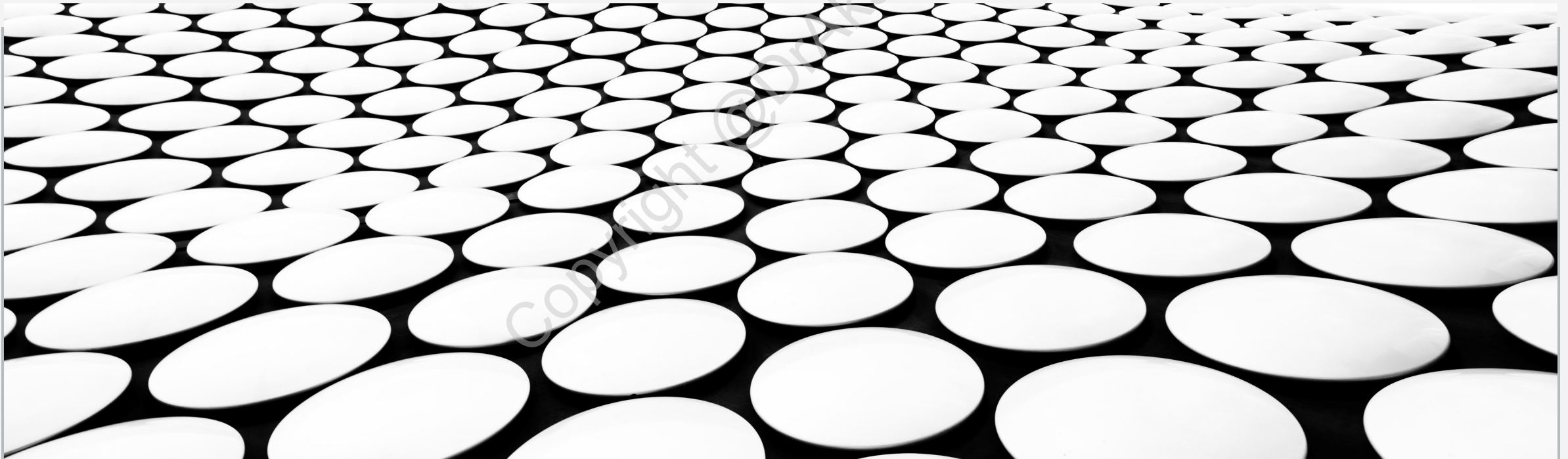


# FROM LARGE TO SMALL: THE RISE OF SMALL LANGUAGE MODELS (SLMs) IN TEXT ANALYTICS

EFFICIENCY, ADAPTABILITY, AND DOMAIN-SPECIFIC INTELLIGENCE



DR. AKSHI KUMAR

# THE GREAT BATTLES OF HISTORY – AND NOW, AI!

A Glimpse into Legendary Battles (Indian Context)

- *Mahabharata*: A war of strategy and intelligence.



# THE GREAT BATTLES OF HISTORY – AND NOW, AI!

A Glimpse into Legendary Battles (Indian Context)

- *Panipat*: A battle of power and endurance.

