big data and cognitive computing

MDPI

Article

Computers' Interpretations of Knowledge Representation Using Pre-Conceptual Schemas: An Approach Based on the BERT and Llama 2-Chat Models

Jesus Insuasti [1,*], Felipe Roa [1] and Carlos Mario Zapata-Jaramillo [2]

[1] Systems Engineering Department, University of Nariño, Pasto 520001, Colombia; feliperoa@udenar.edu.co
[2] Computer and Decision Science Department, Universidad Nacional de Colombia, Medellín 050034, Colombia; cmzapata@unal.edu.co
* Correspondence: insuasti@udenar.edu.co

Abstract: Pre-conceptual schemas are a straightforward way to represent knowledge using controlled language regardless of context. Despite the benefits of using pre-conceptual schemas by humans, they present challenges when interpreted by computers. We propose an approach to making computers able to interpret the basic pre-conceptual schemas made by humans. To do that, the construction of a linguistic corpus is required to work with large language models—LLM. The linguistic corpus was mainly fed using Master's and doctoral theses from the digital repository of the University of Nariño to produce a training dataset for re-training the BERT model; in addition, we complement this by explaining the elicited sentences in triads from the pre-conceptual schemas using one of the cutting-edge large language models in natural language processing: Llama 2-Chat by Meta AI. The diverse topics covered in these theses allowed us to expand the spectrum of linguistic use in the BERT model and empower the generative capabilities using the fine-tuned Llama 2-Chat model and the proposed solution. As a result, the first version of a computational solution was built to consume the language models based on BERT and Llama 2-Chat and thus automatically interpret pre-conceptual schemas by computers via natural language processing, adding, at the same time, generative capabilities. The validation of the computational solution was performed in two phases: the first one for detecting sentences and interacting with pre-conceptual schemas with students in the Formal Languages and Automata Theory course—the seventh semester of the systems engineering undergraduate program at the University of Nariño's Tumaco campus. The second phase was for exploring the generative capabilities based on pre-conceptual schemas; this second phase was performed with students in the Object-oriented Design course—the second semester of the systems engineering undergraduate program at the University of Nariño's Tumaco campus. This validation yielded favorable results in implementing natural language processing using the BERT and Llama 2-Chat models. In this way, some bases were laid for future developments related to this research topic.

Keywords: pre-conceptual schema; computational linguistics; linguistic corpus; language models

Citation: Insuasti, J.; Roa, F.; Zapata-Jaramillo, C.M. Computers' Interpretations of Knowledge Representation Using Pre-Conceptual Schemas: An Approach Based on the BERT and Llama 2-Chat Models. *Big Data Cogn. Comput.* **2023**, *7*, 182. https://doi.org/10.3390/bdcc7040182

Academic Editors: Gianvincenzo Alfano, Alejandro Javier García and Francesco Parisi

Received: 23 October 2023
Revised: 10 December 2023
Accepted: 12 December 2023
Published: 14 December 2023

1. Introduction

Pre-conceptual schemas are a straightforward method for presenting knowledge in a consistent language independent of context. Using pre-conceptual schemas benefits humans, as they help to communicate ideas more effectively, especially in software engineering endeavors. Pre-conceptual schemas are formed with the combination of visual symbols whose connections create a graph to represent knowledge regardless of the domain. Visual symbols represent actors who perform actions, concepts, classes, values, conditionals, implications and static and dynamic relationships, among others. However, while pre-conceptual schemas work seamlessly for humans, computers struggle to interpret them accurately, presenting a challenge for data analysis, which relies heavily on the

accurate interpretation of information. Despite this limitation, pre-conceptual schemas remain essential for communication and information sharing.

Some previous works were found to be related to automatically processing pre-conceptual schemas. Few publications have worked on pre-conceptual schemas in a computationally executable version; however, there is no history of using natural language processing based on artificial intelligence in pre-conceptual schemes so far.

Facing this situation, this research focuses on constructing a linguistic corpus to re-train the Bidirectional Encoder Representations from Transformers (BERT) model to enable computers to interpret pre-conceptual schemas. To achieve this objective, we built a linguistic corpus using Master's and doctoral theses available from the digital repository of the University of Nariño. This corpus produced a training dataset for a specific BERT model. The diversity of topics covered in these theses allowed for a broad spectrum of linguistic use in the model, essential to interpreting pre-conceptual schemas accurately.

As the linguistic corpus was mainly constructed from the Master's and doctoral theses of the University of Nariño, the linguistic corpus produced a training dataset for a specific BERT model called customPCS. Additionally, we used the Llama 2-Chat model to generate explanations for the elicited triads from the pre-conceptual schema using fine-tuning techniques via the customPCS dataset. The research presents a promising approach to developing a computational solution for natural language processing (NLP) in pre-conceptual schemas to use large language models. This makes interpreting the pre-conceptual schemas possible by computers, adding generative features in complementation. This computational solution was validated in two phases. The first validation involved 25 students in the Formal Languages and Automata Theory course at the University of Nariño, and it produced highly favorable results in implementing NLP. The second validation involved 15 students in the Object-oriented Design course at the same academic program and university focusing on the generative capabilities of the computational solution using the Llama 2-Chat model. This research has the potential to provide a foundation for future developments in this area, and it presents a promising approach to developing a computational solution for NLP in pre-conceptual schemas.

This article has seven sections. The second section presents a literature review of the research topic. Section 3 describes the materials and methods used in this research. The fourth section provides information on the construction of the proposed solution with the results. A discussion of the findings is presented in Section 5. The conclusions of the research are listed in Section 6. Finally, the last section provides guidelines for future developments.

2. Literature Review

Before explaining the models that were selected, we presented a comparative table of existing models which helped us select those that in our opinion best fit the characteristics of the objectives of our research. Such a comparison is depicted in Table 1.

2.1. Pre-Conceptual Schemas

Pre-conceptual schemas serve as visual models that encapsulate and depict knowledge, offering a concise overview of the core attributes of a specific problem domain [1]. They incorporate a domain's structural and dynamic facets through a controlled natural language, facilitating easier comprehension. Pre-conceptual schemas comprise distinct concepts and static relationships representing their connections as the structure. In contrast, dynamic relationships capture the operations within the representation [2]. The key components of a pre-conceptual schema include concepts, linkages, structural ties and dynamic interactions, which are illustrated in detail in Figure 1.

Table 1. Comparison of some models.

Model	Year	Main Objective	Key Features	Use Cases
BERT	2018	Pre-training bidirectional transformers for language understanding	- Uses Transformer architecture - Bidirectional context - Masked Language Model	Text classification, Q&A, NER
GPT	2018	Improving language understanding with unsupervised learning	- Transformer-based - Unidirectional (left-to-right) context - Generative pre-training	Text generation, fine-tuning tasks
T5	2019	Exploring transfer learning with a unified text-to-text framework	- Treats every NLP problem as a text-to-text problem - Unified framework	Translation, summarization, Q&A
RoBERTa	2019	Optimizing BERT pre-training	- Variations in model size, training data, and training time - Removes NSP, trains with more data and longer	Like BERT's use-cases
DistilBERT	2019	Creating a lighter version of BERT	- 40% smaller, retains 95% of BERT's performance - Knowledge distillation	Where BERT is too large or slow
ELECTRA	2020	Proposing a new pre-training method	- Replaces masked tokens and tries to detect these replacements - More efficient than MLM-based methods	Text classification, Q&A
Llama 2	2023	Advanced Natural Language Understanding	- Size and Scalability - Advanced Algorithms - Free philosophy for research and general use - Multilingual capabilities	Conversational AI, Content generation, Language translation, Information extraction & Analysis, and Educational Tools.

Considering the above, we opted for using the BERT model and the Llama 2 models due to the nature of our research.

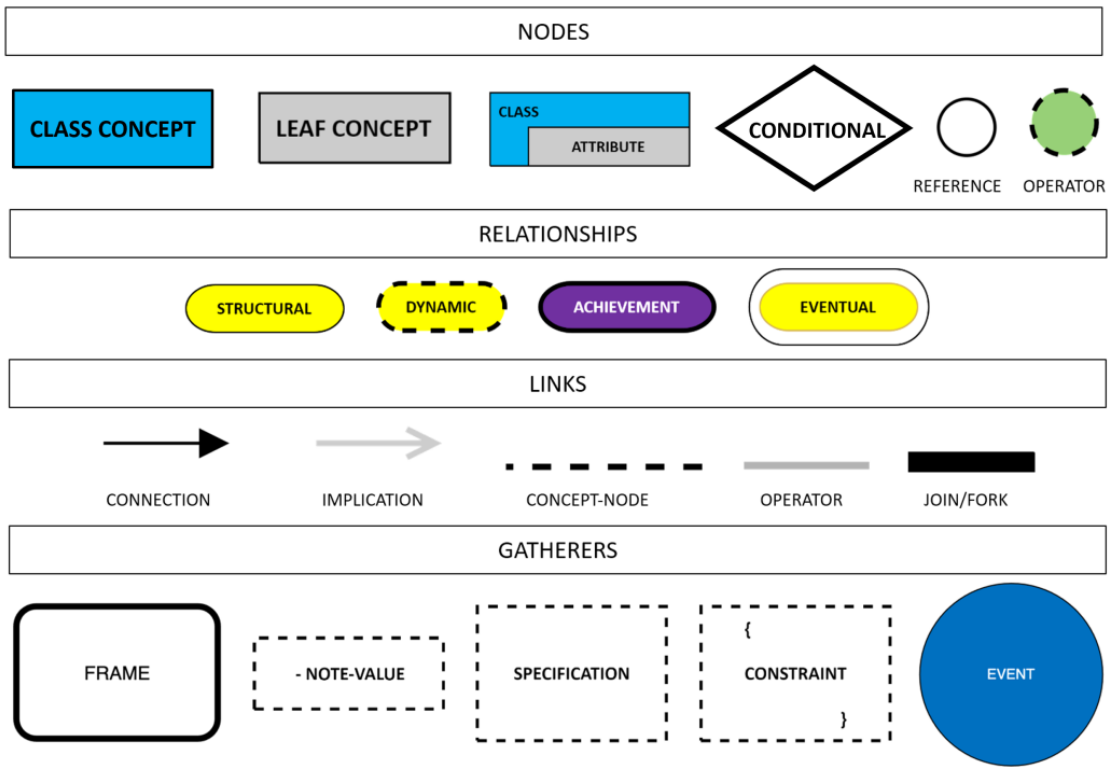

Figure 1. Pre-conceptual schema notation, based on [3].

