, tabular data classification [121], and TC. Larger LLMs typically exhibit superior performance in learning from a few demonstrations due to their extensive parameter capacity.

However, the few-shot prompting method is not without its limitations. The effectiveness of these strategies can be inconsistent, as factors such as the number of examples, the order in which they are presented, and the specific examples chosen can all significantly impact the results [122]. Additionally, LLMs may strongly rely on superficial cues rather than truly understanding the underlying task [123].

Another notable limitation is the constraint on the input and memory context size that LLMs can handle. This restricts the number of examples that can be provided, especially if those examples are lengthy. Taking the powerful GPT-3 (*gpt-3.5-turbo)* [124] for example, the context length limits the sum of the tokens from input plus output up to 16,385 tokens. Such constraints can hinder the application of few-shot learning in scenarios where large or numerous examples are necessary for optimal performance.

C. Chain of Thought

CoT [125] prompting is a technique designed to enhance the reasoning capabilities of LLMs by guiding them to process information in a sequential, step-by-step manner. By prompting the models to "Let's think step by step" or to "Show your thoughts," CoT encourages a more transparent decision-making process. This method not only aids in following the model's logic but also assists in debugging and improving the model's outputs. Employing CoT can be particularly transformative in tasks that require detailed reasoning or complex

decision-making, as it essentially simulates a more human-like approach to problem-solving within LLMs.

Applying CoT to TC, specifically to intent classification tasks, allows LLMs to reason through the classification process explicitly [126]. This means that for each piece of text, the model generates a reasoning chain that maps out the logical steps leading to its classification decision, aligning each text to its predicted intent class. This process not only enhances classification accuracy but also provides valuable insights into the model's thought process, making its decisions more interpretable [127], [128]. The bottom sub-figures in Fig. 2 indicate the idea of CoT with ZSL and few-shot learning.

In practical applications [129]–[131], CoT has shown substantial benefits in domains requiring nuanced classification. For example, when evaluated on TC tasks using datasets like CLINC-150, which includes 150 intents across 10 domains, and Banking77 with 77 banking-related intents, CoT with few-shot integrated within the LLaMA2-13B model demonstrated promising results, achieving performance close to that of GPT-4 [126]. Additionally, in the medical domain, the development of an incremental reasoning CoT prompted methodology reflects the step-by-step decision-making process typical in clinical environments [114]. This approach not only aligns the model's reasoning with real-life medical diagnostics but also enhances the clarity and reliability of its outputs, showcasing the broad applicational potential of CoT in improving both the accuracy and interpretability of TC tasks across diverse fields.

D. Retrieval-Augmented Generation

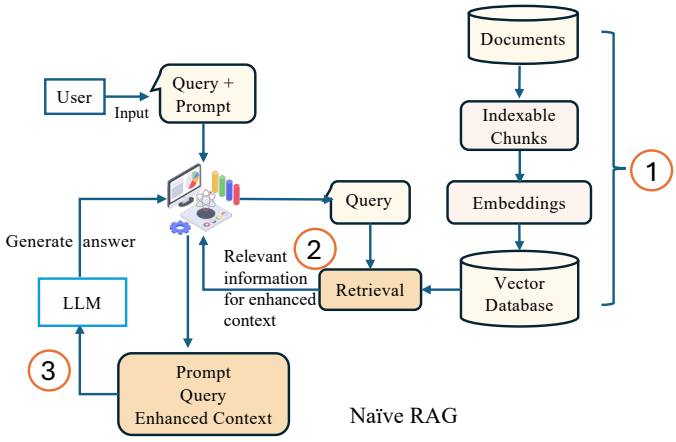

Fig. 3. The flow of naive RAG

RAG [132] represents a sophisticated approach tailored to enhance the capabilities of LLMs by addressing some of their inherent limitations. Typically, LLMs, like GPT-3, derive their knowledge from vast amounts of pre-trained data. This encapsulated knowledge, static by nature, is confined within the parameters set during training. Another thing is that the ability of LLMs to handle the length of input is limited. As a result, the models may exhibit issues such as relying on outdated information or lacking background information due to input limitation, producing irrelevant or inaccurate content, and generating data that may not adhere strictly to factual accuracy (commonly referred to as "hallucination") [133], [134]. RAG confronts these challenges head-on by dynamically integrating external sources of information into the generation process [135]. This not only enriches the context for response generation but also ensures that the information being used is up-to-date and factually accurate, significantly reducing the likelihood of errors and enhancing the model's reliability.

The naive operational mechanism of RAG can be broadly segmented into three pivotal steps [133]. Fig. 3 shows the flow for this process. The first step involves indexing, where documents are split into smaller, manageable chunks. These chunks are then encoded into dense vector representations and stored in a vector database, facilitating rapid and efficient retrieval. The second step is retrieval, and based on the input task(also known as query, Q), the system retrieves the top 'k' document chunks with the highest semantic similarity to the task. This retrieval is powered by vector representations, allowing for an accurate selection of information that closely aligns with the query's context. The third step is generation. In this phase, both the retrieved content chunks and the original task prompt are fed into the LLM. This enriched input allows the LLM to generate outputs that are not only contextually enriched but also updated and factually precise, leading to more accurate and relevant results.

Incorporating RAG into LLMs brings significant advancements in TC tasks. By extending the knowledge base beyond static training datasets, RAG enables models to access and utilize the latest information, akin to a continuous learning process. A recent study [136] demonstrates that dynamically adding a set of samples retrieved by an external pre-trained dense retriever model can significantly improve multi-label classification tasks. This improvement occurs because the prompt includes only the most relevant labels for the current example, making the classification process more accurate and efficient.