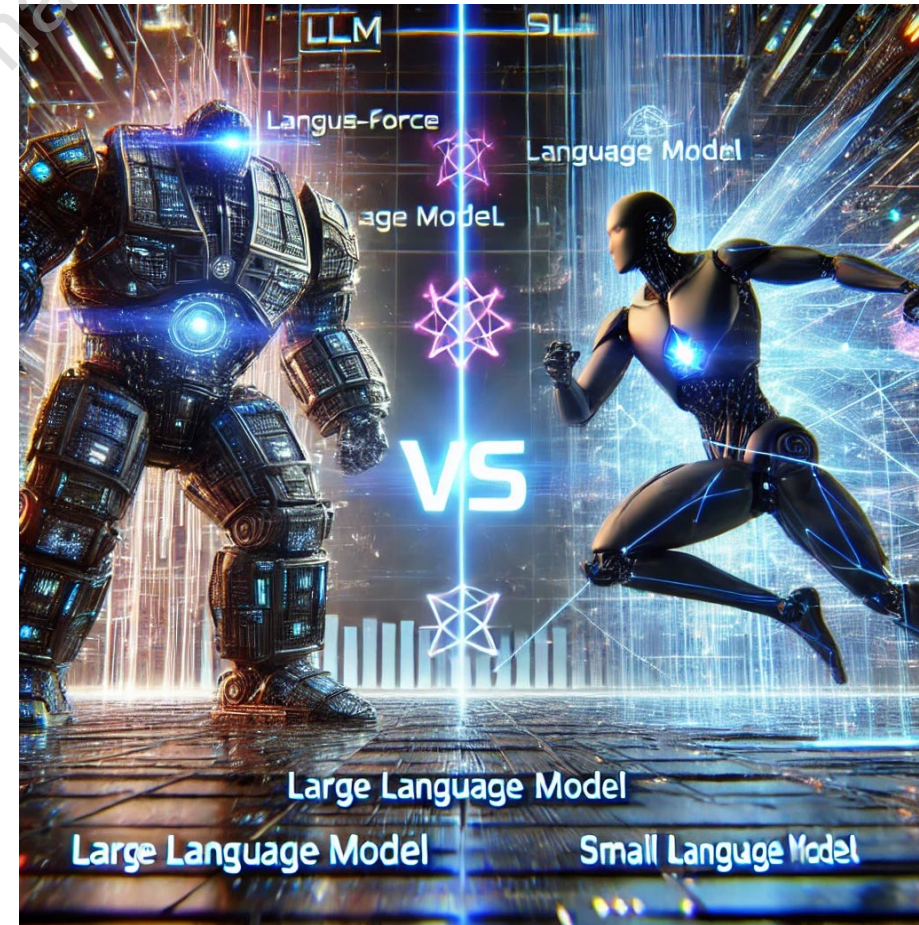


# THE GREAT BATTLES OF HISTORY – AND NOW, AI!

But imagine:

- What if the future history books talk about the AI battles of today?
- Would your grandchildren study LLMs vs. SLMs as an epic technological clash?

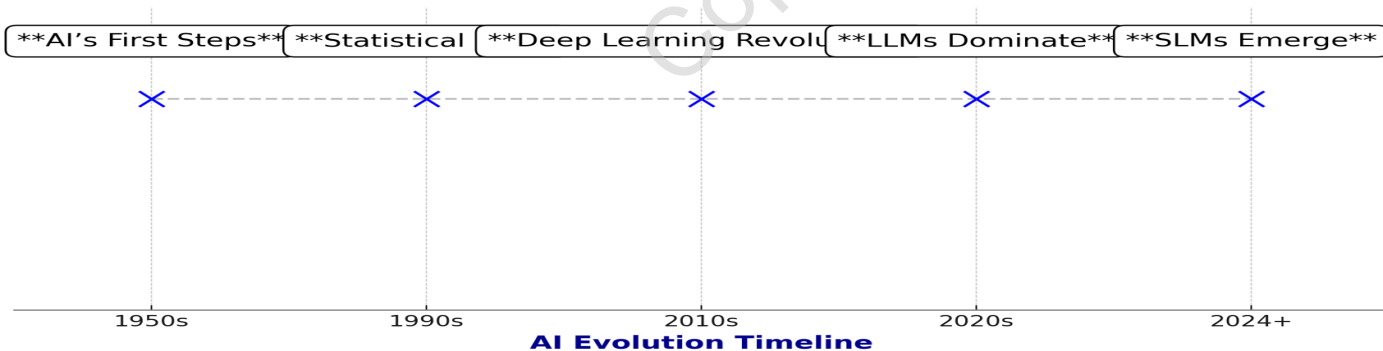
**AI vs. AI:** A battle of computation and efficiency, where Small Language Models (SLMs) challenge Large Language Models (LLMs) for dominance!



# IF AI HISTORY WERE A SCHOOL SUBJECT...

## The AI Timeline - What Would Your Grandkids Learn?

- 1950s: AI's first steps – Simple rule-based systems.
- 1990s: Statistical NLP – AI starts 'reading' text.
- 2010s: Deep Learning revolution – AI gets 'smarter.'
- 2020s: LLMs dominate – AI becomes a 'know-it-all.'
- 2024+: SLMs emerge – AI becomes 'efficient & practical.'