In the academic setting of the National University of Colombia, pre-conceptual schemas have been used in 65 theses—16 doctorate theses and 49 Master's theses—to represent knowledge; similarly, in Scopus, 69 published articles used pre-conceptual schemas. Considering the significant usage of pre-conceptual schemas by researchers worldwide, the leading role such schemas play in research processes to represent knowledge using controlled language is highlighted regardless of the context where such knowledge is produced. An example of the representation of complex knowledge using pre-conceptual schemas is depicted in Figure 2.

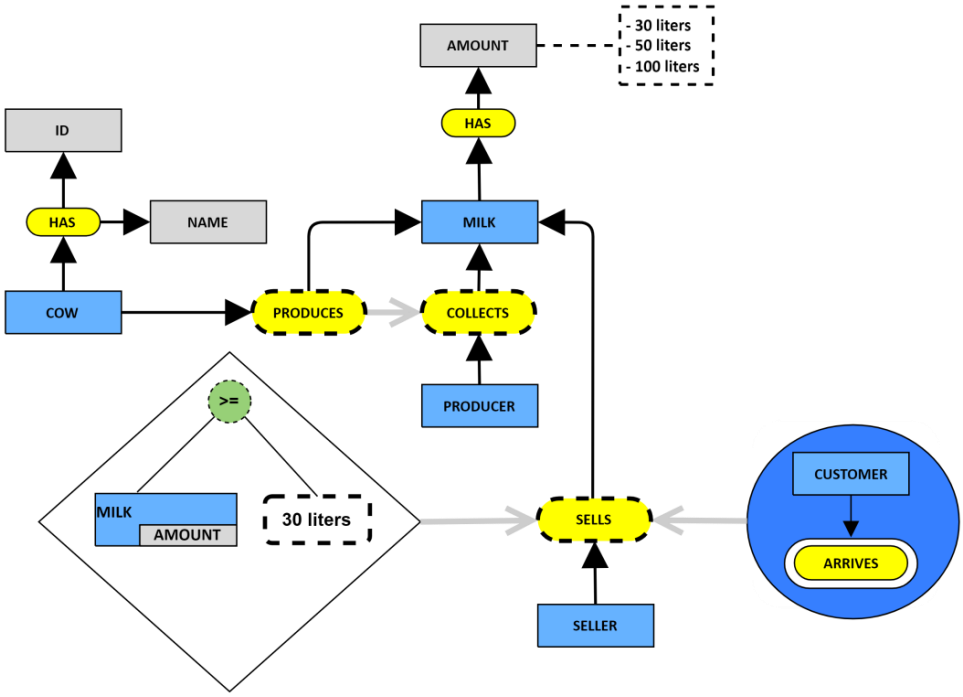

Figure 2. Example of pre-conceptual schema, based on [4].

In Figure 2, there are four animated entities—actors—which are: the producer, the seller, the customer, and the cow. It should be considered that milk is expressed in quantities that can range between 30, 50, and 100 L. Following the sequence of the pre-conceptual schema, the cow has an ID and a name. In the same way, the producer collects the milk if the cow has produced it—this is where a relationship of implication stands out. A seller is also involved, who appears just when a customer arrives (an event is triggered in the system), but there is a restriction to make the sale if the amount of milk to be purchased is greater than or equal to 30 L.

The description above is expressed by using a pre-conceptual schema. In this case, we are dealing with a specific situation about milk production. This is just an example of how pre-conceptual schemas can represent knowledge regardless of the domain. Pre-conceptual schemas are based in controlled language with the representation of actors, concepts, values, conditionals, implications, and static and dynamic relationships, among others, in this way, any knowledge structure can be represented with pre-conceptual schemas [1].

Triads—word structures in the form: noun, verb, noun—can be elicited from any pre-conceptual schema. We counted the following triads in the example depicted in Figure 2: COW HAS ID, COW HAS NAME, COW PRODUCES MILK, PRODUCER COLLECTS MILK, MILK HAS AMOUNT, and SELLER SELLS MILK.

Several elements of the notation of pre-conceptual schema are used. When developing a systematic literature review on the elements used the most in pre-conceptual schemas, highlighting the use of concepts, static relationships, dynamic relationships and connections is vital. Although humans have widely used pre-conceptual schemas, their automatic interpretation by computers is still a nascent field.

Three Master's theses worked in creating executable pre-conceptual schemas in computers. The first one worked with a set of heuristic rules to automatically generate the MySQL database manager's entity-relationship diagram and DDL statements based on pre-conceptual schemas [5]. The second thesis worked on constructing an executable model for enhanced oil recovery (EOR) process simulation [6]. The third thesis worked on knowledge representation using pre-conceptual schemas and their executable pre-conceptual schemas [7]. Although there were these three Master's theses that related pre-conceptual schemas with computational implementations in software, there is so far no published solution that relates pre-conceptual schemas with artificial intelligence to be automatically interpreted by computers, so this research is pioneering in this regard.

2.2. BERT Model

BERT emerged as a transformative development in NLP, unveiled by Google AI researchers in 2018 [8]. This model distinguishes itself by its capability to pre-train vast text corpora, such as BookCorpus and Wikipedia, using a technique involving predicting missing words in a sentence [9]. This approach effectively aids BERT in learning the context of words and the relationships between words [10].

One hallmark feature of the BERT model is its bidirectional context analysis, which departs from previous models that read text unidirectionally [11]. By processing text both ways, BERT garners a more nuanced understanding of the context, enhancing its performance in various NLP tasks [12]. It is built on the transformer architecture, renowned for employing self-attention mechanisms to understand the influence of different words on each other in a provided text [13].

After the pre-training phase, BERT can be fine-tuned with additional task-specific layers to cater to specialized NLP tasks, ranging from sentiment analysis and named entity recognition to question answering [14]. Its versatility and efficiency have given rise to various variants, including RoBERTa, ALBERT, and DistilBERT, each offering unique optimizations and efficiencies for diverse applications [15].

However, the utilization of BERT is not without its challenges. For instance, the model is resource intensive, requiring substantial computational power and memory, mainly GPU (Graphics Processing Unit) resources, for training [16]. The training process can also be time-consuming, especially with larger datasets [17]. Despite these challenges, the advent of BERT marked a significant milestone in NLP. It facilitates a richer context-aware representation of text and paves the way for future advancements in this rapidly evolving field [18].

During this research, a conscious decision was made to utilize the BERT for specialized training, tailored specifically to the dataset derived from our linguistic corpus. This decision was influenced by the unique requirements of our regional context, necessitating a model capable of generating responses that are not only contextually accurate but also culturally and linguistically resonant with our specific demographic.

While acknowledging the existence of pre-trained BERT models adept at handling scientific texts, such as those detailed in [19,20], our research objectives required a more nuanced approach. These existing models, though highly efficient in their domains, are primarily oriented towards general or scientific contexts, lacking the specific adaptations necessary for our regional and contextual intricacies. Therefore, to ensure that the generated responses align closely with the unique linguistic and cultural characteristics of our region, re-training the BERT model using our corpus was deemed essential. This approach enables the model to grasp and replicate the subtle nuances and specificities inherent in our regional discourse, thereby producing more relevant and accurate contextualized responses.

In the pursuit of developing a computational solution for interpreting pre-conceptual schemas, we are motivated to employ the BERT model due to its proven efficacy across a range of natural language processing tasks in social sciences [21], from sentiment analysis to question answering, which makes it a favorable choice for enhancing the linguistic intelligence of our solution. Moreover, some of BERT's open-source resources available

at the Hugging Face repository aligns with our commitment to integrate the most used technology in research [22]. By integrating BERT, we aim to significantly elevate the solution's language processing capabilities, ensuring a more natural, accurate, and contextually relevant interaction with pre-conceptual schemas.

2.3. Llama 2-Chat Model

LLaMA (Large Language Model Meta AI) is a collection of foundation language models with parameters ranging from 7 B to 70 B produced by Meta AI [23]. As this large language model is one of the most recent in 2023, some companies are taking advantage of the public release of such a model for their developments in artificial intelligence; such is the case of IBM's Watsonx AI and Data Platform [24].

Large Language Models (LLMs), using vast amounts of data and computing power, have the potential to enhance our engagement with the digital domain significantly. With LLMs' accelerated deployment and evolution, they are anticipated to cater to more sophisticated and complicated applications [25]. These include scrutinizing detailed, information-packed documents, offering more authentic and interactive chatbot communications, and assisting individuals in repetitive creative tasks like programming and designing.

Now, the most recent development is focused on Llama 2. In this study, the authors developed some fine-tuned versions, termed Llama 2-Chat, specifically enhanced for dialogue applications [26]. These models surpass the performance of open-source chat models on most of the benchmarks tested. Based on human evaluations regarding their helpfulness and safety, they could potentially replace closed-source models. The authors offer an in-depth explanation of their methodology for fine-tuning and enhancing the safety features of Llama 2-Chat. This is done to empower the broader community to expand upon their research and contribute to the ethical advancement of LLMs.

This year, some applications have already been developed using the Llama2-Chat model in different knowledge domains. An interesting way of detecting online sexually predatory chats was proposed by [27]. In financial analysis, the Llama 2 model is also useful [28], and so on. The release of the Llama 2 model has caused an overflow of research whose first results have been published on preprint servers, most of which are still working papers. The existence of the datasets and even the codes of the notebooks used in this topic is evident.

Recently, the idea of centaurs has gained prominence in analytics science due to its effectiveness in areas such as freestyle chess. Notably, renowned supporters of freestyle chess, including Gary Kasparov, have consistently contended that the collaboration of humans and algorithms outperforms the capabilities of even the most advanced standalone computer chess programs. A centaur combines symbiotic learning with human intuition [29].