RAG technique is also particularly valuable in fields where new data emerges rapidly, such as news categorization, medical research updates, or dynamic regulatory changes. For instance, in medical diagnosis classification, RAG can provide the most current research findings or case studies [94], [137], allowing the model to make more informed and accurate classifications based on the latest medical insights. Similarly, in legal document classification, RAG can fetch the most recent laws and legal precedents relevant to the case at hand [138]–[140], thereby enhancing the model's precision and reliability. Hence, RAG not only boosts the accuracy of classifications by keeping the information current but also adapts to changes in data over time, which is critical for maintaining the continued relevancy and effectiveness of LLMs in practical applications.

E. Fine-tuning

Unlike prompting strategies such as zero-shot, CoT, few-shot, and RAG, which primarily modify the input text fed into

an LLM without altering the model's internal parameters, fine-tuning is a distinct approach. Fine-tuning is a widely used deep learning technique that involves further training a pre-trained model with task-specific data [141]. This process adjusts the LLM's parameters to enhance performance on targeted tasks, tailoring the model to produce outputs optimally aligned with specific requirements.

While larger LLMs generally deliver superior performance across a broad spectrum of tasks due to their extensive training datasets and capacity, smaller LLMs can attain comparable results on specific tasks through fine-tuning with only a limited number of data points [121], [142], such as a fine-tuned 7B model can superior to GPT-4 [113] in TC tested over four datasets. Moreover, these fine-tuned models can be further enhanced through few-shot prompting, allowing them to adapt quickly to new, related tasks with minimal data.

However, the specialization gained through fine-tuning comes with trade-offs. While a fine-tuned LLM often achieves higher accuracy and outputs that are better adapted to specific use cases, recent studies [143], [144] show that this specialization may compromise or significantly drop the model's performance on more general tasks for which it was not specifically trained. Additionally, fine-tuning demands access to appropriately labeled data, which may not always be available. The process also requires significant computational resources and can be time-consuming, even if the model only has around 7 billion parameters [145], factors that must be considered when planning to fine-tune LLMs.

VII. Limitation and Challenges

While LLMs bring significant advancements to the TC landscape, they are not without their own set of limitations and challenges. Understanding these gaps is crucial for effectively deploying LLMs in TC tasks.

A. Non-Standard Output

Unlike traditional ML or DL models that produce well-defined, consistent outputs within a specified scope, LLMs can sometimes yield non-standard outputs [146]. For instance, there are instances where the model may repeat parts of the input text, produce irrelevant responses [147], or assign labels that diverge from the predefined set. This variability can complicate the integration of LLMs into systems that demand high precision and reliability.

B. Content Sensitivity

LLMs are typically programmed with content filters to avoid generating risky content such as hate speech, politically sensitive, adult, or gambling-related material [147], [148]. While these filters aim to ensure ethical AI usage, they pose a significant challenge in contexts where the ability to process sensitive content is crucial—such as spam or fraud detection systems. The refusal to engage with certain types of content can create blind spots in these critical applications.

C. Input Size Limitations

Handling long text is another area where LLMs face limitations due to their maximum input size constraints [87]. Some models are restricted to 4096 tokens or fewer, which can be insufficient for documents requiring comprehensive analysis. This restriction necessitates splitting longer texts into manageable chunks, which can lead to a loss of context and reduced classification accuracy. At the same time, with the development of LLMs, some models, such as Claude 3, can support up to 200K, and the abilities decay when the input size increases.

D. Computational Resource Intensiveness

The pre-train, fine-tuning, and deployment of LLMs are computationally demanding [149]. These models require significantly more processing power and memory compared to traditional ML and DL models. The high computational cost can limit the feasibility of using LLMs, particularly in resource-constrained environments or real-time applications where rapid inference is essential.

E. Ethical and Bias Considerations

LLMs often inherit biases present in their training data [150], [151]. These biases can manifest in various forms, from gender and racial biases to cultural and linguistic prejudices [152]. Ensuring fair and unbiased output from LLMs remains a significant challenge and necessitates ongoing vigilance and corrective measures.

In summary, while LLMs show great capabilities for TC, addressing their limitations is essential for maximizing their utility. Continued research and development are needed to mitigate these challenges and fully harness the potential of LLMs in diverse TC applications. As we advance, a balanced approach leveraging both the strengths of LLMs and the reliability of traditional methods may offer the most robust solutions for TC challenges.

VIII. Future Research Directions

Although LLMs have demonstrated significant potential in text classification, there are several areas where further research can enhance their effectiveness and broaden their applicability. Based on the insights from our survey, we outline several future research directions.

A. Diversifying Methods Employed by LLMs in Text Classification

While methods like prompt engineering and fine-tuning have been widely employed for TC, other techniques such as RAG, instruction tuning, and multi-modal techniques are underexplored. Further research could investigate these less-utilized methodologies to assess their potential to improve TC performance and efficiency.

B. Broadening Domain Applications

Although LLMs have seen substantial application across sectors such as medicine, finance, and technology, their deployment in fields like agriculture, social welfare, public policy, and more remains limited. These areas often deal with specialized vocabulary and unique contextual challenges that LLMs could learn and adapt to. Research directed towards cultivating LLMs' capabilities in these fields could not only widen the applicability of these models but also result in significant societal impacts.

C. Leveraging LLMs Beyond Direct Classification

In addition to being used as direct text classifiers, LLMs can play a crucial role in supporting TC tasks through data augmentation, data interpretation, and information extraction. By generating synthetic data, LLMs can mitigate issues related to data scarcity and imbalance. As data interpreters, they can provide richer contextual analysis that enhances the decision-making processes of traditional ML and DL models. Furthermore, LLMs can serve as powerful information extractors, automatically identifying and highlighting key information within complex text corpora to streamline the classification process.

