applied sciences

MDPI

Article

A Generative Artificial Intelligence Using Multilingual Large Language Models for ChatGPT Applications

Nguyen Trung Tuan [1], Philip Moore [2], Dat Ha Vu Thanh [3] and Hai Van Pham [3,]*

[1] School of Information Technology and Digital Economics, National Economics University, 207 Giai Phong Road, Hai Ba Trung District, Hanoi 10000, Vietnam; tuannt@neu.edu.vn
[2] School of Information Science and Engineering, Lanzhou University, Feiyun Building, 222 Tianshui South Road, Chengguan Qu, Lanzhou 730030, China; ptmbcu@gmail.com
[3] School of Information and Communication Technology, Hanoi University of Science and Technology, 1 Dai Co Viet, Le Dai Hanh, Hai Ba Trung District, Hanoi 10000, Vietnam; dat.hvthanh@gmail.com
* Correspondence: haipv@soict.hust.edu.vn

Abstract: ChatGPT plays significant roles in the third decade of the 21st Century. Smart cities applications can be integrated with ChatGPT in various fields. This research proposes an approach for developing large language models using generative artificial intelligence models suitable for small- and medium-sized enterprises with limited hardware resources. There are many generative AI systems in operation and in development. However, the technological, human, and financial resources required to develop generative AI systems are impractical for small- and medium-sized enterprises. In this study, we present a proposed approach to reduce training time and computational cost that is designed to automate question–response interactions for specific domains in smart cities. The proposed model utilises the BLOOM approach as its backbone for using generative AI to maximum the effectiveness of small- and medium-sized enterprises. We have conducted a set of experiments on several datasets associated with specific domains to validate the effectiveness of the proposed model. Experiments using datasets for the English and Vietnamese languages have been combined with model training using low-rank adaptation to reduce training time and computational cost. In comparative experimental testing, the proposed model outperformed the 'Phoenix' multilingual chatbot model by achieving a 92% performance compared to 'ChatGPT' for the English benchmark.

Keywords: generative AI; language comprehension; multilingual language models; large language models; support systems; technological determinism; chatbot; ChatGPT

check for updates

Citation: Tuan, N.T.; Moore, P.; Thanh, D.H.V.; Pham, H.V. A Generative Artificial Intelligence Using Multilingual Large Language Models for ChatGPT Applications. *Appl. Sci.* **2024**, *14*, 3036. https://doi.org/10.3390/app14073036

Academic Editor: João M. F. Rodrigues

Received: 5 March 2024
Revised: 26 March 2024
Accepted: 2 April 2024
Published: 4 April 2024

1. Introduction

Currently, ChatGPT can integrate applications for real-time tracking of many areas of a smart city such as traffic management, energy management [1], environmental monitoring, healthcare [2], and emergency response. ChatGPT can integrate applications of smart cities in real time with effectiveness and efficiency. Generative artificial intelligence (hereafter termed GenAI) is a rapidly developing technology that has gained significant traction, which has arguably been driven by the release of ChatGPT by OpenAI (OpenAI: https://openai.com/ (accessed on 10 December 2023)). In practice, GenAI is an important example of a disruptive innovation (DI) where novel technologies can result in technological determinism (TD) [3]. GenAI has been the subject of many ethical, societal, technological, and practical risks expressed by a diverse range of stakeholders as discussed in Section 2.

GenAI models have gained traction in multiple domains of interest, and the influence exerted by GenAI is clear as shown by research studies published in the literature. The design and development of GenAI models is highly resource intensive, requiring a large investment in financial, technological, computational, social analysis, and human resources [4–6]. Additionally, data corpus-assisted data-driven learning remains a critical element [7] and there is a need for a suitable large language model (LLM) [8].

While there are cloud-based options (a range of models and plans) available from GenAI developers (for example see the OpenAI ChatGPT plans and OpenAI pricing: https://openai.com/pricing (accessed on 10 December 2023)), GenAI models are generally domain-specific and the resource intensive nature of such models along with the related LLMs limits the ability of small–medium enterprises (SMEs) to adopt an appropriate GenAI model. Identifying a resolution to this problem is important as chatbots can offer significant organisational and commercial benefits for organisations of all types [9–11].

Recently, many applications for large language models (LLMs) have considered reasoning mechanisms in LLMs for reasoning and making decisions using ChatGPT. The state-of-art characteristics of GPT models can be combine with the language understanding capabilities applied to many application domains [12,13]. An approach of using reasoning techniques has been performed by using distinct datasets of GPT-3.5, GPT-4, and BARD models [14]. All the models mentioned above deal with high costs and GPU hardware requirements, so medium-size companies or organisations in cities lack the high-cost hardware resources needed to run ChatGPT. Compared to these studies, LLMs require a large infrastructure or system for reasoning and answering questions [12]. Our study can develop LLMs and ChatGPT deployed on medium-size GPU servers which are suitable for SME infrastructure. The latest versions of ChatGPT models and Google's BARD [13,14] have been used in the evaluation of reasoning domains such as deductive, inductive, and question-answering tasks. Both ChatGPT and BARD may sometimes produce plausible but incorrect outcomes and inaccurate interactions in large domains. We have investigated BLOOM [15] for improving the accuracy in question-answering challenges and the model's performance in dealing with a variety of application domains that suitably consider a medium-size infrastructure.

In this paper, we present our proposed method (termed *Expert-B*) which utilises open-source program code based on BLOOM [15] as its backbone. BLOOM is an open-access language model trained on the ROOTS language corpus [16] instruction dataset (hereafter termed 'ROOTS') introduced in Section 3.4 and Figure 5. The implementation employs 'LoRA: Low-Rank Adaptation of Large Language Models' [17] with 'DeepSpeed' (For 'DeepSpeed' see: https://www.microsoft.com/en-us/research/project/deepspeed/ (accessed on 10 December 2023)); we provide a detailed discussion on the implementation process in Section 3. Our contributions include the following:

1. This research contributes to the discussion of how GenAI can be leveraged to maximum effect for SMEs.
2. The proposed *Expert-B* GenAI model provides an effective and flexible basis for bespoke development and implementation of a GenAI-driven chatbot.
3. The creation of a chatbot that can adapt to multiple languages; in this study, our focus is on Vietnamese and English.
4. The adoption of open-source program code trained using the *Expert-B* model contributes to a reduction in training time and computational cost.
5. The creation of bilingual instruction datasets for English and Vietnamese when combined with the *Expert-B* model trained using 'Low-Rank Adaptation' (LoRA) and 'DeepSpeed'.

The motivation for this paper is to optimise a pipeline for the training process in computational resources for large language models (LLM) as follows: (1) LoRA is to reduce the number of parameters used during training while maintaining the model's performance at a satisfactory level; (2) DeepSpeed is applied to the training process for distributed training, thus alleviating the training pressure on GPU VRAM. Additionally, a synthetic dataset has been created using the expert-prompting technique, thus creating high-quality datasets across various domains from large language models such as GPT-3.5, GPT-4, etc. This innovative method enables small- and medium-sized enterprises (SMEs) to construct their own large language models for chatbot technology. This can be conducted by organisations in heterogeneous domains in smart cities to increase domain knowledge in a specific topic or domain, such as healthcare, business, customer service, or finance. It can be accomplished by fine-tuning the proposed model using datasets including the following:

- In this study, we consider the development of a 'bespoke' domain-specific GenAI-driven chatbot for SMEs designed to automate question–response interactions.
- This research aims to address the problem by creating a GenAI model for a chatbot complete with a LLM [7] that can adapt to multiple languages (in this research the focus is on Vietnamese and English) for use in GenAI models suitable for resource limited SMEs.
- Turning to potential language difficulties, a multilingual chatbot can cater to the needs of customers who speak different languages [8].
- The proprietary GenAI models are highly resource intensive; by developing an approach that uses public resources combined with low computational costs, SMEs can also take advantage of GenAI chatbot technology.

In experimental testing, the proposed *Expert-B* model achieved a significant performance improvement, and in a comparative analysis our proposed model outperformed the 'Phoenix' multilingual chatbot model by achieving a 92% performance compared to ChatGPT as the English benchmark.

The remainder of this paper is structured as follows: The state of art and related research is considered in Section 2. The proposed *Expert-B* model is introduced in Section 3 with experimental testing introduced in Section 4. The results and an analysis are set out in Section 5. Section 6 presents a discussion along with open research questions and directions for future research. The paper closes with concluding observations in Section 7.

2. Related Research

In this section, we consider GenAI along with an overview of ChatGPT and LLM.

2.1. Generative Artificial Intelligence and Chatbots

Recently, GenAI has played significant roles in many diverse domains in smart cities. There is a large and growing body of published GenAI research applied in domains including the supply chain [18], science and healthcare [19], and education and academic integrity [20–23].

The term GenAI identifies methods capable of generating text, images, or other media, using generative models. GenAI models have been developed by multinational companies (e.g., Microsoft, Google, and Baidu) along with many smaller developers also creating GenAI models. There is no doubt that GenAI models have generated significant traction, particularly in knowledge-based roles, often replacing human respondents in on-line question–response interactions. In practice, automation may eliminate some occupations entirely (over the next decade) and it might affect most roles to various degree dependent on the type of occupation (Mckinsey: https://www.mckinsey.com/capabilities/mckinsey-digital/our-insights/where-machines-could-replace-humans-and-where-they-cant-yet (accessed on 10 December 2023)). The traction generated by GenAI has been arguably driven by OpenAI and ChatGPT (OpenAI and ChatGPT: https://openai.com/ (accessed on 10 December 2023)) and similar models from other developers.

Sætra in [24] poses the question "Generative AI: Here to stay, but for good?" and observed that GenAI has "taken the world by storm" with GenAI models being adopted by organisations of all types [9–11], and that it has applications in many diverse domains including: *culture*, *literature*, *software engineering*, *product design*, *healthcare*, *finance*, *gaming*, *sales and marketing*, and *fashion* [18,20].

GenAI models provide important potential opportunities; however, there are potential risks [20]. Sætra in [24] considers some key questions on *macro*, *meso*, and *macro* levels. The three levels represent potential dangers and challenges for GenAI and are modelled in Figure 1.

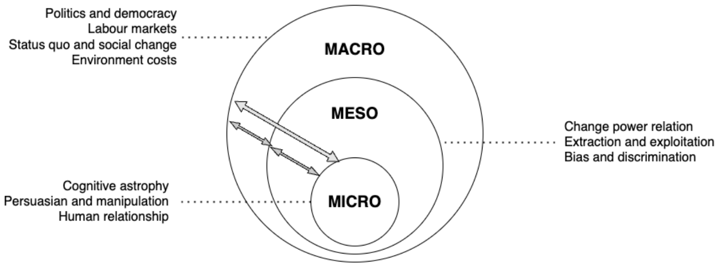


Figure 1. A model of *micro*, *meso*, and *macro* level dangers for GenAI (source: [24]).

Evans et al. in [22] have considered the potential benefits of GenAI (with a focus on ChatGPT) in the domains of healthcare, education, and business, and they have identified the need to consider risks including ethical considerations and the need for human oversight. When viewed from a strategic perspective, the evaluation of disruptive technologies and TD generally requires an analysis predicated on identifying the 'Strengths', 'Weaknesses', 'Opportunities', and 'Threats' (SWOT) an an analysis. Albool in [25] considered GenAI and ChatGPT and carried out a SWOT analysis with the issues affecting the stakeholders of ChatGPT in education and provided recommendations before concluding that: "... if ChatGPT is to fulfil its potential, there must be a clear understanding of the various issues involved ...".