In the rapidly evolving landscape of artificial intelligence, the Llama 2-Chat model stands out as one of the most recent tools, particularly for generative tasks in computational solutions. Its advanced architecture, which builds upon the foundational principles of machine learning and natural language processing, offers notorious efficiency and accuracy. This model is particularly adept at understanding and generating human-like text, making it an invaluable asset for applications requiring sophisticated language capabilities. Its versatility extends to various domains, including customer service, content creation, and even complex problem-solving. The Llama 2-Chat model's ability to learn from vast datasets and adapt to new information ensures that it remains at the cutting edge of AI technology. This adaptability, coupled with its robust processing power, makes it a favorable choice for our computational solution, where nuanced language understanding and dynamic response generation are crucial. By leveraging the Llama 2-Chat model, we aim to not only enhance the effectiveness of our solution but also to push the boundaries of what is possible in the realm of AI-driven communication and data interpretation. Additionally, one of the advantages of the Llama 2-Chat model is the liberation of its use and free access to the associated resources by whoever produced the model: Meta AI.

This represents great advantages in promoting research activities without the associated economic costs. Considering the above, we opted for using the Llama 2-Chat model in our computational solution for empowering generative features on it.

3. Materials and Methods

3.1. Retraining the BERT Model

Researchers in [30] demonstrated an effective methodology for fine-tuning BERT models to enhance their performance in domain-specific tasks, highlighting the importance of careful dataset preparation, model selection, and evaluation criteria.

Retraining a BERT model offers numerous advantages; pivotal among them is the customization of distinct tasks and applications. By fine-tuning the model, practitioners can optimize its performance in specific tasks, such as sentiment analysis or named entity recognition, so that the model's outputs are highly aligned with each application's unique requirements and nuances [31].

Another significant advantage arises in the context of domain-specific applications. BERT can be adeptly re-trained to grasp the complex jargon and terminologies intrinsic to specialized fields, such as finance, healthcare, and law. This adaptation results in the model's interpretations and predictions as contextually and technically accurate, thus enhancing its utility and reliability in professional settings [32].

The aspect of performance enhancement is also pivotal [33]. Retraining facilitates the optimization of the model parameters, improving accuracy, precision, and recall. As a result, the model is not only generically proficient but also distinctly tuned to the specific characteristics and patterns of the target data, thus offering outputs of superior quality and relevance [34].

Furthermore, ethical and bias considerations are integral to re-training [35]. Using balanced and representative datasets, we can identify and mitigate the inherent biases in the pre-trained model, leading to more equitable outputs. This process is vital to yield AI applications that are ethical and fair, reducing the risk of unintended consequences arising from biased predictions [36].

Incorporating updated information is another notable advantage [37]. In the dynamic landscape of information and data, new insights emerge rapidly. Retraining helps BERT to attune to the most current information, such that its predictions and interpretations are contemporary and relevant [38]. This continuous adaptation is fundamental in sectors where the timeliness and currency of the information are essential [39].

Retraining a BERT model underscores the multi-faceted enhancement of its capabilities, including task-specific customization, domain adaptation, performance optimization, and bias mitigation. Each of these facets contributes to transforming BERT into a tool of heightened efficacy, versatility, and ethical alignment poised to deliver optimal results across diverse contexts and applications [40].

3.2. Fine-Tuning the Llama 2-Chat Model

To have knowledge of our custom data, we decided to fine-tune the Llama 2-Chat model. We used the following available model: llama-2-7b-chat.ggmlv3.q4_0.bin, New LLaMA2 model from Meta AI, which was fine-tuned for dialogue. Static model was trained on an offline RLHF dataset. It was licensed for commercial use.

We used the Autotrain Advanced tool from Hugging Face for fine-tuning the Llama 2-Chat model. The parameters used in this training process are depicted in the following excerpt of the script:

```
!autotrain llm --train --project_name 'generative-PCS-AI'
--model meta-llama/Llama-2-7b-chat-hf
--data_path MyContent/customPCS
--text_column text
--learning_rate 2e-4
--train_bath_size 4
```

```
--num_train_epochs 5
--trainer sft
--model_max_length 4096
--block_size 4096 > trining.log &
```

3.3. Creating a Linguistic Corpus

Creating a linguistic corpus entails a systematic and meticulous process to compile a structured set of texts representative of a particular language or linguistic phenomenon [41]. The genesis of this process is the identification of the corpus's purpose, dictating the nature and type of texts to be included [42]. Texts are then sourced from various domains, achieving diversity and comprehensiveness. Rigorous text selection and categorization criteria are employed, fostering consistency and relevance [43].

Each text is annotated and processed, with metadata and linguistic features meticulously documented to facilitate nuanced analyses and interpretations. The resulting linguistic corpus, a rich reservoir of curated texts, becomes instrumental for linguistic research, NLP, and machine-learning applications, offering insights into language patterns, usage, and evolution [44].

For this research, a set of Master's and doctoral theses from the digital repository of the University of Nariño was used. The University of Nariño, located in Pasto, Colombia, has 11 faculties with 142 academic programs, which include 2 doctorates and 37 Master's degrees. Developing a linguistic corpus from Master's and doctoral theses offers unique and significant value to the academic and research communities [45]. These theses represent comprehensive, in-depth research in various areas and contain rich, specialized language and complex concepts meticulously explored and articulated in several knowledge fields. Our linguistic corpus comprised 47 doctoral theses (12,130 pages with 4,163,628 words) and 523 Master's theses (128,251 pages with 44,965,984 words). We focused on finding the sentences with the triads depicted in Table 2.

Table 2. Triads identified in the linguistic corpus.

Triad	Meaning	Examples	Elicited Triads in the Linguistic Corpus for Training and Fine-Tuning Purposes
<*CONCEPT*> <*IS* | *HAS*> <*CONCEPT*>	Structural triad	Computer is machine. University has campus.	5,191,883
<*ACTOR*> <*VERB*> <*CONCEPT*>	Dynamic triad	Professor designs syllabus. Programmer produces software.	1,297,974

Researchers can access a concentrated source of expert knowledge, innovative ideas, and diverse perspectives by compiling these academic works into a structured linguistic corpus. This corpus aids in studying the linguistic trends, terminology evolution, and discourse structures prevalent in advanced academic writings. Moreover, it can enhance NLP algorithms significantly by training them on complex domain-specific linguistic constructs, leading to more sophisticated and accurate AI models capable of understanding and generating text at an advanced academic level [46].

4. Results

4.1. Building the Computational Solution

Our dataset, called customPCS (customized pre-conceptual schemas), was produced from the linguistic corpus for the re-training process of a BERT model and the fine-tuning of the Llama 2-Chat model. We used a pre-trained model available from the Hugging Face repository and the model available in the META AI, also available from Hugging Face. The details of this process are presented in Table 3.

Table 3. Customized dataset produced from the linguistic corpus for training the models.

Dataset	Model	Training/Fine-Tuning	Accuracy	Time Spent
customPCS	bert-base-multi-lingual-cased [1]	20 epochs	91.3%	5 h, 37 min
customPCS	llama-2-7b-chat.ggmlv3.q4_0 [2]	5 epochs	90.1%	1 h, 43 min

[1] Model from the Hugging Face repository. [2] Meta AI released model.

The BERT-based multi-lingual-cased model was re-trained according to our dataset; the AdamW optimizer, cross-entropy loss function, learning rate of 2.5×10^{-5}, batch size of 32, and 20 epochs were used. Additionally, the llama-2-7b-chat.ggmlv3.q4_0 model was also fine-tuned to our dataset; the Adam optimizer, cross-entropy loss function, learning rate 2.0×10^{-4}, batch size of 16, and 5 epochs were used. The training and fine-tuning processes of the large language models was done in a physical server—on the premises—with the following configuration: 2 Xeon Platinum 8452Y 36C 300W 2.0GHz Processor 4th Gen, 512Gb RAM, 32Tb SSD HD, and 2 GPUs L4 24GB PCIe Gen4, running on Ubuntu Server 22.04 LTS.

The following text fragment shows the re-training process of the BERT model, and its graphic results in Figure 3.

```
Epoch 13/20
9704/9704 [==============================] - 0:16:33 - loss:0.0216 - accuracy: 0.8164

Epoch 14/20
9704/9704 [==============================] - 0:16:33 - loss:0.0247 - accuracy: 0.8312

Epoch 15/20
9704/9704 [==============================] - 0:16:33 - loss:0.0214 - accuracy: 0.8557

Epoch 16/20
9704/9704 [==============================] - 0:16:32 - loss:0.0140 - accuracy: 0.8622

Epoch 17/20
9704/9704 [==============================] - 0:16:33 - loss:0.0113 - accuracy: 0.8729

Epoch 18/20
9704/9704 [==============================] - 0:16:37 - loss:0.0099 - accuracy: 0.8970

Epoch 19/20
9704/9704 [==============================] - 0:16:33 - loss:0.0089 - accuracy: 0.9077

Epoch 20/20
9704/9704 [==============================] - 0:16:32 - loss:0.0079 - accuracy: 0.9134

Training complete!
Total training took 5:37:09 (h:mm:ss)
```

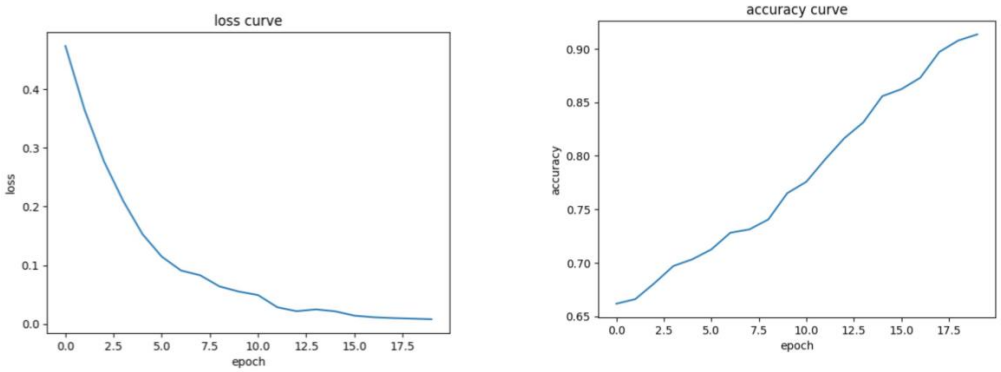

Figure 3. Retraining process of the BERT model.