D. Integrating LLMs with Other Techniques

To unlock the full potential of LLMs in TC, it is essential to explore their integration with other advanced techniques. Combining LLMs with reinforcement learning methods, such as Markov Decision Processes or Q-learning, could enhance their ability to learn from sequential tasks and improve classification accuracy over time. Additionally, hybrid models that incorporate graph neural networks, attention mechanisms, or Bayesian optimization techniques could provide more robust and interpretable classification outcomes. Research in this direction could pave the way for more sophisticated and intelligent text classification systems.

IX. Conclusion

Throughout this paper, we have addressed the fundamental aspects of employing LLMs in TC tasks—detailing what LLMs are, why they are particularly suited for TC, when their use is most advantageous, where they can be applied most effectively, and how they can be harnessed to achieve optimal results. The insights provided here underscore the versatility and potential of LLMs to transform TC across a multitude of industries, from healthcare and finance to media and academia.

Moreover, our discussion highlighted not only the strengths but also the potential challenges associated with LLM deployment, such as issues related to data privacy, model bias, and the need for substantial computational resources. Recognizing these challenges is essential as we move toward more ethical and sustainable AI practices. We also discussed future research directions.

In conclusion, the integration of LLMs into TC processes is not just a technical improvement but a transformative shift in data analysis. This survey serves as a foundational text for those looking to understand or enhance their application of LLMs in TC, providing a pathway for future research and practical application in this dynamic field. As LLMs continue to evolve, so too will their impact on our ability to manage and interpret the growing digital landscape, promising ongoing enhancements in accuracy, efficiency, and contextual understanding in text-based analytics.

References

[1] J. Jiang, F. Wang, J. Shen, S. Kim, and S. Kim, "A survey on large language models for code generation," *arXiv preprint arXiv:2406.00515*, 2024.

[2] T. Zhang, F. Ladhak, E. Durmus, P. Liang, K. McKeown, and T. B. Hashimoto, "Benchmarking large language models for news summarization," *Transactions of the Association for Computational Linguistics*, vol. 12, pp. 39–57, 2024.

[3] 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.

[4] 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.

[5] 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*, vol. 15, no. 3, pp. 1–45, 2024.

[6] 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.

[7] A. Chowdhery, S. Narang, J. Devlin, M. Bosma, G. Mishra, A. Roberts, P. Barham, H. W. Chung, C. Sutton, S. Gehrmann *et al.*, "Palm: Scaling language modeling with pathways," *arXiv preprint arXiv:2204.02311*, 2022.

[8] Q. Li, H. Peng, J. Li, C. Xia, R. Yang, L. Sun, P. S. Yu, and L. He, "A survey on text classification: From traditional to deep learning," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 13, no. 2, pp. 1–41, 2022.

[9] L. Loukas, I. Stogiannidis, O. Diamantopoulos, P. Malakasiotis, and S. Vassos, "Making llms worth every penny: Resource-limited text classification in banking," in *Proceedings of the Fourth ACM International Conference on AI in Finance*, 2023, pp. 392–400.

[10] M. Bețianu, A. Mălan, M. Aldinucci, R. Birke, and L. Chen, "Dallmi: Domain adaption for llm-based multi-label classifier," in *Pacific-Asia Conference on Knowledge Discovery and Data Mining*. Springer, 2024, pp. 277–289.

[11] S. Xu, Z. Wu, H. Zhao, P. Shu, Z. Liu, W. Liao, S. Li, A. Sikora, T. Liu, and X. Li, "Reasoning before comparison: Llm-enhanced semantic similarity metrics for domain specialized text analysis," 2024. [Online]. Available: https://arxiv.org/abs/2402.11398

[12] R. S. Wahidur, I. Tashdeed, M. Kaur, and H.-N. Lee, "Enhancing zero-shot crypto sentiment with fine-tuned language model and prompt engineering," *IEEE Access*, 2024.

[13] R. Bhat and B. Jain, "Stock price trend prediction using emotion analysis of financial headlines with distilled llm model," in *Proceedings of the 17th International Conference on PErvasive Technologies Related to Assistive Environments*, 2024, pp. 67–73.

[14] J. Liu, P. Zhou, Y. Hua, D. Chong, Z. Tian, A. Liu, H. Wang, C. You, Z. Guo, L. Zhu *et al.*, "Benchmarking large language models on cmexam-a comprehensive chinese medical exam dataset," *Advances in Neural Information Processing Systems*, vol. 36, 2024.

[15] B. Csanády, L. Muzsai, P. Vedres, Z. Nádasdy, and A. Lukács, "Llambert: Large-scale low-cost data annotation in nlp," *arXiv preprint arXiv:2403.15938*, 2024.

[16] S. Jamal, H. Wimmer, and I. H. Sarker, "An improved transformer-based model for detecting phishing, spam and ham emails: A large language model approach," *Security and Privacy*, p. e402, 2024.

[17] W. Stigall, M. A. Al Hafiz Khan, D. Attota, F. Nweke, and Y. Pei, "Large language models performance comparison of emotion and sentiment classification," in *Proceedings of the 2024 ACM Southeast Conference*, 2024, pp. 60–68.

[18] Y. Feng, X. Xu, Y. Zhuang, and M. Zhang, "Large language models improve alzheimer's disease diagnosis using multi-modality data," in *2023 IEEE International Conference on Medical Artificial Intelligence (MedAI)*. IEEE, 2023, pp. 61–66.

[19] H. Zhang, J. Li, Y. Wang, and Y. Songi, "Integrating automated knowledge extraction with large language models for explainable medical decision-making," in *2023 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*. IEEE, 2023, pp. 1710–1717.

[20] A. Peña, A. Morales, J. Fierrez, I. Serna, J. Ortega-Garcia, I. Puente, J. Cordova, and G. Cordova, "Leveraging large language models for topic classification in the domain of public affairs," in *International Conference on Document Analysis and Recognition*. Springer, 2023, pp. 20–33.