In considering the opportunity/risk profile for GenAI models, GenAI may be viewed as a DI [26,27] with effects similar to those discussed in research addressing TD [28–30], while GenAI offers many opportunities, there are also concerns around the use of GenAI models including *cyberchrime*, *fake news*, and *deepfakes* [31], all of which can be used to deceive or manipulate people. Notwithstanding the ethical and practical challenges, the uptake of GenAI has demonstrated its potential. Saetra argues in [24] that "there is no longer much point in discussing whether generative AI will be influential (and) the discussion is now centred in how influential it will be, and what potential harms arise when we use AI to generate text and other forms of content".

The central problem with DI (or novel technologies) lies in their disruptive nature as discussed in [3,26,27]. DI can be compared to sustaining innovation (SI), which mainly "Improves or evolves existing value creation models and markets" [32]. DI is a term originally conceived to refer to any technology(s) with the potential "disrupt traditional value creation models and markets" [32]. The issue is that, over time, the concept (introduced by Clayton Christensen in the 1990s) has been generally applied to describe almost every type of novel innovation [32]. However, Markides in [33] has identified the domain-specific nature of DI and questioned Christensen's 1997 DI theory because over time the theory has been wrongly applied to many domains.

Despite of the traction generated by GenAI and, while recognising the potential benefits and opportunities, there are still significant challenges (worries or threats) when viewed from a societal and technological perspective [34]. Significant threats identified by academics in terms of ethical and practical concerns have been identified [24]. Research has investigated such opportunities and threats in terms of DI and TD [3].

We may conclude from this brief analysis that identifying potential effects impacting all stakeholders are essential and, moreover, that there is a delay in understanding the socio-technological affects following implementation of DI. Subsequent research leads to a better understanding of such affects with research studies informing future developments [35].

2.2. Large Language Models and Chatbots

A chatbot such as ChatGPT (Chat Generative Pre-Trained Transformer) is a software application (generally on-line) that typically utilises GenAI and an LLM. When considering chatbots in e-commerce, there are two approaches to interactions: (a) using *formal* language and (b) using *informal* (possibly colloquial language which is generally nationality and ethnically specific [36]) language used in ordinary or familiar conversation [37].

As discussed in [37], the results derived "through the mediating role of 'parasocial' interaction" [37] show that when chatbots adopt an informal language style customers' reactions are positive with increased use and a positive brand awareness. Parasocial interaction (PSI) refers to a psychological relationship experienced by an audience in their mediated encounters with performers in the mass media and on-line platforms [37–41].

The goal of a chatbot is to maintain a conversation with a user in natural language and simulate how a human would behave as a conversational partner [34]. ChatGPT is a transformer-based deep neural network-enabled model capable of accepting natural language prompts as input based on LLMs [42]. There is an interesting parallel between the concept of a chatbot using GenAI and LLMs and the Turing test (devised in 1950) [34]. However, notwithstanding that the Turing test is not representative of AI, in the opinion of many in human–chatbot interactions "ChatGPT not only passed but obliterated the Turing test" [34].

While the core function of a chatbot is to mimic a human conversationalist, GenAI-driven chatbots have demonstrated the capability to: (a) write and debug computer programs, (b) compose music, stories, and drama scripts, (c) draft student essays and assignments, (d) answer test questions (on occasion above the level of a human), (e) generate business ideas, (f) write poetry and song lyrics, (g) translate and summarise text, (h) emulate a Linux system, (i) simulate entire chat rooms, (j) play games (like tic-tac-toe), or (k) simulate an ATM. The scope, and the related potential threats, for GenAI and chatbots is clear [20,22,23,34,43–45].

Turning to the limitations and issues in GenAI models and ChatGPT:

1. There is a recognition by OpenAI (OpenAI: https://openai.com/ (accessed on 10 December 2023)) that ChatGPT "sometimes writes plausible sounding but incorrect or nonsensical answers"; a feature common to LLMs often termed "hallucination". To address (or at least mitigate) hallucination, ChatGPT operates a reward model which is predicated on "human oversight". However, the reward model can be over-optimised and thus hinder performance, which is an example of an optimisation pathology known as Goodhart's law [46].
2. ChatGPT has limited knowledge of events that occurred after September 2021 resulting in significant errors. Moreover, as discussed in Section 5.4, errors (e.g., inaccurate translation) can be the result of semantic misunderstanding or the language corpus.
3. In training ChatGPT, human reviewers preferred longer answers, regardless of actual comprehension or factual content. Additionally, training data also suffers from algorithmic bias, which may be revealed when ChatGPT responds to prompts including descriptors of people. In one instance, ChatGPT generated a rap indicating that women and scientists of colour were inferior to white male scientists.
4. There is an issue with plagiarism by GenAI and therefore by ChatGPT. It is necessary to address this problem which is a recognised problem in the education field [20,30].
5. In an attempt to mitigate plagiarism, it has been reported that OpenAI (for ChatGPT) has investigated using a digital watermark for text generation systems to combat "bad actors using their services for academic plagiarism or spam".

The future for GenAI models presents many opportunities and threats which may be identified using a SWOT analysis. For example, Microsoft announced an experimental framework and gave a rudimentary demonstration of how ChatGPT could be used to control robotics with intuitive open-ended natural language commands.

We have considered the positive and negative aspects of GenAI and chatbots with a focus on ChatGPT. We have noted the disruptive nature of GenAI and the need for research to understand the socio-technological implications of technology. GenAI is 'out of the bag' and it may be viewed as a 'Pandoras box'—a mythical box which once opened releases "all the troubles of the world, never to be recaptured".

3. The Proposed *Expert-B* Model

In this section, we introduce our *Expert-B* model together with the inputs, outputs, and methods. The proposed model using BLOOM aims to create a multilingual chatbot that can generate relevant responses, which consists of components such as input IDs, attention marks, and label IDs. An application interface designed as Web GUI allows the proposed model to chat in real-time with a domain. The pre-processed instruction data is transmitted through a network that integrates the BLOOM model and an adapter. Subsequently, the model combined with the adapter is deployed on the Web GUI. The input–output process is described, followed by an introduction to 'BLOOM' [15], the dataset, the training objectives, the instruction dataset, LRA, DeepSpeed, and Phoenix. The section then closes with conclusions. The evaluation and testing is discussed in Section 4 with the results set out in Section 5. An overview of the proposed system architecture showing the data processing pipeline, model architecture, training process, and deployment is shown in the conceptual model in Figure 2.

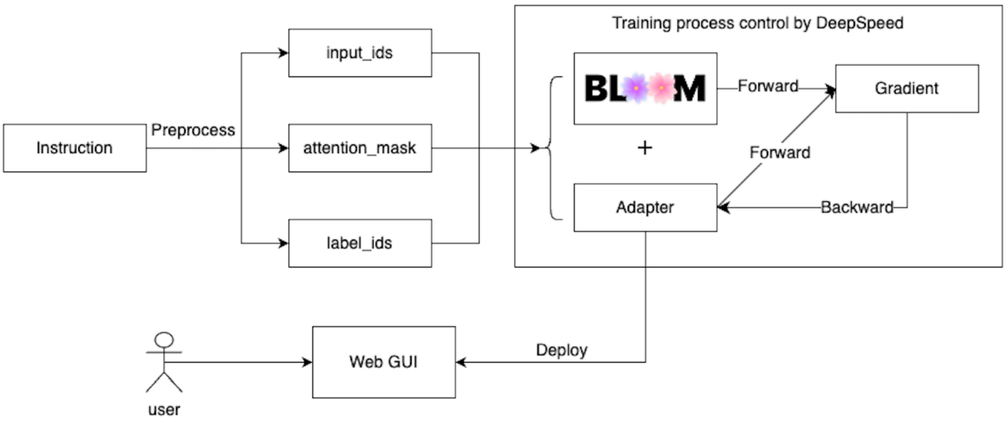


Figure 2. System architecture overview with data processing pipeline, model architecture, training process, and deployment.

3.1. Overview

We have noted ChatGPT's success in the conversational AI domain. However, the limitations of the models discussed in Section 2 include a heavy reliance on substantial computational resources for maintenance. To address this issue, Stanford introduced an approach which utilises a publicly accessible backbone called *LLaMA* [47] and fine-tunes it on their public instruction following a dataset named *Alpaca* [48]. This approach has arguably become the optimal method for achieving ChatGPT-like performance using publicly available resources, specifically for the English language.

In the proposed *Expert-B* method, we aim to further enhance the capabilities of Chat-GPT, not just for English as in the case of Alpaca [48], but for multiple languages by using fine-tuned 'BLOOM' [15]. To accomplish this, a comprehensive instruction dataset that encompasses a wide range of tasks is leveraged using Alpaca [47] as a seed. By harnessing the adaptability of BLOOM [15] for both English and Vietnamese, the aim is the development of a multilingual chatbot capable of generating contextually relevant responses in both languages. Moreover, to optimise the training process and efficiently utilise GPU memory, the LoRA [17] and LRA [49] methods are used with the DeepSpeed ZeRO-offload [50] method. These techniques assist in managing memory constraints while enabling smooth training of the model and maintaining its performance.

3.2. Input–Output

The methodology for *Expert-B* involves fine-tuning the BLOOM [15] model by using a specific instruction dataset with the aim of creating a multilingual chatbot capable of following instructions and generating contextually relevant responses. The input for the model consists of instruction prompts which can be in the form of commands or queries that the chatbot must understand and provide appropriate responses as specified in Equation (1):

$$\mathcal{D} = (x_1,\ y_1)\ (x_2, y_2)\ \ldots\ (x_N,\ y_N) \tag{1}$$

where dataset D contains N examples and for example i, N is the number of examples of *instruction–output* pairs. x_n is the n^{th} instruction, and y_n is the output corresponding to the n^{th} instruction.

Figure 3 shows an example for *instruction–output* pairs in the training dataset. It is important to note that the input provided and output examples are generalised illustrations and the actual instructions and corresponding outputs may vary depending on the specific instruction dataset and fine-tuning process employed. The methodology focuses on training the 'BLOOM' [15] model to understand and follow instructions, thus enabling it to generate appropriate and informative responses based on the given prompts.

Instruction: `Describe the structure of an atom.`

Ouput: `The structure of an atom can be thought of as comprising of three main components: a central nucleus containing protons and neutrons, and electrons that orbit around it in distinct energy levels or shells. The nucleus is the dense, positively charged center of the atom, while the electrons are negatively charged particles that orbit around the nucleus at varying distances. The number of protons present in the nucleus determines the atomic number of the atom, which in turn defines its unique chemical properties and place in the periodic table of elements. The number of neutrons present in the nucleus can vary, giving rise to isotopes of the atom. The electrons occupy distinct energy levels or shells around the nucleus, with those closest to the nucleus having lower energy levels than those further away. These energy levels determine the reactivity and chemical bonding behaviour of the atom. The overall structure of an atom is governed by the electromagnetic force that holds the negatively charged electrons in orbit around the positively charged nucleus, and the strong nuclear force that binds the protons and neutrons together in the nucleus.`

Figure 3. A sample taken from the training dataset.

3.3. BLOOM

In this sub-section, we introduce 'BLOOM' and consider the architecture as shown in Figure 4 along with the dataset (see Figure 5).