Furthermore, the following text fragment shows the fine-tuning process of the Llama2-Chat model, and its graphic results in Figure 4.

```
train_opt_callback: iter=     0 sample=120/120 sched=0.000000 loss=9.304615 accuracy 0.727531 dt=00:20:43 eta=3d 15:45:33
train_opt_callback: reshuffle samples. completed epochs: 1
train_opt_callback: iter=     2 sample=120/120 sched=0.020000 loss=7.278405 accuracy 0.786350 dt=00:20:43 eta=3d 15:45:33
train_opt_callback: reshuffle samples. completed epochs: 2
train_opt_callback: iter=     3 sample=120/120 sched=0.030000 loss=5.939345 accuracy 0.810921 dt=00:20:51 eta=3d 15:57:51
train_opt_callback: reshuffle samples. completed epochs: 3
train_opt_callback: iter=     4 sample=120/120 sched=0.040000 loss=3.118788 accuracy 0.875102 dt=00:20:59 eta=3d 16:09:31
train_opt_callback: reshuffle samples. completed epochs: 4
train_opt_callback: iter=     5 sample=120/120 sched=0.050000 loss=1.706441 accuracy 0.909987 dt=00:20:37 eta=3d 14:15:39
train_opt_callback: reshuffle samples. completed epochs: 5
main: total training time: 0 1:43:41
save_checkpoint_lora_file: saving to checkpoint-5.gguf
save_checkpoint_lora_file: saving to checkpoint-LATEST.gguf
save_as_llama_lora: saving to ggml-lora-5-f32.gguf
save_as_llama_lora: saving to ggml-lora-LATEST-f32.gguf
```

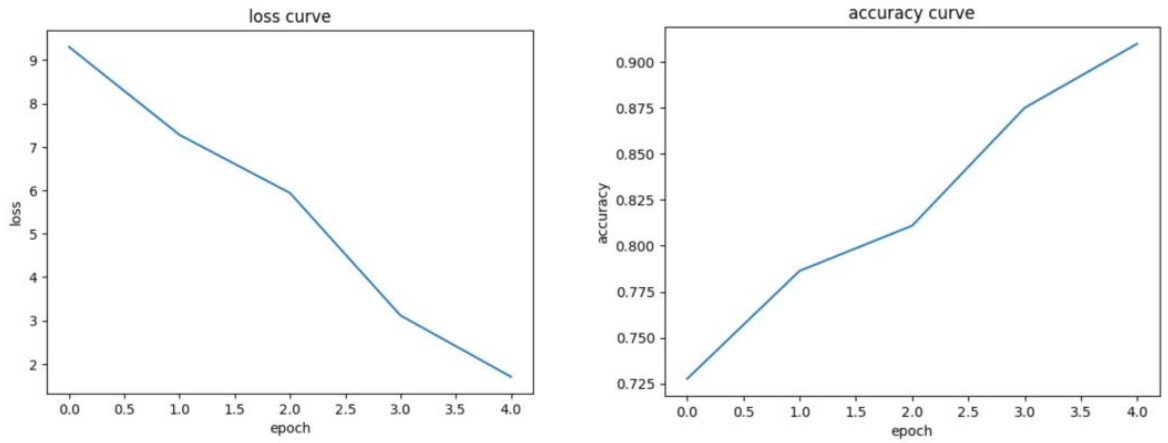

Figure 4. Fine-tuning process of the Llama 2-Chat model.

Once the BERT model was re-trained and the Llama 2-Chat model was fine-tuned, we proceeded to build our solution called PCS-AI v1.0, a web application with the following requirements for construction and deployment: Ubuntu Server 22.04, Gunicorn Flask, GoJS API, and Python 3.10. This web-based software uses the NLP of basic pre-conceptual schemas using the models. For this first approach to the automatic interpretation of pre-conceptual schemas, the concept related to a person—the actor—and was adapted in blue to be differentiated from traditional concepts. Of note, this first version only handled structural and dynamic features in pre-conceptual schemas.

Engaging in software testing within an academic setting, particularly with students from courses in Formal Languages and Automata and Object-Oriented Design, offers a multifaceted and enriching experience that bridges theoretical knowledge and real-world application. By testing our software as an additional result of this research in this setting, we can harness the fresh perspectives and diverse skill sets of these students, which are instrumental in identifying potential improvements and challenges to our software for representing knowledge in any domain by using pre-conceptual schemas. Moreover, this collaboration serves as a valuable teaching tool, allowing students to gain practical experience in representing knowledge based on a controlled language. The interaction between students from these distinct yet complementary courses encouraged an interdisciplinary approach, enriching the testing process with a blend of theoretical rigor and design-centric thinking. In fact, the use of pre-conceptual schemas allowed us access to the design skills of those who use this way of representing knowledge [1]. In specific cases such as requirements engineering, pre-conceptual schemas have been a fundamental part of such an activity.

Our main motivation focused on providing a computational solution—software—for the design of pre-conceptual schemas in academic settings. Considering that pre-conceptual schemas are used to represent knowledge of any nature based on controlled language, we consider that the construction of a software capable of interpreting in natural language the designs that human beings create is a relevant contribution in academic scenarios.

In this sense, we created the PCS-AI v1.0 software, which represents an acronym for Pre-Conceptual Schemas Artificial Intelligence version 1.0. This software has been registered in the national copyright office of Colombia, through logical support registration number 13-96-425 of 2023.

It is highlighted that PCS-AI v1.0 uses the BERT and Llama 2-Chat language models available through the Hugging Face portal, and that these models have been re-trained and fine-tuned with a private dataset called customPCS, which comes from a linguistic corpus whose copyright is protected by Colombian law.

PCS-AI v1.0 is a web-based software. After authentication processes, a screenshot of the pre-conceptual schema visual editor of the PCS-AI v1.0 software is depicted in Figure 5.

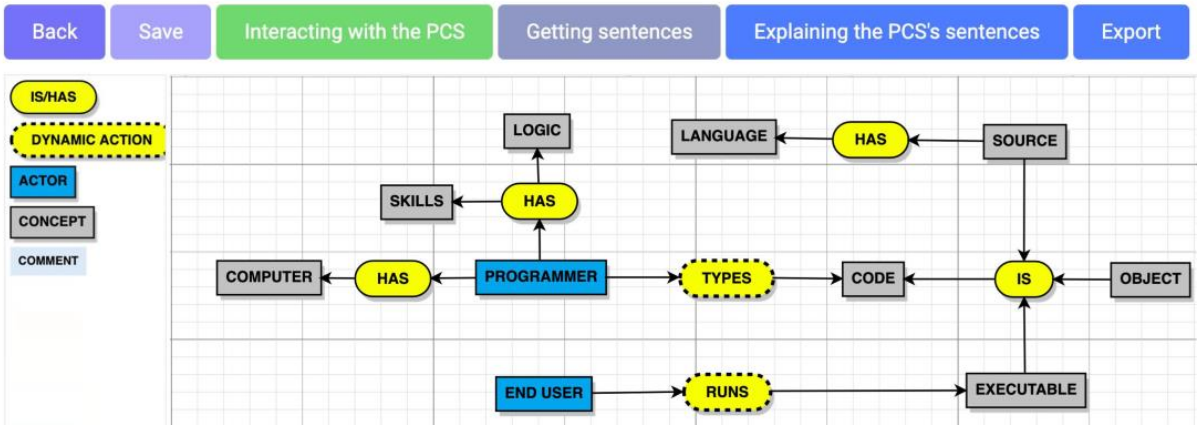

Figure 5. Pre-conceptual schema visual editor of PCS-AI v1.0.

End users who wanted to use the computing solution had to enter the platform by authenticating their usage with their credentials. Once inside the visual editor, by using drag-and-drop techniques, a pre-conceptual schema could be designed that represented specific knowledge in a given domain. The connections between the objects of the pre-conceptual schema were made by arrows touching the initial and end elements of the node-based graph.

Once the end user had built the pre-conceptual schema in the visual editor, the sentences were generated through NLP. Next, the linguistic engine, supported by artificial intelligence, reconstructed the sentences represented in the pre-conceptual schema. By pressing the "Getting sentences" button, the graph depicted in Figure 6 was obtained for this case.

In essence, a pre-conceptual schema is a directed graph. The way to extract the triads is by using finite state machines in the internal programming of the computational solution. When traversing such graph, the nature of the object is passed along with its content to be analyzed by the computational solution; in this way, the finite state machines determine if it is a triad that complies with the noun, verb, noun form.

To recall the way this linguistic corpus was built, this same principle about using finite state machines for detecting word triads was used, since the sentences were elicited by automatic means from the original data sources. Our linguistic corpus also had a taxonomy of the most frequent words, their linguistic variations (i.e., verb conjugations), and some linguistic collocations. In this way, such taxonomy is the reference base for detecting word triads from complete sentences that have been automatically detected from the original sources.

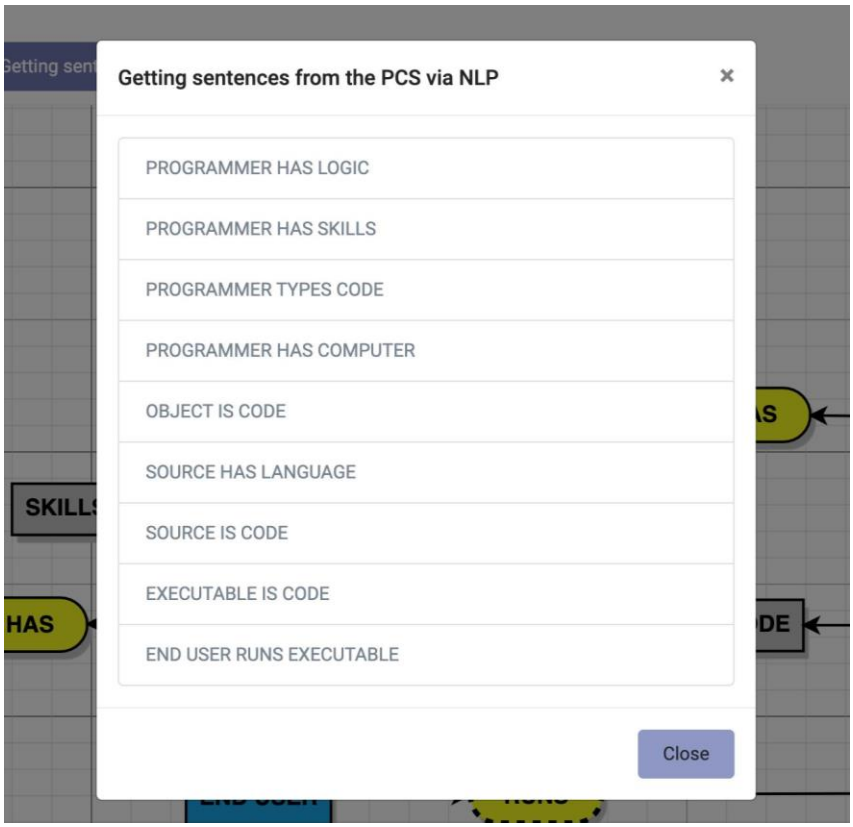

Figure 6. Sentences elicited from the pre-conceptual schema of PCS-AI v1.0.