[21] M. Rehan, M. S. I. Malik, and M. M. Jamjoom, "Fine-tuning transformer models using transfer learning for multilingual threatening text identification," *IEEE Access*, 2023.

[22] A. Ucan, M. Dörterler, and E. Akçapınar Sezer, "A study of turkish emotion classification with pretrained language models," *Journal of Information Science*, vol. 48, no. 6, pp. 857–865, 2022.

[23] V. D. Lai, N. T. Ngo, A. P. B. Veyseh, H. Man, F. Dernoncourt, T. Bui, and T. H. Nguyen, "Chatgpt beyond english: Towards a comprehensive evaluation of large language models in multilingual learning," *arXiv preprint arXiv:2304.05613*, 2023.

[24] Y. Chen, D. Harbecke, and L. Hennig, "Multilingual relation classification via efficient and effective prompting," *arXiv preprint arXiv:2210.13838*, 2022.

[25] B. Zhao, W. Jin, J. Del Ser, and G. Yang, "Chatagri: Exploring potentials of chatgpt on cross-linguistic agricultural text classification," *Neurocomputing*, vol. 557, p. 126708, 2023.

[26] M. A. Uddin and I. H. Sarker, "An explainable transformer-based model for phishing email detection: A large language model approach," *arXiv preprint arXiv:2402.13871*, 2024.

[27] K. Yang, T. Zhang, Z. Kuang, Q. Xie, J. Huang, and S. Ananiadou, "Mentallama: interpretable mental health analysis on social media with large language models," in *Proceedings of the ACM on Web Conference 2024*, 2024, pp. 4489–4500.

[28] C. Qian, H. Tang, Z. Yang, H. Liang, and Y. Liu, "Can large language models empower molecular property prediction?" *arXiv preprint arXiv:2307.07443*, 2023.

[29] P. Sharma, K. Thapa, D. Thapa, P. Dhakal, M. D. Upadhaya, S. Adhikari, and S. R. Khanal, "Performance of chatgpt on usmle: Unlocking the potential of large language models for ai-assisted medical education," *arXiv preprint arXiv:2307.00112*, 2023.

[30] J. Xu, Z. Wu, M. Lin, X. Zhang, and S. Wang, "Llm and gnn are complementary: Distilling llm for multimodal graph learning," *arXiv preprint arXiv:2406.01032*, 2024.

[31] W. X. Zhao, K. Zhou, J. Li, T. Tang, X. Wang, Y. Hou, Y. Min, B. Zhang, J. Zhang, Z. Dong *et al.*, "A survey of large language models," *arXiv preprint arXiv:2303.18223*, 2023.

[32] 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.

[33] M. U. Hadi, R. Qureshi, A. Shah, M. Irfan, A. Zafar, M. B. Shaikh, N. Akhtar, J. Wu, S. Mirjalili *et al.*, "A survey on large language models: Applications, challenges, limitations, and practical usage," *Authorea Preprints*, 2023.

[34] 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.

[35] H. Zhang, "The optimality of naive bayes," *Aa*, vol. 1, no. 2, p. 3, 2004.

[36] T. Joachims, "Text categorization with support vector machines: Learning with many relevant features," in *European conference on machine learning*. Springer, 1998, pp. 137–142.

[37] Y. Luan and S. Lin, "Research on text classification based on cnn and lstm," in *2019 IEEE international conference on artificial intelligence and computer applications (ICAICA)*. IEEE, 2019, pp. 352–355.

[38] G. Soyalp, A. Alar, K. Ozkanli, and B. Yildiz, "Improving text classification with transformer," in *2021 6th International Conference on Computer Science and Engineering (UBMK)*. IEEE, 2021, pp. 707–712.

[39] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," *Advances in neural information processing systems*, vol. 26, 2013.

[40] T. G. Dietterich, "Ensemble methods in machine learning," in *International workshop on multiple classifier systems*. Springer, 2000, pp. 1–15.

[41] S. Minaee, N. Kalchbrenner, E. Cambria, N. Nikzad, M. Chenaghlu, and J. Gao, "Deep learning–based text classification: a comprehensive review," *ACM computing surveys (CSUR)*, vol. 54, no. 3, pp. 1–40, 2021.

[42] G. Divita, G. Luo, L.-T. T. Tran, T. E. Workman, A. V. Gundlapalli, and M. H. Samore, "General symptom extraction from va electronic medical notes," in *MEDINFO 2017: Precision Healthcare through Informatics*. IOS Press, 2017, pp. 356–360.

[43] I. Spasic, G. Nenadic *et al.*, "Clinical text data in machine learning: systematic review," *JMIR medical informatics*, vol. 8, no. 3, p. e17984, 2020.

[44] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.

[45] G. Song, Y. Ye, X. Du, X. Huang, and S. Bie, "Short text classification: a survey." *Journal of multimedia*, vol. 9, no. 5, 2014.

[46] S. Chathuranga and S. Ranathunga, "Classification of code-mixed text using capsule networks," in *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2021)*, R. Mitkov and G. Angelova, Eds. Held Online: INCOMA Ltd., Sep. 2021, pp. 256–263. [Online]. Available: https://aclanthology.org/2021.ranlp-1.30

[47] N. H. Mahadzir *et al.*, "Sentiment analysis of code-mixed text: a review," *Turkish Journal of Computer and Mathematics Education (TURCOMAT)*, vol. 12, no. 3, pp. 2469–2478, 2021.

[48] S. Chathuranga and S. Ranathunga, "Classification of code-mixed text using capsule networks," in *Proceedings of the international conference on recent advances in natural language processing (RANLP 2021)*, 2021, pp. 256–263.