# IF AI HISTORY WERE A SCHOOL SUBJECT...

- Would history books talk about **LLMs vs. SLMs** as the **Mahabharata of machines**?
- Would **neural networks** be the **Kurukshetra** of tomorrow?



The image shows a futuristic AI battle featuring the Chakravyuh strategy, where an LLM fortress traps an SLM warrior in a high-tech digital battlefield.

---

Yes, It's a Lot of AI History... But Here's the Main Idea!

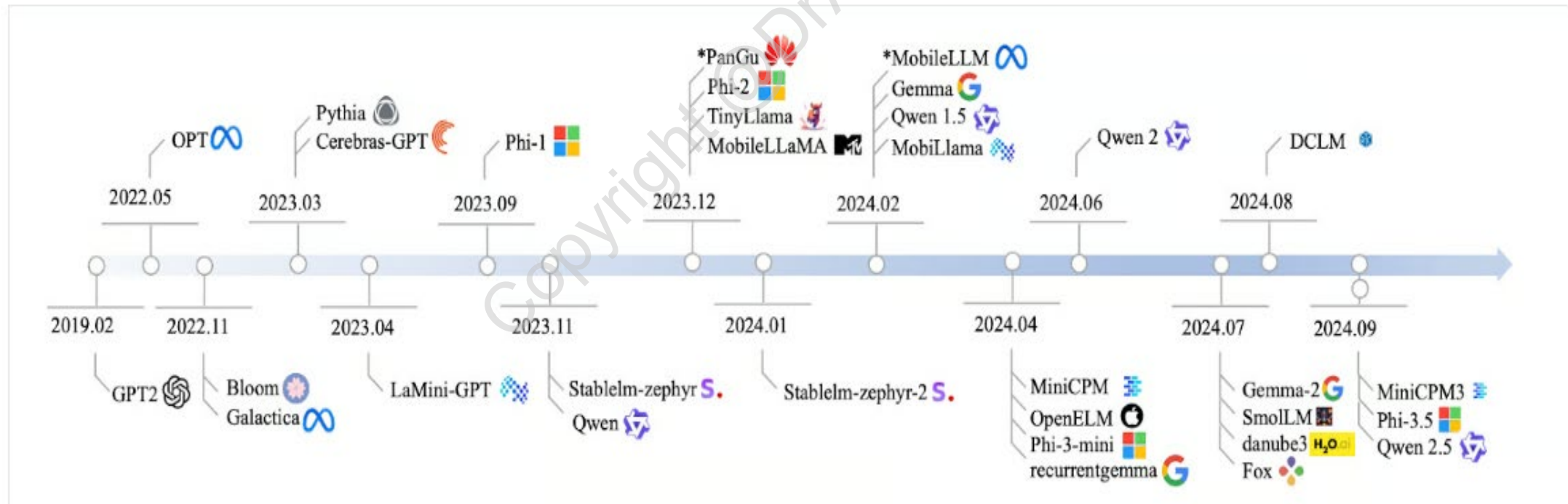
**WHICH ONE DO YOU THINK WINS IN THE REAL-  
WORLD AI LANDSCAPE?**

**'GIANT' OR 'AGILE'**

**WHY DO WE NEED SLMs WHEN LLMs ARE SO  
POWERFUL?**

# WHAT ARE SMALL LANGUAGE MODELS?

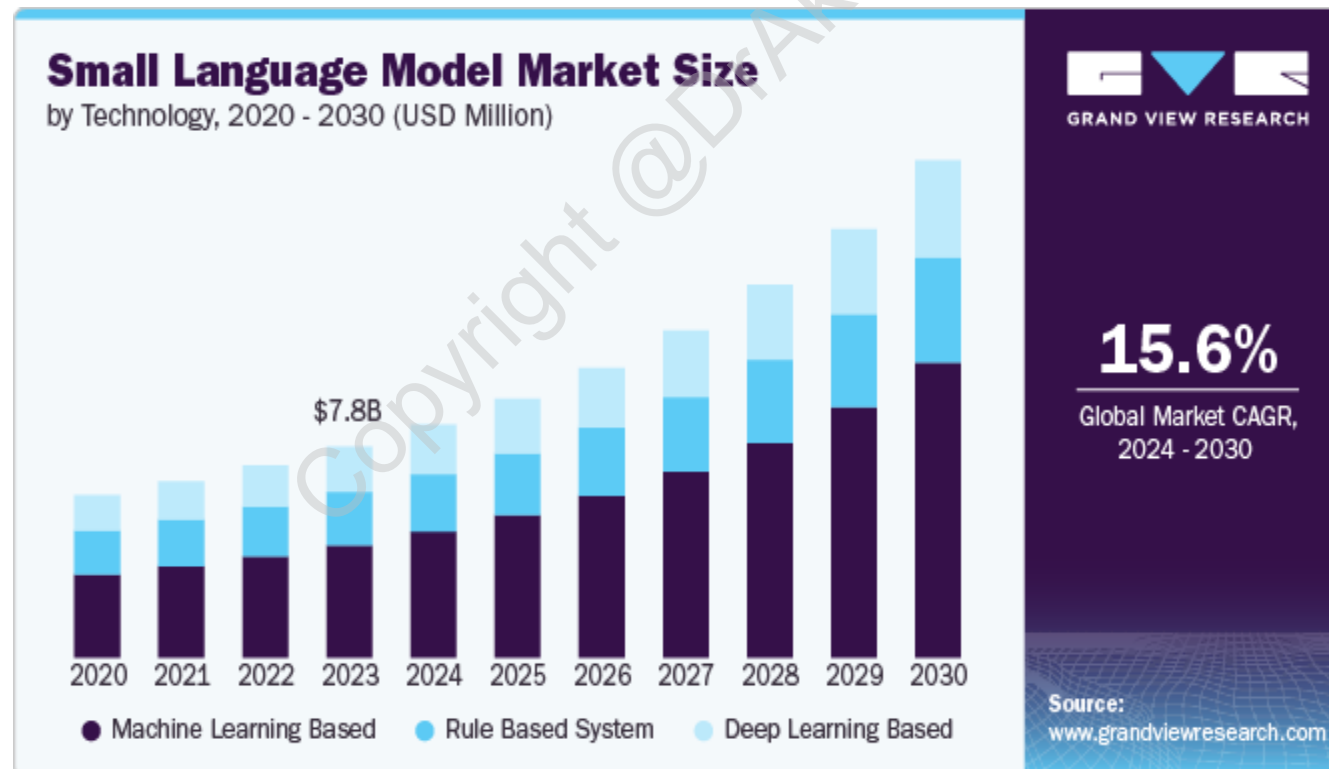
Small language models (SLMs) are artificial intelligence (AI) models capable of processing, understanding and generating natural language content. As their name implies, SLMs are **smaller in scale and scope** than large language models (LLMs).





# SMALL LANGUAGE MODEL MARKET SIZE & TRENDS

- The global small language model market size was estimated at USD 7.76 billion in 2023 and is projected to grow at a CAGR of 15.6% from 2024 to 2030.



# SMALL LANGUAGE MODEL MARKET SIZE & TRENDS

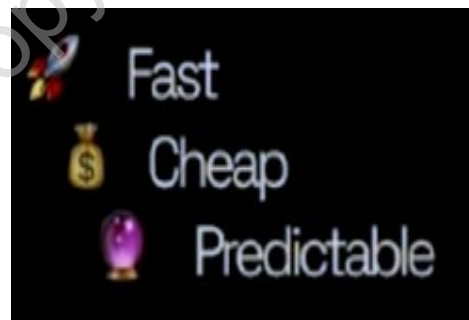
- As businesses manage the complexities of generative AI and small language model are provide promising solution that strikes a balance between **capability and practicality**.
- These models mark an important advancement in AI technology, providing companies with a way to leverage AI's power in a **more controlled, efficient, and customized manner**.
- The continuous improvements and innovations in small language model technology are expected to **significantly influence the future of enterprise AI solutions**.