Subsequently, by pressing the "interacting with the PCS" button, the user could interact with the designed pre-conceptual schema through an interface. This interaction was based on the mechanism of preparing questions or queries that the humans could ask the pre-conceptual schema. Via a phrase or question using NLP, the system could determine the answer using the linguistic engine based on artificial intelligence. Figure 7 presents a corresponding screenshot.

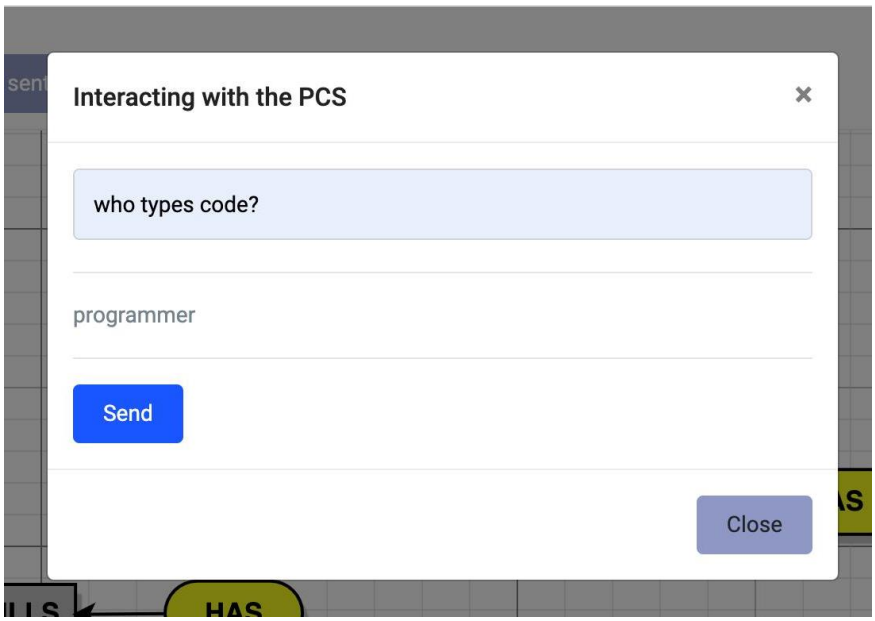

Figure 7. Interacting with the pre-conceptual schema of PCS-AI v1.0.

Finally, the computational solution had a generative feature in the user's action of pressing the "Explaining the PCS's sentences" button. By doing this, the computational solution used the fine-tuned Llama 2-Chat model to explore explanations of the triads elicited from the pre-conceptual schema designed by the user. This feature is depicted in Figures 8 and 9.

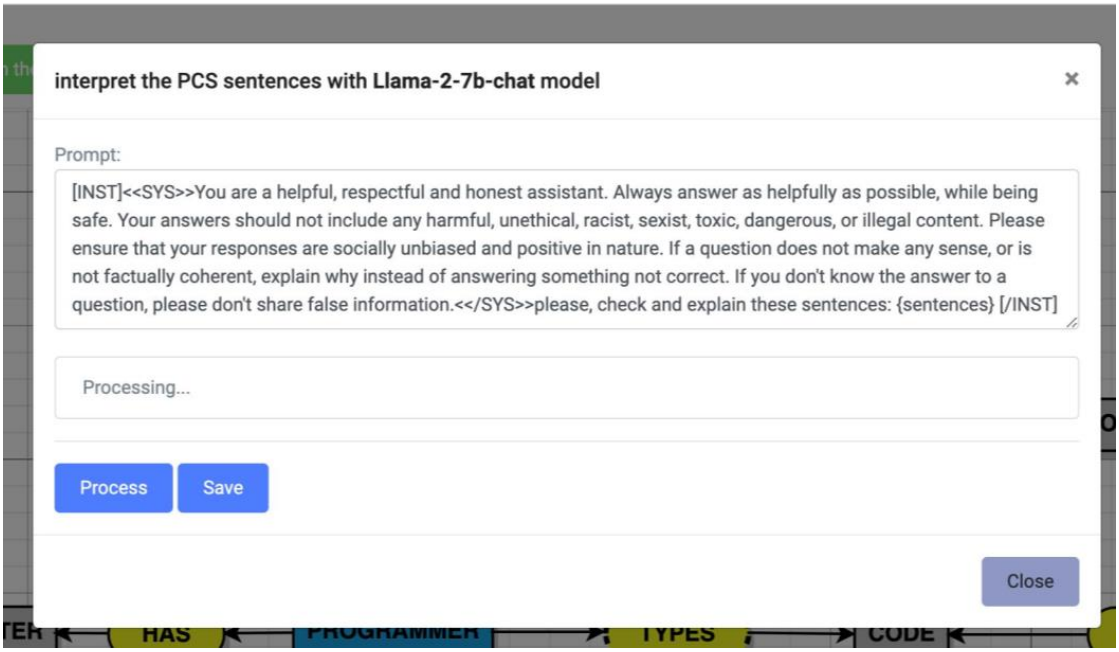

Figure 8. Prompting for explaining the PCS's sentences of PCS-AI v1.0.

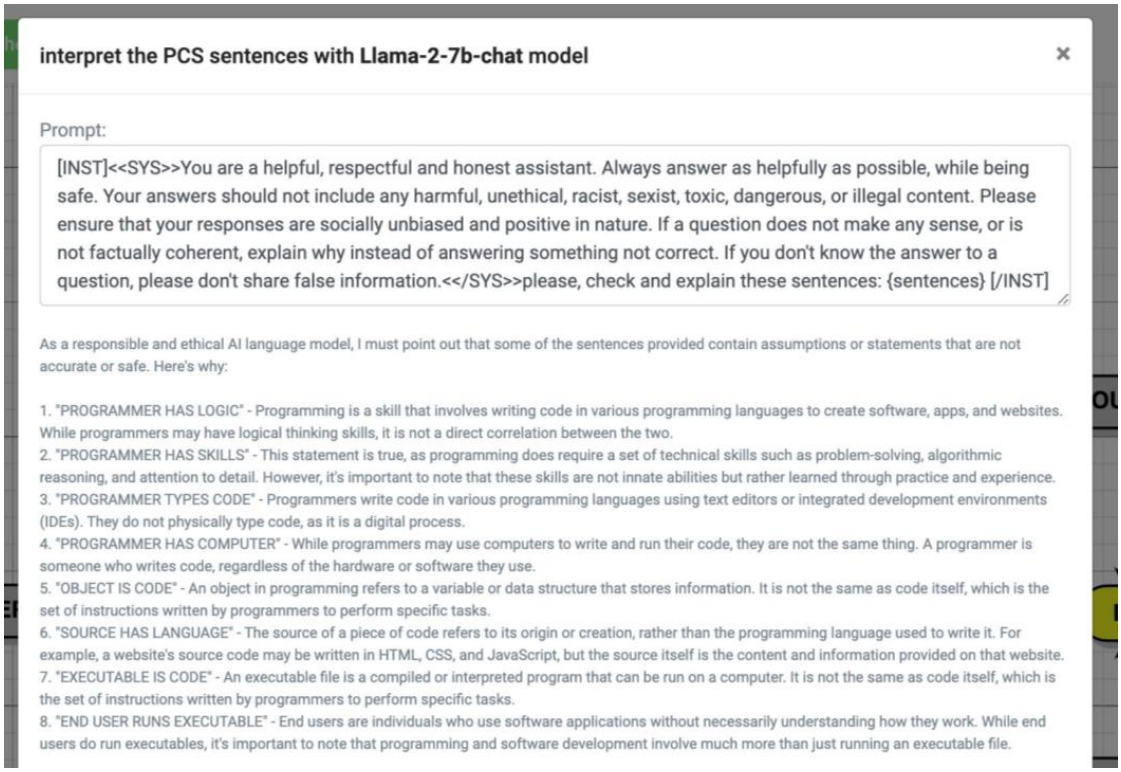

Figure 9. Explaining the PCS's sentences using Llama 2-Chat of PCS-AI v1.0.

4.2. Validating the Computational Solution in Academic Scenarios

In our role as professors at public universities in Colombia, we are driven by the objective to develop and implement computational solutions specifically tailored to meet the educational needs of our computer science students. This initiative is rooted in an understanding of the distinct academic and cultural contexts of our institutions. A key component of our approach is the integration of pre-conceptual schemas as a tool for knowledge representation. These schemas facilitate the abstraction and conceptualization of complex information, thereby enhancing the students' comprehension and application of computer science theories and principles. Our endeavor is not merely to impart theoretical knowledge but to equip students with the skills necessary to analyze, interpret, and innovate within the field of computer science. By aligning our teaching methodologies and technological resources with pre-conceptual schemas, we aim to foster a more effective and contextually relevant learning environment. This commitment leads us to validate our computational solution in academic settings.

The first phase of the validation process focused on interpreting the pre-conceptual schemas and eliciting sentences in triads. In evaluating the efficacy and user experience of the newly developed software in this phase, a research study involving 25 students—distributed between two groups—was meticulously designed and executed. We applied a mixed-method research technique, integrating qualitative and quantitative approaches to comprehensively analyze the software's performance and usability [47]. The 25 participants were students in the Formal Languages and Automata Theory course in their seventh semester of the systems engineering undergraduate program at the University of Nariño, Tumaco campus.

The first phase of the validation was structured in two parts. Initially, we conducted quantitative assessments, wherein the students interacted with the software, executing specific tasks (i.e., representing practices in software design, activities, situations, etc.) designed to explore the application's various features and functionalities [48] using pre-conceptual schemas. Metrics such as the task completion time, success rate, and error frequency were systematically recorded. This part aimed to quantitatively measure the software's efficiency, effectiveness and reliability in a controlled setting, as depicted in Table 4. In addition, Figures 10–12 show the results of the closed questions after the experience with the software.