[49] Y. Chae and T. Davidson, "Large language models for text classification: From zero-shot learning to fine-tuning," *Open Science Foundation*, 2023.

[50] Z. Wang, Y. Pang, and Y. Lin, "Large language models are zero-shot text classifiers," *arXiv preprint arXiv:2312.01044*, 2023.

[51] S. Wang, Y. Liu, Y. Xu, C. Zhu, and M. Zeng, "Want to reduce labeling cost? gpt-3 can help," *arXiv preprint arXiv:2108.13487*, 2021.

[52] 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.

[53] 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*, vol. 33, pp. 1877–1901, 2020.

[54] S. Nagpal, S. Dargan, H. Koneru, and S. Rastogi, "Innovations in code-mixed hate speech detection: The llm perspective."

[55] J. Fields, K. Chovanec, and P. Madiraju, "A survey of text classification with transformers: How wide? how large? how long? how accurate? how expensive? how safe?" *IEEE Access*, 2024.

[56] 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.

[57] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut, "Albert: A lite bert for self-supervised learning of language representations," *arXiv preprint arXiv:1909.11942*, 2019.

[58] P. He, X. Liu, J. Gao, and W. Chen, "Deberta: Decoding-enhanced bert with disentangled attention," *arXiv preprint arXiv:2006.03654*, 2020.

[59] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. R. Salakhutdinov, and Q. V. Le, "Xlnet: Generalized autoregressive pretraining for language understanding," *Advances in neural information processing systems*, vol. 32, 2019.

[60] A. Radford, K. Narasimhan, T. Salimans, I. Sutskever *et al.*, "Improving language understanding by generative pre-training," 2018.

[61] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever *et al.*, "Language models are unsupervised multitask learners," *OpenAI blog*, vol. 1, no. 8, p. 9, 2019.

[62] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," *The Journal of Machine Learning Research*, vol. 21, no. 1, pp. 5485–5551, 2020.

[63] L. Xue, N. Constant, A. Roberts, M. Kale, R. Al-Rfou, A. Siddhant, A. Barua, and C. Raffel, "mt5: A massively multilingual pre-trained text-to-text transformer," *arXiv preprint arXiv:2010.11934*, 2020.

[64] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer, "Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension," *arXiv preprint arXiv:1910.13461*, 2019.

[65] M. Chen, J. Tworek, H. Jun, Q. Yuan, H. P. D. O. Pinto, J. Kaplan, H. Edwards, Y. Burda, N. Joseph, G. Brockman *et al.*, "Evaluating large language models trained on code," *arXiv preprint arXiv:2107.03374*, 2021.

[66] R. Nakano, J. Hilton, S. Balaji, J. Wu, L. Ouyang, C. Kim, C. Hesse, S. Jain, V. Kosaraju, W. Saunders *et al.*, "Webgpt: Browser-assisted question-answering with human feedback," *arXiv preprint arXiv:2112.09332*, 2021.

[67] OpenAI, "GPT-4 Technical Report," https://arxiv.org/pdf/2303.08774v3.pdf, 2023.

[68] ——. (2024) Hello GPT-4o. https://openai.com/index/hello-gpt-4o.

[69] 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.

[70] R. Taori, I. Gulrajani, T. Zhang, Y. Dubois, X. Li, C. Guestrin, P. Liang, and T. B. Hashimoto, "Alpaca: A strong, replicable instruction-following model," *Stanford Center for Research on Foundation Models. https://crfm. stanford. edu/2023/03/13/alpaca. html*, vol. 3, no. 6, p. 7, 2023.

[71] L. ORG, "Vicuna: An open-source chatbot impressing gpt-4 with 90

[72] A. Q. Jiang, A. Sablayrolles, A. Mensch, C. Bamford, D. S. Chaplot, D. d. l. Casas, F. Bressand, G. Lengyel, G. Lample, L. Saulnier *et al.*, "Mistral 7b," *arXiv preprint arXiv:2310.06825*, 2023.

[73] B. Roziere, J. Gehring, F. Gloeckle, S. Sootla, I. Gat, X. E. Tan, Y. Adi, J. Liu, T. Remez, J. Rapin *et al.*, "Code llama: Open foundation models for code," *arXiv preprint arXiv:2308.12950*, 2023.

[74] A. Dubey, A. Jauhri, A. Pandey, A. Kadian, A. Al-Dahle, A. Letman, A. Mathur, A. Schelten, A. Yang, A. Fan *et al.*, "The llama 3 herd of models," *arXiv preprint arXiv:2407.21783*, 2024.

[75] Y. Tay, J. Wei, H. W. Chung, V. Q. Tran, D. R. So, S. Shakeri, X. Garcia, H. S. Zheng, J. Rao, A. Chowdhery *et al.*, "Transcending scaling laws with 0.1% extra compute," *arXiv preprint arXiv:2210.11399*, 2022.

[76] R. Anil, A. M. Dai, O. Firat, M. Johnson, D. Lepikhin, A. Passos, S. Shakeri, E. Taropa, P. Bailey, Z. Chen *et al.*, "Palm 2 technical report," *arXiv preprint arXiv:2305.10403*, 2023.

[77] K. Singhal, S. Azizi, T. Tu, S. S. Mahdavi, J. Wei, H. W. Chung, N. Scales, A. Tanwani, H. Cole-Lewis, S. Pfohl *et al.*, "Large language models encode clinical knowledge," *arXiv preprint arXiv:2212.13138*, 2022.

[78] J. Wei, M. Bosma, V. Y. Zhao, K. Guu, A. W. Yu, B. Lester, N. Du, A. M. Dai, and Q. V. Le, "Finetuned language models are zero-shot learners," *arXiv preprint arXiv:2109.01652*, 2021.