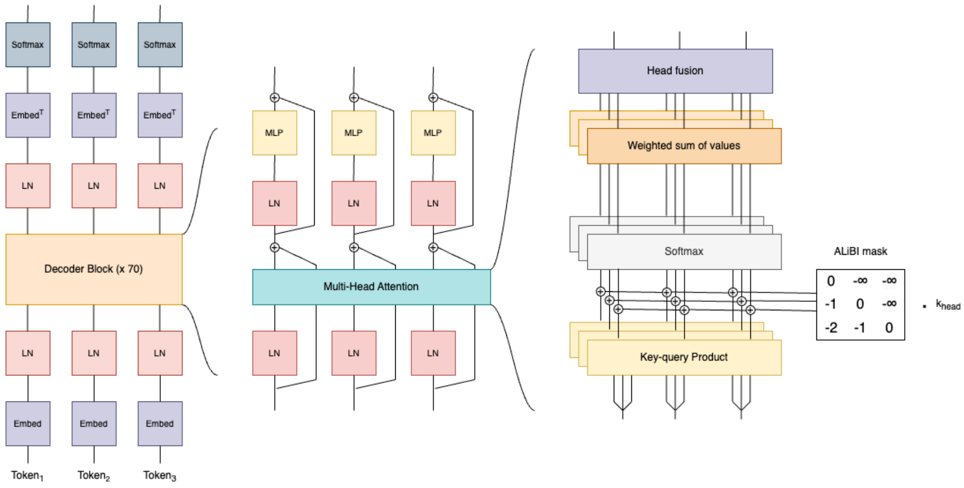
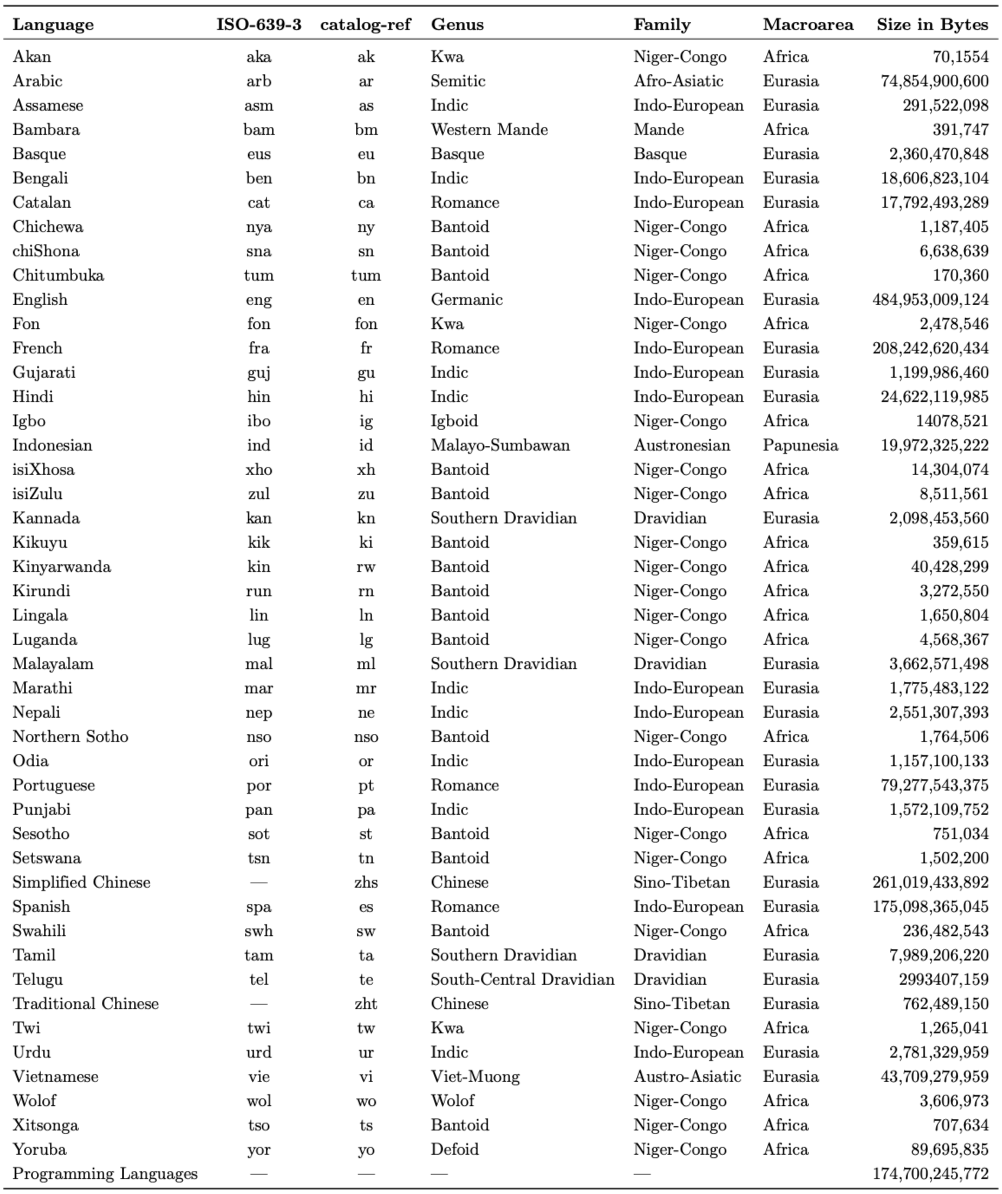


Figure 4. The architecture of BLOOM undergoes a slight modification compared to the original transformers architecture.



Language	ISO-639-3	catalog-ref	Genus	Family	Macroarea	Size in Bytes
Akan	aka	ak	Kwa	Niger-Congo	Africa	70,1554
Arabic	arb	ar	Semitic	Afro-Asiatic	Eurasia	74,854,900,600
Assamese	asm	as	Indic	Indo-European	Eurasia	291,522,098
Bambara	bam	bm	Western Mande	Mande	Africa	391,747
Basque	eus	eu	Basque	Basque	Eurasia	2,360,470,848
Bengali	ben	bn	Indic	Indo-European	Eurasia	18,606,823,104
Catalan	cat	ca	Romance	Indo-European	Eurasia	17,792,493,289
Chichewa	nya	ny	Bantoid	Niger-Congo	Africa	1,187,405
chiShona	sna	sn	Bantoid	Niger-Congo	Africa	6,638,639
Chitumbuka	tum	tum	Bantoid	Niger-Congo	Africa	170,360
English	eng	en	Germanic	Indo-European	Eurasia	484,953,009,124
Fon	fon	fon	Kwa	Niger-Congo	Africa	2,478,546
French	fra	fr	Romance	Indo-European	Eurasia	208,242,620,434
Gujarati	guj	gu	Indic	Indo-European	Eurasia	1,199,986,460
Hindi	hin	hi	Indic	Indo-European	Eurasia	24,622,119,985
Igbo	ibo	ig	Igboid	Niger-Congo	Africa	14078,521
Indonesian	ind	id	Malayo-Sumbawan	Austronesian	Papunesia	19,972,325,222
isiXhosa	xho	xh	Bantoid	Niger-Congo	Africa	14,304,074
isiZulu	zul	zu	Bantoid	Niger-Congo	Africa	8,511,561
Kannada	kan	kn	Southern Dravidian	Dravidian	Eurasia	2,098,453,560
Kikuyu	kik	ki	Bantoid	Niger-Congo	Africa	359,615
Kinyarwanda	kin	rw	Bantoid	Niger-Congo	Africa	40,428,299
Kirundi	run	rn	Bantoid	Niger-Congo	Africa	3,272,550
Lingala	lin	ln	Bantoid	Niger-Congo	Africa	1,650,804
Luganda	lug	lg	Bantoid	Niger-Congo	Africa	4,568,367
Malayalam	mal	ml	Southern Dravidian	Dravidian	Eurasia	3,662,571,498
Marathi	mar	mr	Indic	Indo-European	Eurasia	1,775,483,122
Nepali	nep	ne	Indic	Indo-European	Eurasia	2,551,307,393
Northern Sotho	nso	nso	Bantoid	Niger-Congo	Africa	1,764,506
Odia	ori	or	Indic	Indo-European	Eurasia	1,157,100,133
Portuguese	por	pt	Romance	Indo-European	Eurasia	79,277,543,375
Punjabi	pan	pa	Indic	Indo-European	Eurasia	1,572,109,752
Sesotho	sot	st	Bantoid	Niger-Congo	Africa	751,034
Setswana	tsn	tn	Bantoid	Niger-Congo	Africa	1,502,200
Simplified Chinese	—	zhs	Chinese	Sino-Tibetan	Eurasia	261,019,433,892
Spanish	spa	es	Romance	Indo-European	Eurasia	175,098,365,045
Swahili	swh	sw	Bantoid	Niger-Congo	Africa	236,482,543
Tamil	tam	ta	Southern Dravidian	Dravidian	Eurasia	7,989,206,220
Telugu	tel	te	South-Central Dravidian	Dravidian	Eurasia	2993407,159
Traditional Chinese	—	zht	Chinese	Sino-Tibetan	Eurasia	762,489,150
Twi	twi	tw	Kwa	Niger-Congo	Africa	1,265,041
Urdu	urd	ur	Indic	Indo-European	Eurasia	2,781,329,959
Vietnamese	vie	vi	Viet-Muong	Austro-Asiatic	Eurasia	43,709,279,959
Wolof	wol	wo	Wolof	Niger-Congo	Africa	3,606,973
Xitsonga	tso	ts	Bantoid	Niger-Congo	Africa	707,634
Yoruba	yor	yo	Defoid	Niger-Congo	Africa	89,695,835
Programming Languages	—	—	—	—		174,700,245,772

Figure 5. The ROOT statistics.

3.3.1. Architecture

Compared to the original Transformer Decoder Blocks, BLOOM [15] has a number of modifications: 'ALiBi Positional Embeddings' [51] and 'Embedding LayerNorm' [15]. As an alternative to incorporating positional information into the embedding layer, the 'ALiBi' approach implements a direct attenuation of attention scores based on the relative distance between the keys and queries. The motivation behind the 'ALiBi' approach was initially to enable extrapolation for longer sequences [51]. However, it was observed that this approach also facilitated smoother training and improved downstream performance, even when operating at the original sequence length. 'ALiBi' surpassed the performance of learned embedding methods in terms of overall effectiveness. The 'BLOOM' architecture is modelled in Figure 4.

3.3.2. Positional Embeddings

In the original transformer architecture, positional embeddings are added to the word embeddings at the input layer. This means that the positional information is incorporated into the attention mechanism from the very beginning, as the character passes through the embeddings layer before reaching the scaled-dot product attention.

For an input sequence of length L, the attention sublayer computes attention scores for the ith query $q_i \in R^{1xd}$, $(1 \leq i \leq L)$ in each head, given the first i keys $K \in R^{ixd}$, where d is the head dimension as expressed by Equation (2)

$$Sofmax(q_i K^T) \tag{2}$$

However, in the case of the 'ALiBi' approach, there is no addition of positional embeddings at any point in the network. The only modification made is the inclusion of a static, non-learned bias after the query-key dot product operation [51]. The process is as given in Equation (3):

$$Softmax(q_i K^T \ + \ m \cdot [-(i-1) \ \ldots \ -2, -1, 0]) \tag{3}$$

where scalar (m) is a head-specific slope fixed before training.

After the embedding layer, an additional layer of LayerNorm (LN) is introduced. This change has been observed contributing to the stability of model training. Alongside this enhancement, the k_{head} slope parameters for ALiBi are taken as $2^{\frac{-8i}{n}}$ with n the number of heads and $i \in 1, 2, ..., n$.

3.3.3. Embedding LayerNorm

Using 'Embedding LayerNorm' to enhance the training stability of BLOOM [15], an additional layer normalisation is applied immediately after the embedding layer. This modification has proven to be highly beneficial, as it significantly improves the stability of the training process. By incorporating this extra layer normalisation step after the initial embedding layer, potential instabilities during training are effectively mitigated.