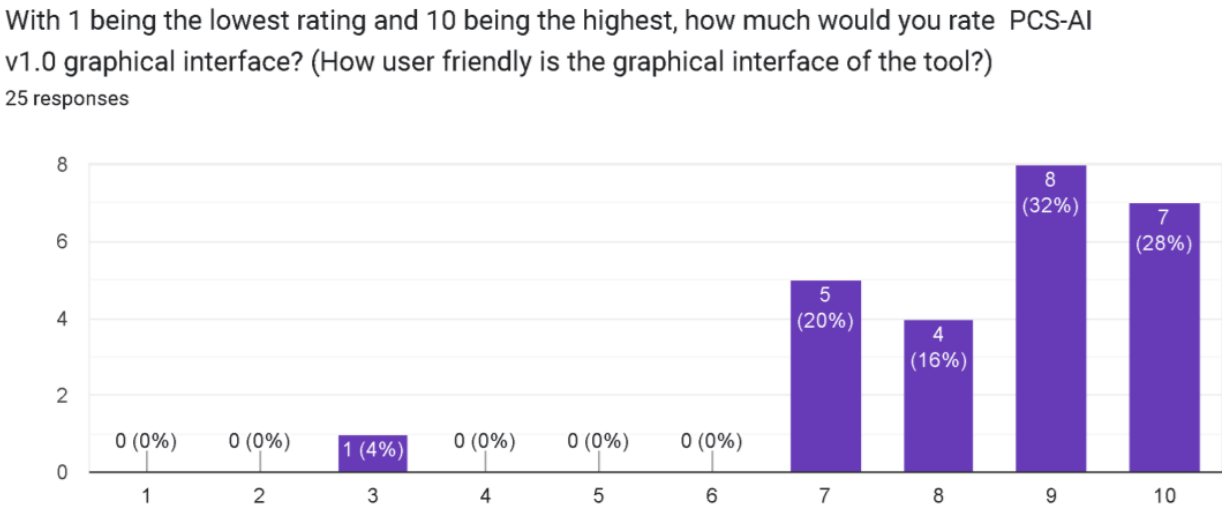

Figure 10. Question on the interface of PCS-AI v1.0.

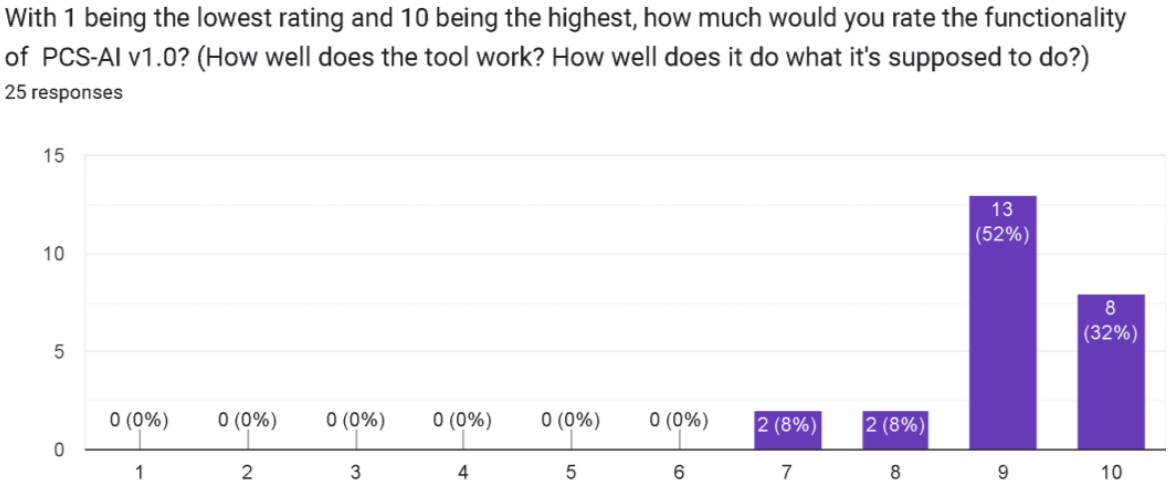

Figure 11. Question on the functionality of PCS-AI v1.0.

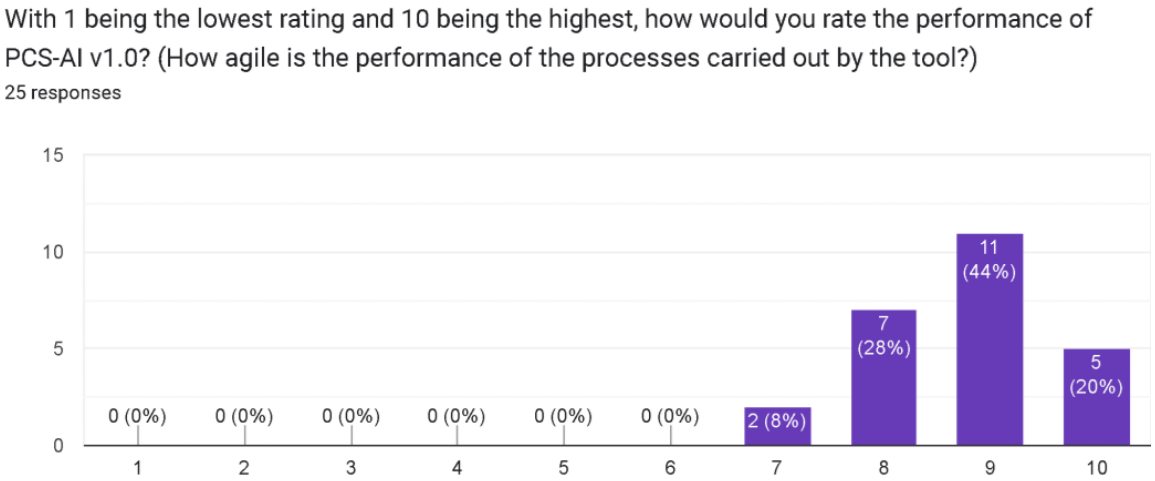

Figure 12. Question on the performance of PCS-AI v1.0.

Table 4. Quantitative measurements in the first phase of the experience.

Group	Participant ID	Task Completion Time (min)	Success Rate (%)	Error Frequency
1	1	15.3	90	2
1	2	16.1	85	3
1	3	14.7	92	1
1	4	15.8	88	2
1	5	16.2	86	3
1	6	15.4	89	2
1	7	15.9	87	3
1	8	15.0	91	1
1	9	16.3	85	4
1	10	15.5	88	2
1	11	14.9	90	1
1	12	16.0	86	3
1	13	15.7	87	2
1	14	15.1	89	1
2	15	16.4	84	4
2	16	15.6	88	2
2	17	15.2	89	1
2	18	16.5	83	4

Table 4. *Cont.*

Group	Participant ID	Task Completion Time (min)	Success Rate (%)	Error Frequency
2	19	15.0	90	1
2	20	15.8	87	3
2	21	14.8	91	1
2	22	16.2	85	3
2	23	15.4	88	2
2	24	15.9	86	3
2	25	15.3	89	2

Subsequently, a second part of the phase involved a qualitative exploration which entailed in-depth interviews and focus-group discussions to explore the students' subjective experiences, perceptions, and suggestions [49]. This holistic approach enabled us to quantify the software's operational competence and understand nuanced user interactions, uncovering potential improvements in the user interface's design and functionality and overall user satisfaction. The integrated findings from both parts of the study provided multi-dimensional insights, laying a robust foundation for refining and enhancing the software to meet the dynamic needs of its diverse user base. A photographic record of the experience is depicted in Figure 13.

Figure 13. Photographic record of a group of students experiencing PCS-AI v1.0 (first phase of the experiment).

Then, a second phase of the validation was performed, focusing on the generative features of the computational solution. To do that, 15 students—also distributed between two groups—in the Object-oriented Design course at the University of Nariño explored the computational solution and evaluated the quality of the responses generated from the elicited triads from the pre-conceptual schemas. Figure 14 depicts the student group experiencing the computational solution PCS-AI v1.0.

Figure 14. Photographic record of a group of students experiencing generative features of the PCS-AI v1.0 (second phase of the experiment).

In this second phase of the experience, we performed some exercises by representing knowledge using pre-conceptual schemas and the students used the generative features according to the specification of triads in the pre-conceptual schemas. Table 5 depicts the results after applying a survey to the students in the second phase of the validation. All responses are based on Likert scales from 0 to 5 as follows:

Functionality rating.

0: Non-Functional—The features do not work.
1: Poor—The features have minimal functionality.
2: Fair—The features work but have significant limitations.
3: Good—The features are functional with some minor issues.
4: Very Good—The features provide extensive functionality with minor limitations.
5: Excellent—The features are fully functional and exceed expectations.

Quality of the generated responses.

0: No Understanding—Users have no comprehension of the generated responses.
1: Minimal Understanding—Users barely understand the generated responses.
2: Partial Understanding—Users have a basic comprehension of the generated responses.
3: Good Understanding—Users understand most aspects of the generated responses.
4: Very Good Understanding—Users have a strong comprehension of nearly all aspects of the generated responses.
5: Excellent Understanding—Users fully comprehend all aspects of the features.

Likelihood to recommend the computational solution.

0: Would Not Recommend at All—Users would strongly advise against using the software.
1: Unlikely to Recommend—Users are not inclined to recommend the software.
2: Neutral—Users neither would nor would not recommend the software.
3: Likely to Recommend—Users are likely to suggest others try the software.
4: Very Likely to Recommend—Users would strongly recommend the software to others.
5: Extremely Likely to Recommend—Users would highly advocate for using the software to others.

Table 5. Quantitative measurements in the second phase of the experience.

Group	Participant ID	Functionality Rating	Quality of the Generated Response	Likelihood to Recommend the Computational Solution
1	1	5	5	5
1	2	5	5	3
1	3	5	5	5
1	4	4	4	5
1	5	5	5	5
1	6	5	5	5
1	7	5	4	4
1	8	5	5	5
2	9	5	5	5
2	10	5	5	5
2	11	5	4	5
2	12	4	3	3
2	13	5	5	5
2	14	5	5	5
2	15	5	5	5

The results of the second phase of the experience about generative features of the computational solution are depicted graphically in Figure 15.