[79] R. Thoppilan, D. De Freitas, J. Hall, N. Shazeer, A. Kulshreshtha, H.-T. Cheng, A. Jin, T. Bos, L. Baker, Y. Du *et al.*, "Lamda: Language models for dialog applications," *arXiv preprint arXiv:2201.08239*, 2022.

[80] mixtral. mixtral. [Online]. Available: https://mistral.ai/news/mixtral-of-experts/

[81] J. Bai, S. Bai, Y. Chu, Z. Cui, K. Dang, X. Deng, Y. Fan, W. Ge, Y. Han, F. Huang *et al.*, "Qwen technical report," *arXiv preprint arXiv:2309.16609*, 2023.

[82] A. Yang, B. Yang, B. Hui, B. Zheng, B. Yu, C. Zhou, C. Li, C. Li, D. Liu, F. Huang *et al.*, "Qwen2 technical report," *arXiv preprint arXiv:2407.10671*, 2024.

[83] Anthropic, "Introducing the next generation of Claude," https://www.anthropic.com/news/claude-3-family, 2024.

[84] M. Reid, N. Savinov, D. Teplyashin, D. Lepikhin, T. Lillicrap, J.-b. Alayrac, R. Soricut, A. Lazaridou, O. Firat, J. Schrittwieser *et al.*, "Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context," *arXiv preprint arXiv:2403.05530*, 2024.

[85] D. Guo, Q. Zhu, D. Yang, Z. Xie, K. Dong, W. Zhang, G. Chen, X. Bi, Y. Wu, Y. Li *et al.*, "Deepseek-coder: When the large language model meets programming–the rise of code intelligence," *arXiv preprint arXiv:2401.14196*, 2024.

[86] D. Wang, N. Raman, M. Sibue, Z. Ma, P. Babkin, S. Kaur, Y. Pei, A. Nourbakhsh, and X. Liu, "Docllm: A layout-aware generative language model for multimodal document understanding," *arXiv preprint arXiv:2401.00908*, 2023.

[87] S. Minaee, T. Mikolov, N. Nikzad, M. Chenaghlu, R. Socher, X. Amatriain, and J. Gao, "Large language models: A survey," *arXiv preprint arXiv:2402.06196*, 2024.

[88] Meta, "Introducing meta llama 3: The most capable openly available llm to date," https://ai.meta.com/blog/meta-llama-3/, April 2024, accessed: 2024-06-21.

[89] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray *et al.*, "Training language models to follow instructions with human feedback," *Advances in neural information processing systems*, vol. 35, pp. 27 730–27 744, 2022.

[90] Y. H. Yeo, J. S. Samaan, W. H. Ng, X. Ma, P.-S. Ting, M.-S. Kwak, A. Panduro, B. Lizaola-Mayo, H. Trivedi, A. Vipani *et al.*, "Gpt-4 outperforms chatgpt in answering non-english questions related to cirrhosis," *medRxiv*, pp. 2023–05, 2023.

[91] Anthropic, "Anthropic Claude," https://www.anthropic.com/claude, accessed: 2024-06-21.

[92] Z. jiang, G. Hao, Y. He, K. Chen, Y. Wang, and Q. Zhu, "An online-offline computing mode based on apache storm for text classification," in *2019 Chinese Automation Congress (CAC)*, 2019, pp. 3385–3390.

[93] H. Nori, N. King, S. M. McKinney, D. Carignan, and E. Horvitz, "Capabilities of gpt-4 on medical challenge problems," *arXiv preprint arXiv:2303.13375*, 2023.

[94] Z. Huang, K. Xue, Y. Fan, L. Mu, R. Liu, T. Ruan, S. Zhang, and X. Zhang, "Tool calling: Enhancing medication consultation via retrieval-augmented large language models," *arXiv preprint arXiv:2404.17897*, 2024.

[95] L. Caruccio, S. Cirillo, G. Polese, G. Solimando, S. Sundaramurthy, and G. Tortora, "Claude 2.0 large language model: tackling a real-world classification problem with a new iterative prompt engineering approach," *Intelligent Systems with Applications*, p. 200336, 2024.

[96] P. Niszczota and S. Abbas, "Gpt as a financial advisor," *Available at SSRN 4384861*, 2023.

[97] X. Deng, V. Bashlovkina, F. Han, S. Baumgartner, and M. Bendersky, "Llms to the moon? reddit market sentiment analysis with large language models," in *Companion Proceedings of the ACM Web Conference 2023*, 2023, pp. 1014–1019.

[98] B. Zhang, H. Yang, T. Zhou, M. Ali Babar, and X.-Y. Liu, "Enhancing financial sentiment analysis via retrieval augmented large language models," in *Proceedings of the fourth ACM international conference on AI in finance*, 2023, pp. 349–356.

[99] F. Xing, "Designing heterogeneous llm agents for financial sentiment analysis," *arXiv preprint arXiv:2401.05799*, 2024.

[100] M. Angel, A. Patel, A. Alachkar, and P. Baldi, "Clinical knowledge and reasoning abilities of large language models in pharmacy: A comparative study on the naplex exam," in *2023 Tenth International Conference on Social Networks Analysis, Management and Security (SNAMS)*. IEEE, 2023, pp. 1–4.

[101] M. Angel, A. Patel, H. Xing, D. Balsz, C. Arbuckle, D. Bruyette, and P. Baldi, "Ai and veterinary medicine: performance of large language models on the north american licensing examination," in *2023 Tenth International Conference on Social Networks Analysis, Management and Security (SNAMS)*. IEEE, 2023, pp. 1–4.