3.4. Instruction Dataset

'BLOOM' [15] was trained on the ROOTS instruction dataset [16]. ROOTS is a composite multilingual dataset consisting of a collection of 498 Hugging Face datasets in 1.61 terabytes of text spanning 46 natural languages and 13 programming languages; a detailed itemised list of every language along with its linguistic genus, family, and macro area is presented in Figure 5.

3.5. Low-Rank Adaptation

Aghajanyan et al. in [49] shows that pretrained language models have a low intrinsic dimensionality but can still learn efficiently despite a random projection to a smaller subspace. According to the hypothesis, for a pre-trained weight matrix $W_0 \in \mathbb{R}^{d \times k}$, the smaller subspace parameter is δW, which is created by multiplying two types of matrix with much smaller dimension compared to pre-trained weight: A—compression matrix and B—decompression matrix. We constrain its update by representing the latter with a low-rank decomposition in Equation (4). Figure 6 models the relationship (s) and the process.

$$W_0 \ + \ \delta W \ = \ W_0 \ + \ BA \tag{4}$$

where $\left(B \in \mathbb{R}^{d \times r}\right)$, $\left(A \in \mathbb{R}^{r \times k}\right)$, and $(r \ll \min(d,k)))$.

The pretrained weight, denoted as W_0, remains frozen during the training process, and AB denotes the combination of compression and decompression matrices. Two matrices, W_0 and AB, both receiving the input X, operate in conjunction. Subsequently, the hidden state of X after traversing the network is the summation of the results obtained from the two matrices, $W_0 x$ and ABx.

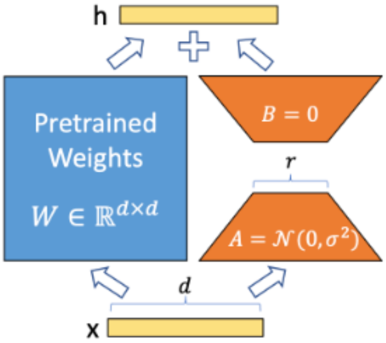


Figure 6. The operational mechanism of LoRA is delineated through the flow depicted in the image.

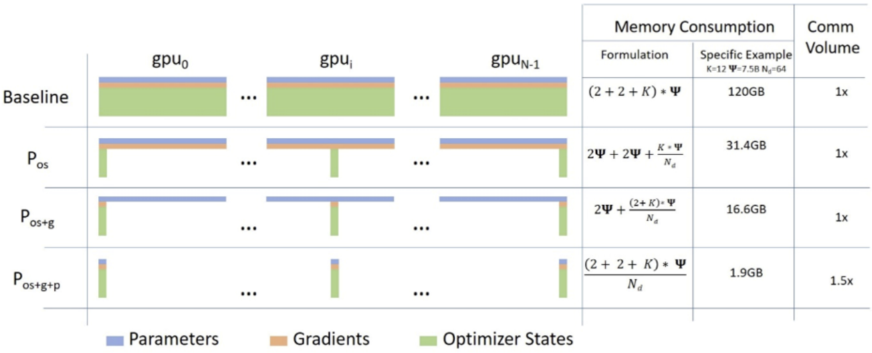
During training, (W_0) is frozen and does not receive gradient updates, while (A) and (B) contain trainable parameters. Assume that $x \in R^k$ is the input to the model, both (W_0) and $(\delta W \;=\; BA)$ are multiplied with the same input, and their respective output vectors are summed coordinate-wise. The hidden state of x-h is found through the model and is now calculated in Equation (5):

$$h \;=\; W_0x \;+\; \delta Wx \;=\; W_0x \;+\; BAx \tag{5}$$

3.6. DeepSpeed

'DeepSpeed' [50] is a deep learning optimisation library developed by Microsoft Research providing advanced techniques to improve the performance and efficiency of deep learning models; the focus lies in addressing challenges related to large-scale model training and memory optimisation. The most popular distributed training libraries (e.g., *torchrun* or *accelerate*) allow for loading data parallelism or model parallelism.

The Zero Redundancy Optimiser (ZeRO) [50] (see Figure 7) is a collection of memory optimisation techniques designed for distributed deep learning on a large scale. ZeRO enables the use of larger models without code re-factoring while maintaining high efficiency. ZeRO achieves this by eliminating memory redundancies inherent in data parallelism and minimizing communication overhead. Instead of replicating the model states (optimiser states, gradients, and parameters) across data-parallel processes, ZeRO partitions them, effectively reducing memory redundancy. This approach improves memory efficiency compared to traditional data parallelism, while preserving computational granularity and communication efficiency.



	Memory Consumption		Comm Volume
	Formulation	Specific Example K=12 Ψ=7.5B N_d=64	
Baseline	$(2+2+K)*\Psi$	120GB	1x
P_{os}	$2\Psi + 2\Psi + \frac{K*\Psi}{N_d}$	31.4GB	1x
P_{os+g}	$2\Psi + \frac{(2+K)*\Psi}{N_d}$	16.6GB	1x
P_{os+g+p}	$\frac{(2+\ 2+\ K)*\ \Psi}{N_d}$	1.9GB	1.5x

Figure 7. ZeRO-offload compared to baseline approach.

While the baseline approach computes parameters, gradients, and optimiser states (p, g, and os) across all GPUs, this consumes a significant amount of GPU memory. ZeRO enables the partitioning of these components across multiple GPUs, which leads to a noticeable reduction in memory consumption when training large models (as explicitly illustrated in in Figure 7). Moreover, in addition to data parallelism, ZeRO offers the flexibility to partition components during training based on ZeRO offloading levels as follows:

- **State 1** (P_{os}) (Optimiser States);
- **State 2** (P_{os+g}) (Optimiser States + Gradients);
- **State 3** P_{os+g+p} (Optimiser States + Gradients + Parameters).

The flexibility to partition components during training provides a basis upon which the optimum level of offloading can be selected. If desired, while resulting in slower training, users can offload the optimiser state or parameters or both to free up GPU resources. During operation, implementing 'DeepSpeed' saves considerable time as it eliminates the need to modify the training code because users only need to add a configuration file that contains important settings such as data type, batch size, ZeRO offload state, etc. 'Deepspeed' handles the remaining configuration operations and the optimisation process, thus simplifying the overall workflow.

3.7. Phoenix

As discussed in this paper, *Expert-B* is compared to the Phoenix [42] and, while there are common features, *Expert-B* generally improves on the Phoenix model. Phoenix was created by fine-tuning 'BLOOM' with the following datasets:

- **Multilingual Instruction:** using the Alpaca instruction dataset as a seed, it was translated into various languages and then used with 'GPT-3.5-turbo' API to generate answers in over 40 different languages.
- **User-Centred Instruction:** various samples in the form of role, instruction, and input were generated from multiple seeds, which were then passed through 'GPT-3.5-turbo' API to generate answers for each sample.
- **Conversation:** this dataset consists of conversation histories shared on the internet between people and ChatGPT, and each sample can contain multiple turns of consecutive conversation.

In summary, the Phoenix dataset consists of 465 k samples and 939 k conversation turns. As discussed in this paper, *Expert-B* was trained on only 104 k samples, equivalent to 104 k conversation turns but the Phoenix dataset is approximately nine times larger, resulting in reductions in training time and computational overhead for the *Expert-B* model.

4. Experimental Testing

In this section, we introduce the evaluation and testing regime, the results derived from experimental testing and a case study based on the Vietnamese rate set out in Section 5.

4.1. Training Objectives

In their iterative releases, the "BigScience" workshop team [15] introduced multiple versions of 'BLOOM' [15] that was implemented using a range of parameters along with clearly specified hyperparameters and configurations for each version, as seen in Table 1.

Given the project's initial aim of developing a chatbot model requiring limited hardware resources, the 'BLOOM' model with a 7 billion parameter set (BLOOM-7B1) has been identified as the most suitable model. Related studies have also identified the BLOOM-7B1 model as their backbone including LLaMA-7B for Alpaca [48] and BLOOM-7B1 for the Phoenix model. All experiments conducted in this study use the BLOOM-7B1 model as the backbone. Recall that an autoregressive language model defines a conditional distribution, where the probability of the ith word—x_i—depends on the contextual meaning or the words $x_{1:i-1}$ as shown in Equation (6):

$$p(x_i \mid x_{1:i-1}) \tag{6}$$

Equation (6) uses the following steps:

- **State 1** Map $(x_{1:i-1})$ to contextual embeddings $\phi(x_{1:i-1})$;
- **Step 2** Apply an embedding matrix $E \in \mathbb{R}^{V \times d}$ to obtain scores for each token $E\phi(x_{1:i-1})_{i-1}$;
- **Step 3** Exponentiate and normalize it to produce the distribution over x_i.

Steps 1–3 are succinctly shown as Equation (7)

$$p(x_{i+1} \mid x_{1:i}) \;=\; \mathrm{softmax}(E\phi(x_{1:i})_i) \tag{7}$$

Maximum likelihood: Let (θ) be all of the parameters of large language models. Let $(\mathcal{D})$ be the training data consisting of a set of sequences. Following the maximum likelihood in the principle function, we define the following negative log-likelihood objective function as a loss function $\mathcal{L}(\theta)$ given in Equation (8):

$$\mathcal{L}(\theta) \;=\; \sum_{x_{1:L}\in\mathcal{D}} -\log p_\theta(x_{1:L}) \;=\; \sum_{x_{1:L}\in\mathcal{D}} \sum_{i=1}^{L} -\log p_\theta(x_i \mid x_{1:i-1}) \tag{8}$$

Table 1. BLOOM training hyperparameter specification.

Hyperparameter	BLOOM-560M	BLOOM-1.1B	BLOOM-1.7B	BLOOM-3B	BLOOM-7.1B	BLOOM
			Architecture Hyperparameters			
Parameters	559 M	1065 M	1722 M	3003 M	7069 M	176,247 M
Precision			float16			bfloat16
Layers	24	24	24	30	30	70
Hidden dim	1024	1536	2048	2560	4096	14,336
Attention Heads	16	16	16	32	32	112
Vocab size			250,680			
Sequence length			2048			
Activation			GELU			
Position emb			ALiBi			
Tied emb			TRUE			

4.2. Theoretical Analysis

The efficiency of two model training techniques, namely LoRA [17,52] with 'DeepSpeed' and ZeRO-offload. LoRA focuses on reducing training time by adapting small trainable layers and freezing the backbone, while ZeRO aims to minimise computational costs by optimizing GPU allocation during the training process. Table 2 sets out a comparison of training times, batch sizes, and memory consumption for full-fine-tuning, LoRA, and when LoRA is combined with DeepSpeed on a single NVIDIA A100 processor.

Table 2. Comparison of training time, batch size, and memory consumption for full-fine-tuning, LoRA when combined with DeepSpeed.

	Time/Epoch	Global Batch Size	Memory
BLOOM	54 h	1	39 GB
BLOOM + LoRA	4 h	1	39 GB
BLOOM + LoRA + DeepSpeed	4 h	1	36 GB
BLOOM + LoRA + DeepSpeed	3 h	2	39.5 GB

4.2.1. Low Rank Adaptation

We have introduced "LoRA: Low-Rank Adaptation of Large Language Models" [17,52] in Section 3.5. The motivation for the use of 'LoRA' with 'BLOOM' (introduced in Section 3.3) in particular and the transformer model in general is that only a very small proportion of the parameters need to be trained when compared to the original model. Moreover, the performance of the model can achieve similar or even better results than training all the parameters in the original model.