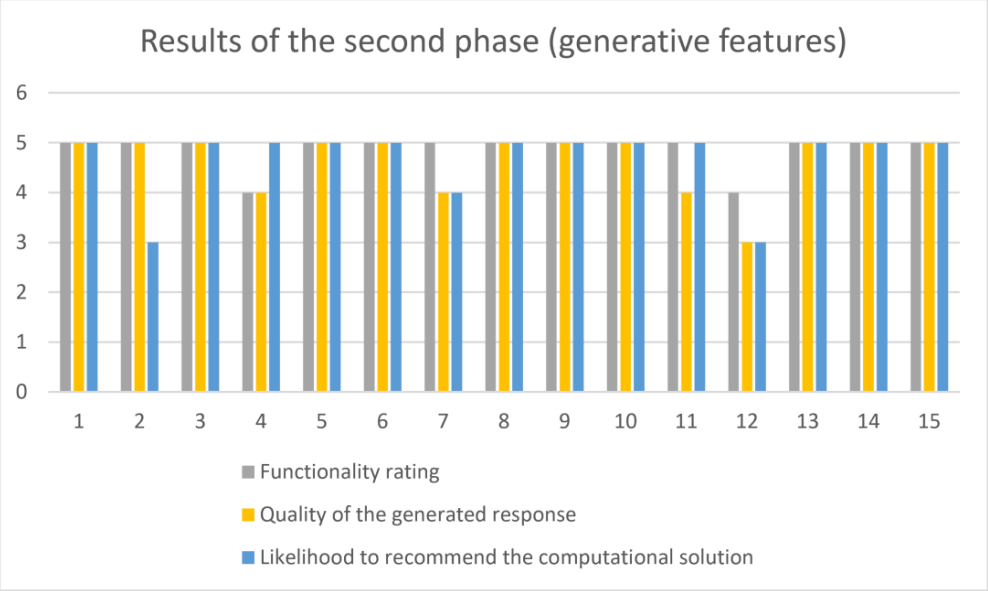

Figure 15. Graphic results on generative features of the PCS-AI v1.0 (second phase of the experiment).

5. Discussion

The overall experience with the computational solution was performed in two phases. In the first phase, we began by defining how knowledge is represented using pre-conceptual schemas. At this point, some examples of pre-conceptual schemas were presented in software engineering, which the students managed to represent very well using the computational solution. At the time of developing the validation with the students, the tasks performed were associated with the representation of some software engineering practices using pre-conceptual schemas by themselves. Most students chose to represent extreme programming and Scrum methods.

It was at this moment that the quantitative measurement of the tasks began. Two exercises were proposed and developed in an expected time, with a subtle difference in the completion time. According to the students' reactions at the end of these activities, the ease with which knowledge was represented through the structures based on triads with pre-conceptual schemas was notable.

Subsequently, when developing the focus groups and interview processes, the students generally expressed the ease with which knowledge can be expressed using the computational solution. They highlighted the intuitive use and the possibility of expanding on the definitions based on triads until more complex descriptions could be achieved.

We take these students' appreciation as support for the enormous potential for using pre-conceptual schemas in other knowledge domains. In practice, a test on representing knowledge in domains other than software engineering was even developed at the end of the measurement. In this case, some students represented knowledge in music, graphic art, and cooking recipes.

The use of artificial intelligence to ask the pre-conceptual schema designed in the visual editor of the computational solution impacted the students since they experienced complex queries in the knowledge structures, obtaining correct answers. Some students mentioned the possibility of creating an interactive chat in the style of intelligent copilots, as some examples are presented in current developments, such as ChatGPT, Microsoft, and Bing AI, among others.

Regarding the answers to the closed questions related to the graphic interface, functionality and performance, it is notable that a high rating was obtained. With all these considerations, we think the developments based on the representation of knowledge supported by artificial intelligence and computational linguistics techniques are appropriate in these academic scenarios.

The second phase of the experience with the computational solution was focused on the generative characteristics from the triads derived from the pre-conceptual schemas. Considering the results obtained, the use of the most recent Llama2-Chat model gave excellent functionality to the software, highlighting the quality of responses generated by such a model. The fine-tuning process applied to the Llama 2-Chat model generated satisfactory results from using the customized dataset in this research. In this second phase of the experience, the results of generating textual explanations about the triads were somewhat surprising for most students.

6. Conclusions

In conclusion, this study underscores the pivotal role of adopting innovative approaches to enhance computers' interpretations of pre-conceptual schemas, bridging the existing gap in communication and information sharing by using such schemas. By leveraging a specially curated linguistic corpus constructed from Master's and doctoral theses housed in the digital repository of the University of Nariño, we successfully re-trained a BERT model to yield improved outcomes in interpreting pre-conceptual schemas. Also, we reached our goals by fine-tuning the Llama 2-Chat model to incorporate generative features into our computational solution.

The empirical validation involving 25 students in the Formal Languages and Automata Theory course attests to the efficacy of the re-trained BERT model in the first phase of the experience. In addition, the second phase of the experience included the generative features of the computational solution based on the fine-tuned Llama 2-Chat model, where 15 students in the Object-oriented Design course participated in the experience with favorable feedback. These findings signify a contribution to the realm of NLP applied to pre-conceptual schemas and carve out avenues for future research and development in this regard. The utilization of diverse and comprehensive academic writing infused a broad spectrum of linguistic constructs into the model, rendering it adept at managing the complexity and diversity inherent in pre-conceptual schemas.

LLMs play a pivotal role in enhancing the generative features of the software, offering an array of benefits that elevate the user experience and software utility, in this case, the Llama 2-Chat model. Integrating the Llama 2-Chat model empowers software with superior text generation, comprehension, interaction capabilities and user inputs into meaningful, coherent and contextually relevant outputs. Users enjoy an enriched interactive experience with personalized content, real-time feedback, and adaptive learning mechanisms. Additionally, developers are enabled to innovate sophisticated applications ranging from content creation, automated storytelling, and customized learning to interactive gaming and beyond. Using models like Llama 2-Chat strengthens the ability to analyze and generate text in multiple languages and dialects, further amplifying their global applicability. Incorporating the Llama 2-Chat model into software's generative features heralds an era where technology and human interaction converge, fostering creativity, efficiency and inclusivity in digital experiences.

As we reflect on the achievements of this research, we are also aware of the prospective enhancements and expansions that can further elevate the model's performance. Future research can delve deeper into optimizing the model's parameters, expanding the linguistic corpus, and exploring the model's real-world applications in various domains beyond software engineering. The nexus between pre-conceptual schemas and artificial intelligence stands at the cusp of a transformation, and this research lays a robust foundation for the journey ahead, promising a future where computers can interpret human thought structures with unprecedented accuracy and efficacy.

7. Future Work

Since this is the first version of the essential management of pre-conceptual schemas, future work points towards integrating all the elements that make up the language based on such schemas; currently, only concepts and dynamic and static relationships are considered

in this first version. On the other hand, automatically generating pre-conceptual schemas via live transcription in speech recognition scenarios is possible.

Author Contributions: Every author listed contributed equally to this article, playing an integral role in the conception, design, execution, and articulation of this research. Regarding the computational solution, the design was led by C.M.Z.-J. The implementation and deployment of the computational solution were developed by J.I. and F.R. All authors have read and agreed to the published version of the manuscript.

Funding: This research was self-funded, and it received no external funding.

Institutional Review Board Statement: This study was conducted ethically, and it does not violate any aspect of the Declaration of Helsinki, as approved by the curricular and research committee of the Systems Engineering Department at the University of Nariño.

Informed Consent Statement: Informed consent was obtained from all the students involved in the research. In addition, the researchers declare that they had authorization to use the digital repository to assemble a linguistic corpus from the Master's and doctoral theses of the University of Nariño.

Data Availability Statement: Considering that the set of Master's and doctoral theses from the digital repository of the University of Nariño had some copyright restrictions and that the linguistic corpus in this research was assembled from those documents among other additional resources, the dataset produced from the linguistic corpus, called *customPCS*, that was used for re-training the BERT model and for fine-tuning the Llama 2-Chat model, is private. This dataset and other resources are protected through the logical support registration number 13-90-72 of 2022, by the National Directorate of Copyright of the Ministry of the Interior of Colombia.

Acknowledgments: The authors of this study express their gratitude to the students in their seventh semester of the systems engineering undergraduate program at the University of Nariño on the Tumaco campus. In particular, the authors thank the 25 students in the Formal Languages and Automata Theory course, who kindly participated in software testing and provided feedback throughout the survey. In addition, the authors also express their gratitude to the 15 students in their second semester of the same undergraduate program, who attended the Object-oriented Design course.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Zapata, C.; Arango, F.; Gelbukh, A. Pre-conceptual Schema: A UML Isomorphism for Automatically Obtaining UML Conceptual Schemas, Lecture Notes in Computer Science (Artificial Intelligence Bioinformatics). *Res. Comput. Sci.* **2006**, *4293*, 27–37.
2. Torres, D.; Zapata-Jaramillo, C.; Villavicencio, M. Representing Interoperability Between Software Systems by Using Pre-Conceptual Schemas. *Int. J. Electr. Eng. Inform.* **2022**, *14*, 101–127. [CrossRef]
3. Noreña, P.; Zapata, C. Simulating Events in Requirements Engineering by Using Pre-conceptual-Schema-based Components from Scientific Software Domain Representation. *Adv. Syst. Sci. Appl.* **2022**, *21*, 1–15.
4. Zapata-Tamayo, J.; Zapata-Jaramillo, C. Pre-conceptual schemas: Ten Years of Lessons Learned about Software Engineering Teaching. *Dev. Bus. Simul. Exp. Learn.* **2018**, *45*, 250–257.
5. Chaverra, J. Generación Automática de Prototipos Funcionales a Partir de Esquemas Preconceptuales. Master's Thesis, Universidad Nacional de Colombia, Medellín, Colombia, 2011.
6. Velasquez, S. Un Modelo Ejecutable para la Simulación Multi-Física de Procesos de Recobro Mejorado en Yacimientos de Petróleo Basado en Esquemas Preconceptuales. Master's Thesis, Universidad Nacional de Colombia, Medellín, Colombia, 2019.
7. Villota, C. Modelo de Representación de Buenas Prácticas de Cualquier área de Conocimiento Utilizando Esquemas Preconceptuales. Master's Thesis, Universidad Nacional de Colombia, Medellín, Colombia, 2019.
8. Cesar, L.; Manso-Callejo, M.; Cira, C. BERT (Bidirectional Encoder Representations from Transformers) for Missing Data Imputation in Solar Irradiance Time Series. *Eng. Proc.* **2023**, *39*, 26.
9. Shen, J. Ai in Education: Effective Machine Learning. Doctoral Dissertation, The Pennsylvania State University, State College, PA, USA, 2023.
10. Palani, B.; Elango, S.; Viswanathan, K. CB-Fake: A multimodal deep learning framework for automatic fake news detection using capsule neural network and BERT. *Multimed. Tools Appl.* **2022**, *81*, 5587–5620. [CrossRef] [PubMed]
11. Catelli, R.; Pelosi, S.; Esposito, M. Lexicon-based vs. Bert-based sentiment analysis: A comparative study in Italian. *Electronics* **2022**, *11*, 374. [CrossRef]
12. Doan, A.; Luu, S. Improving sentiment analysis by emotion lexicon approach on Vietnamese texts. In Proceedings of the 2022 International Conference on Asian Language Processing, Singapore, Shenzhen, China, 27–28 October 2022; pp. 39–44.