[102] J. G. M. Mboma, K. Lusala, M. Matalatala, O. T. Tshipata, P. S. Nzakuna, and D. T. Kazumba, "Integrating llm with blockchain and ipfs to enhance academic diploma integrity," in *2024 International Conference on Artificial Intelligence, Computer, Data Sciences and Applications (ACDSA)*. IEEE, 2024, pp. 1–6.

[103] V. Akuthota, R. Kasula, S. T. Sumona, M. Mohiuddin, M. T. Reza, and M. M. Rahman, "Vulnerability detection and monitoring using llm," in *2023 IEEE 9th International Women in Engineering (WIE) Conference on Electrical and Computer Engineering (WIECON-ECE)*. IEEE, 2023, pp. 309–314.

[104] F. Trad and A. Chehab, "Prompt engineering or fine-tuning? a case study on phishing detection with large language models," *Machine Learning and Knowledge Extraction*, vol. 6, no. 1, pp. 367–384, 2024.

[105] T. Koide, N. Fukushi, H. Nakano, and D. Chiba, "Chatspamdetector: Leveraging large language models for effective phishing email detection," *arXiv preprint arXiv:2402.18093*, 2024.

[106] S. J. Jung, H. Kim, and K. S. Jang, "Llm based biological named entity recognition from scientific literature," in *2024 IEEE International*

Conference on Big Data and Smart Computing (BigComp). IEEE, 2024, pp. 433–435.

[107] F. Wei, R. Keeling, N. Huber-Fliflet, J. Zhang, A. Dabrowski, J. Yang, Q. Mao, and H. Qin, "Empirical study of llm fine-tuning for text classification in legal document review," in *2023 IEEE International Conference on Big Data (BigData)*. IEEE, 2023, pp. 2786–2792.

[108] A. Tewari, "Legalpro-bert: Classification of legal provisions by fine-tuning bert large language model," *arXiv preprint arXiv:2404.10097*, 2024.

[109] N. Prasad, M. Boughanem, and T. Dkaki, "Exploring large language models and hierarchical frameworks for classification of large unstructured legal documents," in *European Conference on Information Retrieval*. Springer, 2024, pp. 221–237.

[110] H. Abburi, M. Suesserman, N. Pudota, B. Veeramani, E. Bowen, and S. Bhattacharya, "Generative ai text classification using ensemble llm approaches," *arXiv preprint arXiv:2309.07755*, 2023.

[111] L. Qin, Q. Chen, Y. Zhou, Z. Chen, Y. Li, L. Liao, M. Li, W. Che, and P. S. Yu, "Multilingual large language model: A survey of resources, taxonomy and frontiers," *arXiv preprint arXiv:2404.04925*, 2024.

[112] C. Liu, W. Zhang, Y. Zhao, A. T. Luu, and L. Bing, "Is translation all you need? a study on solving multilingual tasks with large language models," *arXiv preprint arXiv:2403.10258*, 2024.

[113] Z. Wang, Y. Pang, and Y. Lin, "Smart expert system: Large language models as text classifiers," *arXiv preprint arXiv:2405.10523*, 2024.

[114] O. Gramopadhye, S. S. Nachane, P. Chanda, G. Ramakrishnan, K. S. Jadhav, Y. Nandwani, D. Raghu, and S. Joshi, "Few shot chain-of-thought driven reasoning to prompt llms for open ended medical question answering," *arXiv preprint arXiv:2403.04890*, 2024.

[115] X. Sun, X. Li, J. Li, F. Wu, S. Guo, T. Zhang, and G. Wang, "Text classification via large language models," *arXiv preprint arXiv:2305.08377*, 2023.

[116] X. Sun, X. Li, S. Zhang, S. Wang, F. Wu, J. Li, T. Zhang, and G. Wang, "Sentiment analysis through llm negotiations," *arXiv preprint arXiv:2311.01876*, 2023.

[117] B. Romera-Paredes and P. Torr, "An embarrassingly simple approach to zero-shot learning," in *International conference on machine learning*. PMLR, 2015, pp. 2152–2161.

[118] S. Mohammad, S. Kiritchenko, P. Sobhani, X. Zhu, and C. Cherry, "Semeval-2016 task 6: Detecting stance in tweets," in *Proceedings of the 10th international workshop on semantic evaluation (SemEval-2016)*, 2016, pp. 31–41.

[119] Z. Wang, A. W. Yu, O. Firat, and Y. Cao, "Towards zero-label language learning," *arXiv preprint arXiv:2109.09193*, 2021.

[120] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Language models are unsupervised multitask learners," 2019. [Online]. Available: https://api.semanticscholar.org/CorpusID:160025533

[121] S. Hegselmann, A. Buendia, H. Lang, M. Agrawal, X. Jiang, and D. Sontag, "Tabllm: Few-shot classification of tabular data with large language models," in *International Conference on Artificial Intelligence and Statistics*. PMLR, 2023, pp. 5549–5581.

[122] L. Weber, E. Bruni, and D. Hupkes, "Mind the instructions: a holistic evaluation of consistency and interactions in prompt-based learning," *arXiv preprint arXiv:2310.13486*, 2023.

[123] W. M. I.-C. L. Work, "Rethinking the role of demonstrations: What makes in-context learning work?"

[124] OpenAI, "Gpt-3.5 turbo," 2024, accessed: 2024-06-21. [Online]. Available: https://platform.openai.com/docs/models/gpt-3-5-turbo

[125] 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.

[126] D. Koutsianos, T. Stafylakis, and P. Tassias, "Chain of thought prompting for intent classification using large language models," 2024.

[127] Z. Chu, J. Chen, Q. Chen, W. Yu, T. He, H. Wang, W. Peng, M. Liu, B. Qin, and T. Liu, "A survey of chain of thought reasoning: Advances, frontiers and future," *arXiv preprint arXiv:2309.15402*, 2023.