As discussed in Section 4.1 (training objectives), the 'BLOOM' model has been selected based on the parameter set and layers along with the demonstrable successful use of the model in other similar studies. More specifically, based on Equation (4), the number of trainable parameters depends on the following parameters: (r), (d_{in}), (d_{out}), and the number of layers $\left(n_{layer}\right)$ in each backbone.

In line with the sources cited in this paper, we set hyperparameter $(; = 16)$ along with the other parameters that depend on the 'BLOOM' model including $\left(n_{layer} = 30\right)$ and $(d_{in} = d_{out} = 4096)$. Based on Equation (4), the number of training parameters can be calculated with the choice of $(r = 16)$ which, at approximately 7.5 million parameters (which accounts for only 0.11% of the total parameters that will be trained), contains less than the original 7 billion parameters.

The training time for the model is described in Table 2. For the original model, when training the entire dataset on a single A100 40 GB card, the time to train one epoch consisting of 100 k samples, with a batch size of 1, takes 54 h/100 k samples. With LoRA, the training time for one epoch is reduced to 4 h/100 k samples. Correspondingly, the time to train one epoch is reduced by nearly 14 times, which can help us evaluate a dataset more efficiently and perform full-fine-tuning on that dataset after testing LoRA.

4.2.2. DeepSpeed ZeRO-Offload

The ZeRO-offload [50], when used with 'DeepSpeed' [17,52], enables parallel processing of training components across multiple GPU streams including optimiser states, gradients, and model weights. This provides significant benefits for training models on multiple GPUs, such as increasing training speed and minimizing resource utilisation. Additionally, ZeRO-offload assists in balancing the load between GPUs during the model training process ensuring that each GPU is utilised effectively and does not experience slower performance than other GPUs.

ZeRO-offload collaborates with ZeRO to extend DL training across multiple GPUs. ZeRO comprises three stages, ZeRO-1, ZeRO-2, and ZeRO-3, each handling different aspects of model partitioning, including optimiser states, gradients, and parameters. While ZeRO-1 partitions only optimiser states, ZeRO-2 partitions gradients alongside optimiser states and ZeRO-3 partitions all model states. ZeRO-offload synergises with ZeRO-2. In ZeRO-2, every GPU retains a replica of all parameters but updates only its designated portion during each training step. As a result, each GPU stores only the optimiser states and gradients necessary for its update. Following the update, each GPU transmits its updated parameter subset to all other GPUs through an all-gather communication collective. The computation and communication schedule of ZeRO-2 are outlined as follows: During the forward pass, each GPU computes loss concerning a distinct mini-batch. During backward propagation, gradients are computed and then averaged using a reduce operator at the GPU (s) responsible for the gradient or its segment. Subsequently, each GPU updates its parameter subset and optimiser states using the averaged gradients. Finally, an all-gather operation is performed to obtain the remaining parameter updates computed on the other GPUs.

In summary, ZeRO-offload is an optimisation mechanism for training models on multiple GPUs that increases training speed and optimises resource utilisation. Moreover, with the added offloading mechanism, the CPU can load additional amounts of gradients, optimiser states, or parameters to reduce the burden on the GPU and free up VRAM.

As discussed in this paper, in our experimental testing and evaluation of our *Expert-B* model, following the use of LoRA the model training time is now only 4 h/1 epoch, with 39/40 GB of VRAM NVIDIA A100 serving the training process. Following the use of 'DeepSpeed' and the offload mechanism for optimiser states, the current training configuration with a batch size of 1 now only takes up 36/40 GB of VRAM. As there is still 4 GB of VRAM available, the batch size can be increased to 2. When increasing the batch size to 2, the training time for one epoch is reduced to 3 h/1 epoch on 1 NVIDIA A100s. We have conducted tests using the NVIDIA A100 40 GB GPU. If the VRAM is not fully utilised during training, it suggests that the hardware is not optimised to its full potential. Specifically, employing DeepSpeed with a batch_size of 1 can lower VRAM usage, indicating inefficient GPU utilisation. Consequently, we raised the batch_size to 2, allowing for the optimal utilisation of the GPU's capabilities.

4.2.3. Evaluation Parameters

The evaluation criteria (parameters) include the questions posed and the quality of the answers in terms of helpfulness, relevance, accuracy, and level of detail. The answers will be evaluated using the 'GPT-3.5-turbo' API [53,54] to assign scores where the performance (P) of Model A and Model B will be determined using Formula (9) where (n) is total question in the evaluation benchmark. Equation (9) was used in Phoenix [42]'s publication, where they used this formula to compare it with other language models.

$$Performance = \frac{\sum_{i=1}^{n} score_i^A}{\sum_{i=1}^{n} score_i^B} \tag{9}$$

where $\left(score_i^j\right)$ is the score for $(i - th)$ and the question for the (j) model. In this example, the formula in Equation (9) indicates the performance ratio of Model A compared to Model B. If the value of $Performance > 1$, it indicates that Model A performs better than Model B across the entire evaluation question dataset and vice versa.

4.2.4. Simulation Method

Here we consider the baseline for the study with 'Vicuna' [55]:

Baseline: a comparative analysis between the *Expert-B* and *Phoenix* methods because there are closely related similarities and both models employ the 'BLOOM' and a multilingual dataset. Phoenix has exhibited superior performance when compared to several Chinese language models, specifically reporting an 87% accuracy in English, an improvement over ChatGPT. Given that in the *Expert-B* model training covers both English and Vietnamese, a head-to-head comparison is conducted using Phoenix.

Vicuna: the 'Vicuna' question dataset [55] has been employed as the evaluation benchmark. 'Vicuna' comprises 80 questions categorised into 8 distinct groups. Since its inception, this evaluation protocol by Vicuna has been extensively used to establish evaluation criteria for language models that undergo instruction following fine-tuning. The benchmark dataset enables the assessment of a language model's capacity to comprehend and generate responses similar to those of a human for various types of prompts and questions.

4.2.5. Pre-Processing

The pre-processing stage involves: *prompting*, *word segmentation and encoding*, and the use of *decoding hyperparameters*.

Prompting

Before being used in either the training or inference processes, the question–answer pairs in the case of training, or questions in the case of inference, are subjected to the prompt shown in Figure 8 rather than being directly entered. The rationale for this approach is to initialise an input prompt, thus enabling the *Expert-B* and Phoenix models to activate zero-shot mode and understand the context of an ongoing conversation (i.e., between a human and an assistant-bot) that is also required to deliver not only a helpful response but also one that is polite and courteous.

```
A chat between a curious human and an artificial intelligence assistant.
The assistant gives helpful, detailed, and polite answers to the human's
questions.
Human:  <s> Instruction </s>
Assistant:  <s> Answer </s>.
```

Figure 8. Prompt used to wrap instruction used in the testing and evaluation for Phoenix and Vicuna.

Word Segmentation and Encoding

As the data produced by the GPT-3.5-turbo API is free of "messy characters and stop words", it can be directly used in the word segmentation and encoding stage. For this stage, the 'BLOOM' module forms the backbone, and thus the provided tokeniser in the 'BLOOM' module is used directly. Figure 9 displays the detailed configuration of the 'BLOOM' tokeniser.

```
"unk_token":  "<unk>", "eos_token":  "</s>", "bos_token":  "<s>",
          "pad_token":  "<pad>", "vocab_size":  250880
```

Figure 9. BLOOM tokeniser configuration.

During the training process, each input is divided into three different components: $(input\_ids)$, $(attention\_mask)$, and $(label\_ids)$. Assuming that after segmentation we have a set of (n) tokens of a sentence, then Equations (10) and (11) apply:

$$ids_i \;=\; token\_to\_ids(token_i) \tag{10}$$

$$input\_ \;=\; [ids_1,\, ids_2\, \ldots\, ids_n] \;\in \mathbb{R}^n \tag{11}$$

Based on the rules outlined in (10) every token in a sentence will be transformed into an ID that corresponds to the 'BLOOM' tokeniser vocabulary following segmentation. During the label masking stage, the $(label\_ids)$ (14) is used to identify which tokens in the $(input\_ids)$ sequence that the model needs to learn. The answer is the part of the input sequence the model must learn to correctly respond to the question. Therefore, during masking, all tokens except for the answer's (ids) to $((IGNORE\_ID) \;=\; (-100))$ must be set.

The following Equations (12) and (13) set out the conditions where the IN the answer (or) NOT in the answer apply, respectively; Equation (12) identifies the *label* and Equation (13) identifies the *mask*.

$$label_i = \begin{cases} ids_i & \text{if } ids_i \text{ in Answer} \\ -100 & \text{otherwise} \end{cases} \tag{12}$$

$$mask_i = \begin{cases} 1 & \text{if } ids_i \text{ not in Answer} \\ 0 & \text{otherwise} \end{cases} \tag{13}$$

As discussed in section A on masked self-attention, the attention mask (15) is used to mask the answer during training so that the model is not able to attend to it. In other words, during training, the attention mechanism can only attend to the non-masked tokens and must learn to infer the answer based on the context provided by these tokens.

$$label\_ids \;=\; [label_1,\, label_2 \ldots label_n] \in \mathbb{R}^n \tag{14}$$

$$attention\_mask \;=\; [mask_1,\, mask_2 \ldots mask_n] \;\in \mathbb{R}^n \tag{15}$$

Decoding Hyperparameters

To provide a basis for an unbiased comparison (of the *Expert-B* and Phoenix models) the same decoding *hyperparameters* were used as those employed by Vicuna [55]. In both cases a function is generated provided by 'Hugging Face' (ROOTS) where most of the configurations use the default settings such as (top_k) and (top_p), etc. However, Vicuna adjusts to the $(temperatureparameter)$ by setting it to (0.7) and also sets the (max_new_token) value to (1024).

Evaluation

As previously stated in Section 4.2.3, to assess the relative quality of two answers Phoenix employs the 'GPT-3.5-turbo' API to solicit ratings for potential answers. These ratings are based on criteria such as *helpfulness*, *relevance*, *accuracy*, and *level of detail*. This evaluation process is conducted specifically on the 80 English questions found within the Vicuna test set. The detail prompt was provided in Figure 10.

```
We would like to request your feedback on the performance of two AI assis-
tants in response to the user question displayed above.
Please rate the helpfulness, relevance, accuracy, and level of detail of
their responses.  Each assistant receives an overall score on a scale of 1
to 10, where a higher score indicates better overall performance.
Please first output a single line containing only two values indicating the
scores for Assistant 1 and 2, respectively.  The two scores are separated
by a space.
In the subsequent line, please provide a comprehensive explanation of your
evaluation, avoiding any potential bias and ensuring that the order in
which the responses were presented does not affect your judgement.
[Question]
Instruction
[Answer 1's start]
Answer 1
[Answer 1's end]
[Answer 2's start]
Answer 2
[Answer 2's end]
```

Figure 10. Evaluation prompt submitted to 'GPT-3.5-turbo' API to obtain score and evaluation description for two answers.

Once the scores for two answers are obtained from the 'GPT-3.5-turbo' API, Equation (9) is used to determine which model performed better across the complete evaluation dataset. Table 3 shows the performance ratio results between Expert-B and Phoenix. We can see that Expert-B outperformed Phoenix on the English benchmark, but performed slightly worse on the Vietnamese benchmark.

Table 3. Performance ratio (%) of Expert-B compared to Phoenix after obtaining score through 'GPT-3.5-turbo' and calculated by Equation (9).

Performance Ratio	English	Vietnamese
Expert-B vs. Phoenix	107, 89	96, 73

5. Experimental Results

In this section, we set out the results derived from the experimental testing using a case study. In the case study we set out and evaluate our *Expert-B* model based on the English and Vietnamese languages. Initially we set out a primary benchmark (English) followed by a comparative analysis for Vietnamese.