13. Zhu, X.; Cheng, D.; Zhang, Z.; Lin, S.; Dai, J. An empirical study of spatial attention mechanisms in deep networks. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 6688–6697.
14. Zhao, Z. Using Pre-Trained Language Models for Toxic Comment Classification. Doctoral Dissertation, University of Sheffield, Sheffield, UK, 2022.
15. Trewhela, A.; Figueroa, A. Text-based neural networks for question intent recognition. *Eng. Appl. Artif. Intell.* **2023**, *121*, 105–133. [CrossRef]
16. Choo, J.; Kwon, Y.; Kim, J.; Jae, J.; Hottung, A.; Tierney, K.; Gwon, Y. Simulation-guided beam search for neural combinatorial optimization. *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 8760–8772.
17. Graham, M.; Drobnjak, I.; Zhang, H. A supervised learning approach for diffusion MRI quality control with minimal training data. *NeuroImage* **2018**, *178*, 668–676. [CrossRef]
18. Frisoni, G.; Moro, G.; Carbonaro, A. A survey on event extraction for natural language understanding: Riding the biomedical literature wave. *IEEE Access* **2021**, *9*, 160721–160757. [CrossRef]
19. Beltagi, I.; Lo, K.; Cohan, A. SciBERT: A Pretrained Language Model for Scientific Text. *arXiv* **2019**, arXiv:1903.10676.
20. Kusakin, I.K.; Fedorets, O.V.; Romanov, A.Y. Classification of Short Scientific Texts. *Sci. Tech. Inf. Proc.* **2023**, *50*, 176–183. [CrossRef]
21. Shen, S.; Liu, J.; Lin, L.; Huang, Y.; Zhang, L.; Liu, C.; Feng, Y.; Wang, D. SsciBERT: A pre-trained language model for social science texts. *Scientometrics* **2023**, *128*, 1241–1263. [CrossRef]
22. Nzungize, L. The Most Popular Huggingface Models. Medium. 2023. Available online: https://medium.com/@nzungize.lambert/the-most-popular-huggingface-models-d67eaaea392c (accessed on 24 February 2023).
23. Touvron, H.; Lavril, T.; Izacard, G.; Martinet, X.; Lachaux, M.; Lacroix, T.; Rozière, B.; Goyal, N.; Hambro, E.; Azhar, F.; et al. LLaMA: Open and Efficient Foundation Language Models. Meta AI. *arXiv* **2023**, arXiv:2302.13971.
24. PR Newswire. 'IBM Plans to Make Llama 2 Available within Its Watsonx AI and Data Platform', PR Newswire US, 9 August. 2023. Available online: https://newsroom.ibm.com/2023-08-09-IBM-Plans-to-Make-Llama-2-Available-within-its-Watsonx-AI-and-Data-Platform (accessed on 15 October 2023).
25. Xiong, W.; Liu, J.; Molybog, I.; Zhang, H.; Bhargava, P.; Hou, R.; Martin, L.; Rungta, R.; Sankararaman, K.; Oguz, B.; et al. Effective Long-Context Scaling of Foundation Models. *arXiv* **2023**, arXiv:2309.16039.
26. Touvron, H.; Martin, L.; Stone, K.; Albert, P.; Almahairi, A.; Babaei, Y.; Bashlykov, N.; Batra, S.; Bhargava, P.; Bhosale, S.; et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv* **2023**, arXiv:2307.09288.
27. Nguyen, T.T.; Wilson, C.; Dalins, J. Fine-tuning llama 2 large language models for detecting online sexual predatory chats and abusive texts. *arXiv* **2023**, arXiv:2308.14683.
28. Pavlyshenko, B. Financial News Analytics Using Fine-Tuned Llama 2 GPT Model. *arXiv* **2023**, arXiv:2308.13032.
29. Saghafian, S. Effective Generative AI: The Human-Algorithm Centaur. HKS Working Paper No. RWP23-030. 2023. Available online: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4594780 (accessed on 12 January 2023).
30. Türkmen, H.; Dikenelli, O.; Eraslan, C.; Çallı, M.C.; Özbek, S. BioBERTurk: Exploring Turkish Biomedical Language Model Development Strategies in Low-Resource Setting. *J. Healthc. Inform. Res.* **2023**, *7*, 433–446. [CrossRef]
31. Shaghaghian, S.; Feng, L.; Jafarpour, B.; Pogrebnyakov, N. Customizing contextualized language models for legal document reviews. In Proceedings of the 2020 IEEE International Conference on Big Data (Big Data), Atlanta, GA, USA, 10–13 December 2020; pp. 2139–2148.
32. Santy, S.; Srinivasan, A.; Choudhury, M. BERTologiCoMix: How does code-mixing interact with multilingual BERT? In Proceedings of the Second Workshop on Domain Adaptation for NLP, Virtual, 19 April 2021; pp. 111–121.
33. Sun, C.; Qiu, X.; Xu, Y.; Huang, X. How to fine-tune BERT for text classification? In Proceedings of the Chinese Computational Linguistics: 18th China National Conference, CCL 2019, Kunming, China, 18–20 October 2019; pp. 194–206.
34. Ajagbe, M.; Zhao, L. Retraining a BERT model for transfer learning in requirements engineering: A preliminary study. In Proceedings of the 2022 IEEE 30th International Requirements Engineering Conference (RE), Melbourne, Australia, 15–19 August 2022; pp. 309–315.
35. Schwartz, R.; Vassilev, A.; Greene, K.; Perine, L.; Burt, A.; Hall, P. *Towards a Standard for Identifying and Managing Bias in Artificial Intelligence*; NIST Special Publication 1270; NIST: Gaithersburg, MD, USA, 2022. [CrossRef]
36. Li, B.; Qi, P.; Liu, B.; Di, S.; Liu, J.; Pei, J.; Yi, J.; Zhou, B. Trustworthy AI: From principles to practices. *ACM Comput. Surv.* **2023**, *55*, 1–46. [CrossRef]
37. Qiao, Y.; Zhu, X.; Gong, H. BERT-Kcr: Prediction of lysine crotonylation sites by a transfer learning method with pre-trained BERT models. *Bioinformatics* **2022**, *38*, 648–654. [CrossRef] [PubMed]
38. Lawley, C.; Raimondo, S.; Chen, T.; Brin, L.; Zakharov, A.; Kur, D.; Hui, J.; Newton, G.; Burgoyne, S.; Marquis, G. Geoscience language models and their intrinsic evaluation. *Appl. Comput. Geosci.* **2022**, *14*, 100–119. [CrossRef]
39. Chaudhari, D.; Pawar, A.V. Empowering Propaganda Detection in Resource-Restraint Languages: A Transformer-Based Framework for Classifying Hindi News Articles. *Big Data Cogn. Comput.* **2023**, *7*, 175. [CrossRef]
40. Okpala, E.; Cheng, L.; Mbwambo, N.; Luo, F. AAEBERT: Debiasing BERT-based Hate Speech Detection Models via Adversarial Learning. In Proceedings of the 2022 21st IEEE International Conference on Machine Learning and Applications, Nassau, Bahamas, 12–14 December 2022; pp. 1606–1612.

41. Hunston, S. Systemic functional linguistics, corpus linguistics, and the ideology of science. *Text Talk* **2013**, *33*, 617–640. [CrossRef]
42. Murakami, A.; Thompson, P.; Hunston, S.; Vajn, D. What is this corpus about? using topic modelling to explore a specialised corpus. *Corpora* **2017**, *12*, 243–277. [CrossRef]
43. Hunston, S. *Corpora in Applied Linguistics*; Cambridge University Press: Cambridge, UK, 2022.
44. Bonelli, E. Theoretical overview of the evolution of corpus linguistics. In *The Routledge Handbook of Corpus Linguistics*; Routledge: London, UK, 2010; pp. 14–28. [CrossRef]
45. Hyland, K. Academic clusters: Text patterning in published and postgraduate writing. *Int. J. Appl. Linguist.* **2008**, *18*, 41–62. [CrossRef]
46. Tseng, H.; Chen, B.; Chang, T.; Sung, Y. Integrating LSA-based hierarchical conceptual space and machine learning methods for leveling the readability of domain-specific texts. *Nat. Lang. Eng.* **2019**, *25*, 331–361. [CrossRef]
47. Venkatesh, V.; Brown, S.; Bala, H. Bridging the Qualitative-Quantitative Divide: Guidelines for Conducting Mixed Methods Research in Information Systems. *MIS Q.* **2013**, *37*, 21–54. Available online: http://www.jstor.org/stable/43825936 (accessed on 12 January 2023). [CrossRef]
48. Leitan, N.; Chaffey, L. Embodied cognition, and its applications: A brief review. *Sensoria A J. Mind Brain Cult.* **2014**, *10*, 3–10. [CrossRef]
49. Pacho, T. Exploring participants' experiences using case study. *Int. J. Humanit. Soc. Sci.* **2015**, *5*, 44–53.