[128] Z. Yu, L. He, Z. Wu, X. Dai, and J. Chen, "Towards better chain-of-thought prompting strategies: A survey," *arXiv preprint arXiv:2310.04959*, 2023.

[129] S. Kim, S. J. Joo, Y. Jang, H. Chae, and J. Yeo, "Cotever: Chain of thought prompting annotation toolkit for explanation verification," *arXiv preprint arXiv:2303.03628*, 2023.

[130] S. Diao, P. Wang, Y. Lin, and T. Zhang, "Active prompting with chain-of-thought for large language models," *arXiv preprint arXiv:2302.12246*, 2023.

[131] P. Aggarwal, A. Madaan, Y. Yang, and Mausam, "Let's sample step by step: Adaptiveconsistency for efficient reasoning with llms," *arXiv preprint arXiv:2305.11860*, 2023.

[132] 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.

[133] 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.

[134] Z. Ji, N. Lee, R. Frieske, T. Yu, D. Su, Y. Xu, E. Ishii, Y. J. Bang, A. Madotto, and P. Fung, "Survey of hallucination in natural language generation," *ACM Computing Surveys*, vol. 55, no. 12, pp. 1–38, 2023.

[135] Y. Ding, W. Fan, L. Ning, S. Wang, H. Li, D. Yin, T.-S. Chua, and Q. Li, "A survey on rag meets llms: Towards retrieval-augmented large language models," *arXiv preprint arXiv:2405.06211*, 2024.

[136] A. Milios, S. Reddy, and D. Bahdanau, "In-context learning for text classification with many labels," in *Proceedings of the 1st GenBench Workshop on (Benchmarking) Generalisation in NLP*, 2023, pp. 173–184.

[137] J. C. L. Ong, L. Jin, K. Elangovan, G. Y. S. Lim, D. Y. Z. Lim, G. G. R. Sng, Y. Ke, J. Y. M. Tung, R. J. Zhong, C. M. Y. Koh *et al.*, "Development and testing of a novel large language model-based clinical decision support systems for medication safety in 12 clinical specialties," *arXiv preprint arXiv:2402.01741*, 2024.

[138] MyScale, "3 ways rag technology transforms legal research and analysis," 2024, accessed: 2024-07-06. [Online]. Available: https://myscale.com/blog/rag-technologylegal-research-analysis-transformations/

[139] N. Wiratunga, R. Abeyratne, L. Jayawardena, K. Martin, S. Massie, I. Nkisi-Orji, R. Weerasinghe, A. Liret, and B. Fleisch, "Cbr-rag: case-based reasoning for retrieval augmented generation in llms for legal question answering," in *International Conference on Case-Based Reasoning*. Springer, 2024, pp. 445–460.

[140] D. Tuggener, P. Von Däniken, T. Peetz, and M. Cieliebak, "Ledgar: A large-scale multi-label corpus for text classification of legal provisions in contracts," in *Proceedings of the twelfth language resources and evaluation conference*, 2020, pp. 1235–1241.

[141] A. Zhang, Z. C. Lipton, M. Li, and A. J. Smola, "Dive into deep learning," *arXiv preprint arXiv:2106.11342*, 2021.

[142] T. Schick and H. Schütze, "It's not just size that matters: Small language models are also few-shot learners," *arXiv preprint arXiv:2009.07118*, 2020.

[143] H. Yang, Y. Zhang, J. Xu, H. Lu, P. A. Heng, and W. Lam, "Unveiling the generalization power of fine-tuned large language models," 2024.

[144] K. Shen and M. Kejriwal, "An experimental study measuring the generalization of fine-tuned language representation models across commonsense reasoning benchmarks," *Expert Systems*, vol. 40, no. 5, p. e13243, 2023.

[145] K. VM, H. Warrier, Y. Gupta *et al.*, "Fine tuning llm for enterprise: Practical guidelines and recommendations," *arXiv preprint arXiv:2404.10779*, 2024.

[146] Y. Lee, S. Jeong, and J. Kim, "Improving llm classification of logical errors by integrating error relationship into prompts," in *International Conference on Intelligent Tutoring Systems*. Springer, 2024, pp. 91–103.

[147] J. Yang, Z. Wang, Y. Lin, and Z. Zhao, "Global data constraints: Ethical and effectiveness challenges in large language model," *arXiv preprint arXiv:2406.11214*, 2024.

[148] L. Weidinger, J. Uesato, M. Rauh, C. Griffin, P.-S. Huang, J. Mellor, A. Glaese, M. Cheng, B. Balle, A. Kasirzadeh *et al.*, "Taxonomy of risks posed by language models," in *Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency*, 2022, pp. 214–229.

[149] D. Narayanan, M. Shoeybi, J. Casper, P. LeGresley, M. Patwary, V. Korthikanti, D. Vainbrand, P. Kashinkunti, J. Bernauer, B. Catanzaro *et al.*, "Efficient large-scale language model training on gpu clusters using megatron-lm," in *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, 2021, pp. 1–15.

[150] I. O. Gallegos, R. A. Rossi, J. Barrow, M. M. Tanjim, S. Kim, F. Dernoncourt, T. Yu, R. Zhang, and N. K. Ahmed, "Bias and fairness

in large language models: A survey," *Computational Linguistics*, pp. 1–79, 2024.
[151] Y. Yu, Y. Zhuang, J. Zhang, Y. Meng, A. J. Ratner, R. Krishna, J. Shen, and C. Zhang, "Large language model as attributed training data generator: A tale of diversity and bias," *Advances in Neural Information Processing Systems*, vol. 36, 2024.
[152] H. Kotek, R. Dockum, and D. Sun, "Gender bias and stereotypes in large language models," in *Proceedings of the ACM collective intelligence conference*, 2023, pp. 12–24.