5.1. Case Study of English Benchmark

In our evaluation of *Expert-B* we have based the performance ratio on the one introduced in Section 4.2.3. The results may be summarised as follows:

- When in a comparison with Phoenix [42] the result is greater than 1, it can be concluded that *Expert-B* outperformed Phoenix for the English benchmark. This observation is further supported by the category scores shown in Table 4 where our *Expert-B* model improved on the performance of Phoenix with a total score of 51 wins out of 80 categories, compared to Phoenix which achieved 29 wins.
- The regeneration method used to generate the study dataset has provided more detailed answers compared to the original method used by Alpaca [48]. This suggests that the performance of *Expert-B* in the English benchmark is even more impressive, as

it was able to outperform Phoenix using more detailed answers. In particular, *Expert-B* dominates Phoenix in the remaining criteria categories, namely writing, knowledge, and generic, with *Expert-B* taking almost all of the scores in these categories.

- Overall, the results (see Table 4) suggest that *Expert-B* has a better performance than Phoenix in the English benchmark.

Table 4. Details of the number of wins for each model over the categories in both English and Vietnamese. The bold numbers indicate the model that won in each category.

	English			Vietnamese	
Categorical	**Phoenix**	**Expert-B**	**Total**	**Phoenix**	**Expert-B**
coding	**5**	2	7	**6**	1
common-sense	2	**8**	10	4	**6**
counter-factual	4	**6**	**6**	4	
fermi	**9**	1	10	**8**	2
generic	2	**8**	10	**5**	**5**
knowledge	1	**9**	10	3	**7**
math	1	**2**	3	**3**	0
roleplay	4	**6**	10	4	**6**
writing	1	**9**	10	**5**	**5**
Total wins	29	**51**	80	**44**	36

5.2. Case Study of Vietnamese's Vicuna Benchmark

The experimental results shown in Table 4 are the results related to the Vietnamese benchmark. In a comparative analysis we can see that the *Expert-B* and Phoenix models displayed closer competition.

For a total of 80 categories, the *Expert-B* model showed an improvement of model's performance in 36 categories while the Phoenix model performed better in 44 categories for the Vietnamese benchmark. The Phoenix model dominated in more categories such as coding, fermi, and knowledge. In contrast, *Expert-B* only managed to generate categories.

In summary, the results for the Vietnamese benchmark are as follows:

- The Phoenix model showed a better performance than *Expert-B* in some categories, particularly in coding and fermi. However, *Expert-B* demonstrated better performance in several other categories such as common-sense, counter-factual, and writing.
- The overall results for the two models for the Vietnamese benchmark were much closer than in the English benchmark, with the margin for the Phoenix model's performance improvements being relatively small.

5.3. Case Study of VLSP Benchmark

This benchmark utilised is the VLSP-LLM 2023 [56], which mirrors HuggingFace's Open-LLM Leaderboard [57]. However, it is customised for the Vietnamese language. It consists of four unique evaluations: ARC Challenge, HellaSwag, MMLU, and TruthfulQA, which are fine-tuned for the Vietnamese language. This extensive suite of benchmarks facilitates a thorough assessment of the language models' abilities to comprehend Vietnamese text across diverse domains and levels of complexity. Within this benchmark, we have conducted an analysis among our model, Expert-B, and the subsequent models as depicted in Table 5 as follows:

- Bkai-foundation-models/vietnamese-llama2-7b-40 GB: This model is a LLaMA-2 variant, which has an extended vocabulary size in Vietnamese and has been pretrained on a Vietnamese corpus.
- SeaLLMs/SeaLLM-7B-v1 [58]: Multilingual LLM supported language in South-East Asia Countries, which achieved SOTA on a multi-SEA benchmark.
- Vinai/PhoGPT-7B5-Instruct [59]: A monolingual model that has been developed for an instruction following ability for chatbots operating in Vietnamese.

Table 5. VLSP benchmark score of aforementioned models.

Model	arc_vi	hellaswag_vi	mmlu_vi	truthfulqa_vi	Average
vinai/PhoGPT-7B5-Instruct	26.41	40.55	26.24	45.82	34.76
bkai-foundation-models/vietnamese-llama2-7b-40GB	29.49	43.92	33.83	45.28	38.13
sealion-7b	27.01	48.33	26.50	42.77	36.15
Expert-B	**33.68**	**49.10**	**35.57**	**51.4**	**42.44**

Experimental results show that Expert-B has demonstrated superior performance, surpassing all other models with impressive results and significantly outstripping its counterparts. In the realm of supervised fine-tuning, our findings consistently demonstrate the exceptional performance of our model, Expert-B, surpassing other models in the comparison. This notable accomplishment underscores the effectiveness of the fine-tuning methodology and the model's adeptness at capitalizing on its training data to achieve outstanding results, even when pitted against larger counterparts. Furthermore, it is noteworthy that Expert-B does not undergo continued pretraining in Vietnamese, yet it still outperforms the aforementioned models, achieving remarkable performance levels. The success of Expert-B emphasises the substantial potential of well-executed fine-tuning approaches with synthetic data in enhancing the capabilities of large language models, particularly in contexts necessitating specialised language processing.

5.4. Analysis

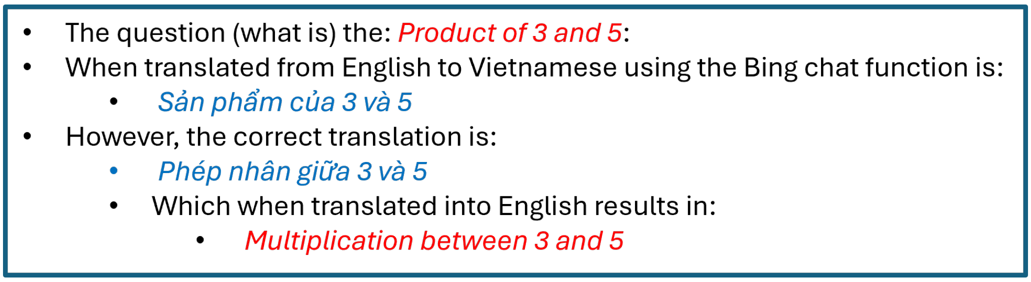
Considering the results derived from our experimental testing in the comparative analysis we may draw a number of conclusions and observations.

5.4.1. Dataset

In previous section(s) we have noted the performance improvements of *Expert-B* over Phoenix for the English benchmark, while the performance ratio was not significantly higher, this still demonstrated that the combination of the English instruction dataset and the identify model produced much better answers compared to the basic method of generating answers using only instruction. However, in terms of Vietnamese data generation *Expert-B* shows an inferior performance as compared to Phoenix with a performance rate of only 96%.

When compared directly to Phoenix there are two main differences that we consider that lead to the lower performance for *Expert-B* with respect to the Vietnamese language:

1. The Vietnamese data generation process consists of two phases: using 'GPT-3.5-turbo' API to translate instructions into Vietnamese, and then generating answers using the pipeline dataset introduced in part A.
2. If the translated instructions are not semantically accurate and in line with the original Vietnamese, the resulting data can significantly affect the quality and accuracy of the translation. An example of the problem and the disadvantage of using the 'GPT-3.5-turbo' API for translation are shown in Figure 11 where the inaccurate (i.e., wrong) translation from English to Vietnamese is demonstrated. The translation shown in Figure 11 was carried out using the Bing Chat function in the Microsoft Edge browser Version 117.0.2045.47 (Official build) (64-bit).



- The question (what is) the: *Product of 3 and 5*:
- When translated from English to Vietnamese using the Bing chat function is:
  - *Sản phẩm của 3 và 5*
- However, the correct translation is:
  - *Phép nhân giữa 3 và 5*
  - Which when translated into English results in:
    - *Multiplication between 3 and 5*

Figure 11. A simple example demonstrating a semantic translation error and the disadvantage of using the 'GPT-3.5-turbo' API for a translation from English to Vietnamese.

3. The inaccuracy is clear and is the result of a fundamental misunderstanding of the semantics in the question "Product of 3 and 5" where the word "product" relates to a mathematical function (i.e., "multiplication"). Frequently in translation software general English is handled well but scientific terms (not part of the language corpus) are incorrectly translated.
4. For this study, we believe the reason for the incorrect translation is because Phoenix has a conversation dataset consisting of dialogue between users and ChatGPT with the resulting questions being clearer and created by actual human users. Such inaccuracies are all too common in translation software.

5.4.2. Parameter Efficient Fine-Tuning

Parameter-Efficient Fine-Tuning (PEFT) is a type of model tuning that selectively fine-tunes only a subset of a model's parameters. This approach is beneficial because it requires only a small fraction of the total number of parameters in the backbone model to achieve a substantial improvement in performance. In some cases, PEFT can even surpass traditional fine-tuning methods. PEFT encompasses various techniques including P-tuning [8], prompt-tuning [60], and LoRA [52].

PEFT's popularity stems from its convenience in reducing the computational overhead associated with fine-tuning very large models, a feature particularly relevant to LLMs. However, there are drawbacks with PEFT including the inability to utilise all of the parameters in a model to be learnt; this may limit its ability to match the performance of traditional fine-tuning methods. Numerous studies, for example see [17], provide a comparative analysis to compare the performance of PEFT to that of full fine-tuning; studies have viewed this as a trade-off between computational cost and model accuracy and quality.

Retrospectively examining these concerns, we can draw a comparison between *Expert-B* and Phoenix that while Phoenix has undergone complete fine-tuning, *Expert-B* has prioritised optimising computational cost by utilising LoRA. Consequently, the limited data available may not permit the exploitation of all learning parameters, leading to inferior performance by *Expert-B* for in the Vietnamese benchmark Vicuna [55] evaluation dataset compared to Phoenix.

5.4.3. Further Investigation Improvement

Firstly, as analysed above, the applied 'GPT-3.5-turbo' results in decreased quality of Vietnamese instructions as it may produce inaccurate or loosely related outcomes compared to the original instructions. To address this issue, we can consider two alternative approaches: using a different machine translation model to translate the instructions or collecting instructions from the Internet.

Secondly, indirect LoRA usage limits the model's performance as it only learns from a small number of parameters. To improve this, full-parameter fine-tuning can be employed, although it may lead to significant consumption of training resources.

Thirdly, recent advancements in large language models, such as LLaMA 2 and Mistral, continuously raise the performance bar, gradually diminishing the performance of older models like BLOOM. To enhance the model, we can replace the backbone with newer, better-performing models like LLaMA or Mistral, depending on the specific use case of SMEs. For instance, LLaMA and Mistral models excel primarily in English language tasks.

These strategies can help overcome the identified limitations and enhance the performance of the model, ensuring its suitability for diverse enterprise applications.

6. Discussion

The design and development of a GenAI chatbot is a highly resource intensive activity that also requires an appropriate LLM (a language specific corpus); such demands make the development of a 'bespoke' chatbot by SMEs impractical. Chatbots are generally domain-specific and while there are proprietary cloud-based options, it is generally impractical to make any significant changes to such systems that may not suit a specific domain. The motivation for this study lies in the growing demand for chatbot technology from organisations of all sizes in heterogeneous domains. Here, we consider the development of a 'bespoke' domain-specific GenAI-driven chatbot for SMEs designed to automate question–response interactions. To achieve this aim, we created a GenAI model for a chatbot complete with an LLM that can adapt to multiple languages (in this study, the focus is on Vietnamese and English) for use in GenAI models suitable for resource limited SMEs.

Identifying a resolution to this problem is important as chatbots can offer significant organisational and commercial benefits for organisations of all types. To facilitate the development of a chatbot with an appropriate LLM, we developed the *Expert-B* model which utilises an 'open-source' code that uses 'BLOOM' as its backbone. The *Expert-B* model provides benefits which include a reduction in computational training time and overhead with an effective and flexible basis for bespoke implementation(s). This research contributes to the discussion on how GenAI can be leveraged to maximum effect for SMEs. In this study, we propose a method for creating bilingual instruction datasets for English and Vietnamese which, when combined with model training using 'Low-Rank Adaptation' and 'DeepSpeed', will contribute to a reduction in training time and computational cost. Moreover, we posit that our proposed approach will generalise to other languages.

In experimental testing, *Expert-B* achieves approximately 107% performance compared to Phoenix, which achieved 92% performance compared to ChatGPT on the English benchmark. Moreover, the training time was reduced to be 18 times shorter than the normal training method.

We have considered the positive and negative aspects of GenAI and chatbots with a focus on ChatGPT, and in Section 6 we consider ORQ with proposed directions for future research. However, as briefly considered in Section 2, there are issues relating to the socio-technical affects of DI. (GenAI is an example of DI [20,22,24,26,27,32,33], which is reflected in delays in understanding the impact(s) [35] and the nature of the affects [24]). Such issues are beyond the scope of this paper but represent significant challenges from a design, implementation, and research perspective and represent important topics for future research.

Moreover, GenAI-driven chatbots must be designed with strict guidelines and ethical considerations [20,22] to consider the following :

- Prevent them from sharing sensitive or inappropriate information;
- Ensure the safety and privacy of users;
- These considerations are essential to build trust in chatbots and enable their adoption

However, as discussed in Section 2, while the affects of DI are understood, there are still delays in understanding the related impacts and affects of DI [35] along with the nature of such affects [24]. Such issues are beyond the scope of this paper but warrant serious consideration and represent important topics for future research into information systems design. There is a correlation between this observation and the argument made in [61] that, with reference to AI, which states: "not only do we lack the tools to determine what achievements will be attained in the near future, but we even ignore what various technologies in present-day AI are capable of". GenAI is 'out of the bag' [24] and it may be viewed as a 'Pandoras box', the opening of which is irrevocable.

Open Research Questions

In this paper, we have considered chatbots and LLMs, and the development of LLMs has shown great potential in the improvement of chatbots' performance. While this study has addressed a number of research questions, persistent open research questions (ORQ) and problems remain that need to be addressed in question–response interactions. To address the problems for incorrect outcomes and inaccurate interactions, we have considered the following potential solution(s):

- A Reinforcement Learning from Human Feedback (RLHF) method can be designed to improve the quality and safety of chatbot responses. By receiving feedback on responses in the experiments, we can evaluate the usefulness, safety, and other aspects of each response and then develop a reward model to ensure the quality of the response.
- RLHF may be integrated into the chatbot training process with the chatbot generating responses based on its current model and users provide iterative feedback on the quality and safety of the responses. The chatbot can be trained using reinforcement learning algorithms to maximise the reward score assigned to each response, resulting in higher quality and safer responses.

By incorporating a RLHF system into the *Expert-B* model, a chatbot can learn from human feedback and adapt to user preferences, while also ensuring the safety and privacy of responses. This can help build trust and engagement with users, leading to a more effective and user-friendly chatbot experience. Incorporating a RLHF system represents an interesting and potentially fruitful direction for future research. Notwithstanding the ORQ, we posit that the *Expert-B* model, when combined with the RLHF system, provides a promising approach to address the challenges faced by chatbots and LLMs. From a practical managerial significance perspective, the proposed method as set out in this study has the potential to significantly enhance the performance and reduce the querying cost of ChatGPT in large domains. Furthermore, training RLHF has become less challenging. As RLHF datasets are now more common and widely publicised, this facilitates the training of large language models making it simpler and cost-effective compared to manual labelling processes. A further approach to generating domain-specific RLHF data for businesses is to deploy the model on a specific user group, collect chat logs, and then evaluate them. Although this method may be more costly and time-consuming, the data quality will be higher as it targets a specific user group.

In the domain of large language models, the generation of hallucination responses is an unavoidable challenge. Applying large language models as applications must be carefully considered for user questions–answers. We can build rules to filter them before feeding them into the model. Alternatively, we can train the model to reject questions likely to contain such information. One of the most popular techniques currently used to help models avoid sensitive cases and respond according to human preference is RLHF, which is extensively employed in current LLMs. Human preference datasets can be collected from user chat data or taken from public datasets. After being fine-tuned with these data, the model can provide safer responses and better meet user requirements.

7. Conclusions

We have presented our *Expert-B* model designed to provide an effective basis upon which a GenAI-driven chatbot with an appropriate domain-specific LLM can be realised for resource-limited SMEs. This research contributes to the discussion on how generative AI can be leveraged to maximum effect for small- and medium-sized enterprises constructively. Specifically, we introduced the expert-prompting method to generate high-quality synthetic data, which was then validated by our model outperforming Phoenix on English domains. Additionally, we optimised the model training processes by combining two techniques: LoRA and DeepSpeed. The pervasive nature of GenAI and chatbots is demonstrated by their adoption in heterogeneous domains and systems. GenAI may be considered in term of a domain-specific information system and, accordingly, information system design must attempt to address (or at least mitigate) the negative affects while still promoting the positive aspects. This research contributes to the discussion on how GenAI can be leveraged to maximum effect for SMEs. The proposed *Expert-B* model provides an effective basis upon which this objective may be realised constructively.

In future work, we will investigate how to create a virtual assistant that approximates the quality of ChatGPT using only open-source resources and minimizing computational costs for domains in smart cities. This virtual assistant will be based on augmenting the answer sentences for each instruction by adding an identity role to each instruction, and we will train the model using Parameter Efficient Fine-Tuning and DeepSpeed techniques in order to save computational resources. Further studies will investigate personalisation, conversational capabilities, and trustworthiness by dealing with multimodal design for the future of ChatGPT.

Author Contributions: Conceptualisation and methodology: N.T.T., H.V.P. and P.M.; software: D.H.V.T. and H.V.P.; validation: H.V.P., D.H.V.T. and P.M.; formal analysis: N.T.T. and D.H.V.T.; investigation: D.H.V.T. and H.V.P.; data: D.H.V.T.; writing—original draft preparation: N.T.T. and H.V.P.; writing—review and editing: H.V.P. and P.M.; project administration: H.V.P. All authors have read and agreed to the published version of the manuscript.

Funding: This research is funded by National Economics University, Vietnam.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The source code and data sets of the study are available at https://github.com/dat-browny/Expert-B/tree/master/models (accessed on 20 December 2023).

Acknowledgments: The authors thank the language experts who participated in the experiments using ChatGPT in both English and Vietnamese languages.

Conflicts of Interest: The authors declare no conflicts of interest. This research does not involve any human or animal participation. All authors have checked and agreed with the submission.

References

1. Thomas, A.M.; Moore, P.; Evans, C.; Shah, H.; Sharma, M.; Mount, S.; Xhafa, F.; Pham, H.V.; Barolli, L.; Patel, A.; et al. Smart care spaces: Pervasive sensing technologies for at-home care. *Int. J. Ad Hoc Ubiquitous Comput.* **2014**, *16*, 268–282. [CrossRef]
2. Pham, H.V.; Le Hoang, T.; Hung, N.Q.; Phung, T.K. Proposed Intelligent Decision Support System Using Hedge Algebra Integrated with Picture Fuzzy Relations for Improvement of Decision-Making in Medical Diagnoses. *Int. J. Fuzzy Syst.* **2023**, *25*, 3260–3270. [CrossRef]
3. Christensen, C.M.; McDonald, R.; Altman, E.J.; Palmer, J.E. Disruptive Innovation: An Intellectual History and Directions for Future Research. *J. Manag. Stud.* **2018**, *55*, 1043–1078. [CrossRef]
4. Van Pham H.; Kinh Phung, T.; Hung, N.Q.; Dong, L.D.; Trung, L.T.; Dao, N.T.X.; Kieu, P.T.T.; Xuan, T.N. Proposed Distance and Entropy Measures of Picture Fuzzy Sets in Decision Support Systems. *Int. J. Fuzzy Syst.* **2023**, *44*, 6775–6791. [CrossRef]
5. Pham, H.V.; Duong, P.V.; Tran, D.T.; Lee, J.-H. A Novel Approach of Voterank-Based Knowledge Graph for Improvement of Multi-Attributes Influence Nodes on Social Networks. *J. Artif. Intell. Soft Comput. Res.* **2023**, *13*, 165. [CrossRef]
6. Pham, V.H.; Nguyen, Q.H.; Troung, V.P.; Tran, L.P.T. The Proposed Context Matching Algorithm and Its Application for User Preferences of Tourism in COVID-19 Pandemic. In Proceedings of the International Conference on Innovative Computing and Communications, Delhi, India, 19–20 February 2022; Volume 471. [CrossRef]
7. Crosthwaite, P.; Baisa, V. Generative AI and the end of corpus-assisted data-driven learning? Not so fast! *Appl. Corpus Linguist.* **2023**, *3*, 100066. [CrossRef]
8. Liu, X.; Ji, K.; Fu, Y.; Tam, W.; Du, Z.; Yang, Z.; Tang, J. P-Tuning: Prompt Tuning Can Be Comparable to Fine-tuning Across Scales and Tasks. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics, Dublin, Ireland, 22–27 May 2022; Volume 2: Short Papers, pp. 61–68. [CrossRef]
9. Marjanovic, O.; Skaf-Molli, H.; Molli, P.; Godart, C. Collaborative practice-oriented business processes Creating a new case for business process management and CSCW synergy. In Proceedings of the 2007 International Conference on Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom 2007), New York, NY, USA, 12–15 November 2007; pp. 448–455.
10. Clarysse, B.; He, V.F.; Tucci, C.L. How the Internet of Things reshapes the organization of innovation and entrepreneurship. *Technovation* **2022**, *118*, 102644. [CrossRef]
11. Puranam, P.; Alexy, O.; Reitzig, M. What's "New" About New Forms of Organizing? *Acad. Manag. Rev.* **2014**, *39*, 162–180. [CrossRef]

12. Hagendorff, T.; Fabi, S.; Kosinski, M. Human-like intuitive behavior and reasoning biases emerged in large language models but disappeared in ChatGPT. *Nat. Comput. Sci.* **2023**, *3*, 833–838. [CrossRef]
13. Anam Nazir, Z.W. A comprehensive survey of ChatGPT: Advancements, applications, prospects, and challenges. *Meta-Radiol.* **2023**, *1*, 100022. [CrossRef]
14. López Espejel, J.; Ettifouri, E.H.; Yahaya Alassan, M.S.; Chouham, E.M.; Dahhane, W. GPT-3.5, GPT-4, or BARD? Evaluating LLMs reasoning ability in zero-shot setting and performance boosting through prompts. *Nat. Lang. Process. J.* **2023**, *5*, 100032. [CrossRef]
15. Scao, T.L.; Fan, A.; Akiki, C.; Pavlick, E.; Ilić, S.; Hesslow, D.; Castagné, R.; Luccioni, A.S.; Yvon, F.; Gallé, M.; et al. BLOOM A 176B-Parameter Open-Access Multilingual Language Model. *arXiv* **2022**, arXiv:2211.05100. [CrossRef]
16. Laurençon, H.; Saulnier, L.; Wang, T.; Akiki, C.; Villanova del Moral, A.; Le Scao, T.; Von Werra, L.; Mou, C.; González Ponferrada, E.; Nguyen, H.; et al. The BigScience ROOTS Corpus: A 1.6TB Composite Multilingual Dataset. In *Proceedings of the Advances in Neural Information Processing Systems*; Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., Oh, A., Eds.; Curran Associates, Inc.: New York, NY, USA, 2022; Volume 35, pp. 31809–31826.
17. Sun, X.; Ji, Y.; Ma, B.; Li, X. A Comparative Study between Full-Parameter and LoRA-based Fine-Tuning on Chinese Instruction Data for Instruction Following Large Language Model. *arxiv* **2023**, arXiv:2304.08109.
18. Fosso Wamba, S.; Queiroz, M.M.; Chiappetta Jabbour, C.J.; Shi, C.V. Are both generative AI and ChatGPT game changers for 21st-Century operations and supply chain excellence? *Int. J. Prod. Econ.* **2023**, *265*, 109015. [CrossRef]
19. Varghese, J.; Chapiro, J. ChatGPT: The transformative influence of generative AI on science and healthcare. *J. Hepatol.* **2023**, *80*. [CrossRef]
20. Dwivedi, Y.K.; Kshetri, N.; Hughes, L.; Slade, E.L.; Jeyaraj, A.; Kar, A.K.; Baabdullah, A.M.; Koohang, A.; Raghavan, V.; Ahuja, M.; et al. Opinion Paper: "So what if ChatGPT wrote it?" Multidisciplinary perspectives on opportunities, challenges and implications of generative conversational AI for research, practice and policy. *Int. J. Inf. Manag.* **2023**, *71*, 102642. [CrossRef]
21. Eke, D.O. ChatGPT and the rise of generative AI: Threat to academic integrity? *J. Responsible Technol.* **2023**, *13*, 100060. [CrossRef]
22. Evans, O.; Wale-Awe, O.; Osuji, E.; Ayoola, O.; Alenoghena, R.; Adeniji, S. ChatGPT impacts on access-efficiency, employment, education and ethics: The socio-economics of an AI language model. *BizEcons Q.* **2023**, *16*, 1–17.
23. Baidoo-Anu, D.; Owusu Ansah, L. Education in the era of generative artificial intelligence (AI): Understanding the potential benefits of ChatGPT in promoting teaching and learning. *SSRN* **2023**. [CrossRef]
24. Sætra, H.S. Generative AI: Here to stay, but for good? *Technol. Soc.* **2023**, *75*, 102372. [CrossRef]
25. Alabool, H.M. ChatGPT in Education: SWOT analysis approach. In Proceedings of the 2023 International Conference on Information Technology (ICIT), Amman, Jordan, 9–10 August 2023; pp. 184–189. [CrossRef]
26. Utterback, J.M.; Acee, H.J. Disruptive technologies: An expanded view. *Int. J. Innov. Manag.* **2005**, *9*, 1–17. [CrossRef]
27. Christensen, C.; Raynor, M.E.; McDonald, R. *Disruptive Innovation*; Harvard Business Review: Brighton, MA, USA, 2013.
28. Fleck, J.; Howells, J. Technology, the Technology Complex and the Paradox of Technological Determinism. *Technol. Anal. Strateg. Manag.* **2001**, *13*, 523–531. [CrossRef]
29. Wyatt, S. Technological determinism is dead; long live technological determinism. In *The Handbook of Science and Technology Studies*; MIT Press: Cambridge, MA, USA, 2008; Volume 3, pp. 165–180.
30. Hallström, J. Embodying the past, designing the future: Technological determinism reconsidered in technology education. *Int. J. Technol. Des. Educ.* **2020**, 17–31. [CrossRef]
31. Sandrini, L.; Somogyi, R. Generative AI and deceptive news consumption. *Econ. Lett.* **2023**, *232*, 111317. [CrossRef]
32. Möslein, K.M.; Neyer, A.K. Disruptive Innovation. In *Wiley Encyclopedia of Management*; John Wiley & Sons: Hoboken, NJ, USA, 2015; pp. 1–3. [CrossRef]
33. Markides, C. Disruptive Innovation: In Need of Better Theory. *J. Prod. Innov. Manag.* **2006**, *23*, 19–25. [CrossRef]
34. Mackenzie, D. Surprising Advances in Generative Artificial Intelligence Prompt Amazement—And Worries. *Engineering* **2023**, *25*, 9–11. [CrossRef]
35. Checkland, P.; Holwell, S. *Information, Systems and Information Systems: Making Sense of the Field*; John Wiley and Sons: Chichester, UK, 1997.
36. Lybaert, C.; Van Hoof, S.; Deygers, B. The influence of ethnicity and language variation on undergraduates' evaluations of Dutch-speaking instructors in Belgium: A contextualized speaker evaluation experiment. *Lang. Commun.* **2022**, *84*, 1–19. [CrossRef]
37. Li, M.; Wang, R. Chatbots in e-commerce: The effect of chatbot language style on customers' continuance usage intention and attitude toward brand. *J. Retail. Consum. Serv.* **2023**, *71*, 103209. [CrossRef]
38. Xin, D.; Mao, J.; Liu, M. The Effects of Parasocial Relationships in the Adoption of Mobile Commerce Application: A Conceptual Model. In Proceedings of the 2010 International Conference on E-Business and E-Government, Guangzhou, China, 7–9 May 2010; pp. 149–152. [CrossRef]
39. Li, R.Y.; Lin, S. Stages of Concern and Parasocial Interaction: Perception, Attitude, and Adoption of Social Media. In Proceedings of the 2018 1st IEEE International Conference on Knowledge Innovation and Invention (ICKII), Jeju, Republic of Korea, 23–27 July 2018; pp. 362–364. [CrossRef]

40. Hanief, S.; Handayani, P.W.; Azzahro, F.; Pinem, A.A. Parasocial Relationship Analysis on Digital Celebrities Follower's Purchase Intention. In Proceedings of the 2019 2nd International Conference of Computer and Informatics Engineering (IC2IE), Banyuwangi, Indonesia , 10–11 September 2019; pp. 12–17. [CrossRef]
41. Chen, W.K.; Wen, H.Y.; Silalahi, A.D.K. Parasocial Interaction with YouTubers: Does Sensory Appeal in the YouTubers' Video Influences Purchase Intention? In Proceedings of the 2021 IEEE International Conference on Social Sciences and Intelligent Management (SSIM), Taichung, Taiwan , 29–31 August 2021; pp. 1–8. [CrossRef]
42. Chen, Z.; Jiang, F.; Chen, J.; Wang, T.; Yu, F.; Chen, G.; Zhang, H.; Liang, J.; Zhang, C.; Zhang, Z.; et al. Phoenix: Democratizing ChatGPT across Languages. *arXiv* **2023**, arXiv2304.10453.
43. Kohnke, L.; Moorhouse, B.L.; Zou, D. Exploring generative artificial intelligence preparedness among university language instructors: A case study. *Comput. Educ. Artif. Intell.* **2023**, *5*, 100156. [CrossRef]
44. Dai, Y.; Liu, A.; Lim, C.P. Reconceptualizing ChatGPT and generative AI as a student-driven innovation in higher education. *Procedia CIRP* **2023**, *119*, 84–90. [CrossRef]
45. Yilmaz, R.; Karaoglan Yilmaz, F.G. The effect of generative artificial intelligence (AI)-based tool use on students' computational thinking skills, programming self-efficacy and motivation. *Comput. Educ. Artif. Intell.* **2023**, *4*, 100147. [CrossRef]
46. Thomas, R.L.; Uminsky, D. Reliance on metrics is a fundamental challenge for AI. *Patterns* **2022**, *3*, 100476. [CrossRef] [PubMed]
47. Touvron, H.; Lavril, T.; Izacard, G.; Martinet, X.; Lachaux, M.A.; Lacroix, T.; Rozière, B.; Goyal, N.; Hambro, E.; Azhar, F.; et al. LLaMA: Open and Efficient Foundation Language Models. *arXiv* **2023**, arXiv:2302.13971.
48. Taori, R.; Gulrajani, I.; Zhang, T.; Dubois, Y.; Li, X.; Guestrin, C.; Liang, P.; Hashimoto, T. *Stanford Alpaca: A Strong, Replicable Instruction-Following Model*; Center for Research on Foundation Models, Stanford University: Stanford, CA, USA, 2023.
49. Aghajanyan, A.; Zettlemoyer, L.; Gupta, S. Intrinsic Dimensionality Explains the Effectiveness of Language Model Fine-Tuning. *arXiv* **2020**, arXiv:2012.13255.
50. Ren, J.; Rajbhandari, S.; Aminabadi, R.Y.; Ruwase, O.; Yang, S.; Zhang, M.; Li, D.; He, Y. ZeRO-Offload: Democratizing Billion-Scale Model Training. *arXiv* **2021**, arXiv:2101.06840.
51. Press, O.; Smith, N.A.; Lewis, M. Train Short, Test Long: Attention with Linear Biases Enables Input Length Extrapolation. *arXiv* **2022**, arXiv:2108.12409.
52. Hu, E.; Shen, Y.; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; Wang, L.; Chen, W. LoRA: Low-Rank Adaptation of Large Language Models. *arXiv* **2022**, arXiv:2106.09685.
53. Han, S.J.; Ransom, K.J.; Perfors, A.; Kemp, C. Inductive reasoning in humans and large language models. *Cogn. Syst. Res.* **2024**, *83*, 101155. [CrossRef]
54. Kunst, J.R.; Bierwiaczonek, K. Utilizing AI questionnaire translations in cross-cultural and intercultural research: Insights and recommendations. *Int. J. Intercult. Relations* **2023**, *97*, 101888. [CrossRef]
55. Zheng, L.; Chiang, W.L.; Sheng, Y.; Zhuang, S.; Wu, Z.; Zhuang, Y.; Lin, Z.; Li, Z.; Li, D.; Xing, E.P.; et al. Judging LLM-as-a-judge with MT-Bench and Chatbot Arena. *arXiv* **2023**, arXiv:2306.05685.
56. Cuong, L.A.; Hieu, N.T.; Cuong, N.V.; Que, N.N.; Nguyen, L.-M.; Nguyen, C.-T. Vlsp 2023 Challenge on Vietnamese Large Language Models 2023. 2023. Available online: https://vlsp.org.vn/vlsp2023/eval/vllm (accessed on 30 March 2024).
57. Beeching, E.; Fourrier, C.; Habib, N.; Han, S.; Lambert, N.; Rajani, N.; Sanseviero, O.; Tunstall, L.; Wolf, T. Open LLM Leaderboard. 2023. Available online: https://huggingface.co/spaces/HuggingFaceH4/open_llm_leaderboard (accessed on 30 March 2024).
58. Nguyen, Xu.; Zhang, W.; Li, X.; Aljunied, M.; Tan, Q.; Cheng, L.; Chen, G.; Deng, Y.; Yang, S.; Liu, C.; Zhang, H.; Bing, L. SeaLLMs—Large Language Models for Southeast Asia. *arXiv* **2023**, arXiv:2312.00738.
59. Nguyen, D.Q.; Nguyen, L.T.; Tran, C.; Nguyen, D.N.; Phung, D.; Bui, H. PhoGPT: Generative Pre-training for Vietnamese. *arXiv* **2023**, arXiv:2311.02945.
60. Lester, B.; Al-Rfou, R.; Constant, N. The Power of Scale for Parameter-Efficient Prompt Tuning. *arXiv* **2021**, arXiv:2104.08691.
61. Martínez-Plumed, F.; Gómez, E.; Hernández-Orallo, J. Futures of artificial intelligence through technology readiness levels. *Telemat. Inform.* **2021**, *58*, 101525. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.