# WHY SMALL LANGUAGE MODELS?

- **Lower Computational Costs:** SLMs require significantly fewer processing resources compared to LLMs, making them ideal for businesses with limited computing power.
- **Faster Response Times:** Due to their smaller size, SLMs can process requests much quicker, making them more efficient for real-time applications.
- **Easier Deployment:** SLMs can be deployed on edge devices such as mobile phones and IoT devices, reducing reliance on cloud-based infrastructure.

# WHY SMALL LANGUAGE MODELS?

- **Domain-Specific Fine-Tuning:** Unlike LLMs, which require massive datasets for training, SLMs can be fine-tuned on smaller, domain-specific data, improving accuracy and efficiency. This makes them well-suited for specific applications like customer support, healthcare, or education
- **Energy Efficiency:** SLMs consume significantly less power, contributing to sustainable AI development and reducing environmental impact.



# LLMs Vs. SLMs: PARAMETER SIZE & EFFICIENCY

- SLMs range from a few million to a few billion parameters, while LLMs can have hundreds of billions or even trillions.
- Parameters include weights and biases, which shape how a model learns and responds.
  - *What is a parameter?*  
In simple terms, parameter in a language model refers to the variables that the model uses to make predictions. Each parameter represents a concrete part of the model that can change or adapt based on the data it's trained on.

# LLMs Vs. SLMs: PARAMETER SIZE & EFFICIENCY

- **SLMs are optimized for efficiency**, requiring less memory and computational power.
- **Perfect for resource-constrained environments** such as mobile devices, IoT, and offline applications

# COMPARISON: COMPUTE REQUIREMENTS OF LLMs Vs. SLMs

Feature	Large Language Models (LLMs)	Small Language Models (SLMs)
Model Size	Billions of parameters	Millions of parameters
Compute Power	Requires high-end GPUs and TPUs	Can run on standard CPUs and edge devices
Training Time	Weeks to months	Hours to days
Latency	Slower response due to size	Faster real-time inference
Cost	Expensive to train and maintain	More cost-effective
Deployment	Requires cloud infrastructure	Can run locally and on edge devices
Energy Usage	High energy consumption	Lower power requirements

# SLMs AREN'T JUST A SMALLER VERSION OF SOMETHING BIGGER

THEY ARE THE FUTURE OF AI, BRINGING INTELLIGENCE  
DIRECTLY TO WHERE IT MATTERS MOST: YOUR DEVICES AND  
YOUR LIFE!

COMPACT DESIGN FOR BIG IMPACT...



# HOW SMALL LANGUAGE MODELS WORK?

- **SLMs are built on LLM foundations**, utilizing transformer-based architectures.

## Next word prediction

- Just like LLMs, SLMs work by predicting the *next word in a sequence of text*.
- SLMs use patterns from the text they've been trained on to guess what comes next.
- 
- It's a simple but powerful concept that lies at the heart of all language models.

For example, given the input: *"In the Harry Potter series, the main character's best friend is named Ron..."* An SLM would analyze this context and predict the most likely next word - in this case, *"Weasley."*

# HOW SMALL LANGUAGE MODELS WORK?

## Transformer architecture

- The transformer architecture is key to how LLMs and SLMs understand and generate language.
- *Transformers* can be understood as the brain behind language models.
- Key components of a transformer model:
  - **Encoders:** Convert input sequences into numerical embeddings capturing semantics.
  - **Self-attention mechanism:** Identifies key tokens, improving context awareness.
  - **Decoders:** Use self-attention and embeddings to generate relevant responses.
- They use *self-attention* to figure out which words in a sentence are most relevant to each other. This helps the model understand the context—for example, recognizing that “Paris” refers to the city or the person you know from work.

# HOW SMALL LANGUAGE MODELS WORK?

## Size and performance balance

- The power of SLMs lies in their ability to balance size and performance.
- They use significantly fewer parameters than LLMs, typically ranging from millions to a few billion, compared to hundreds of billions in LLMs.
- With fewer parameters, SLMs require less computational power and data to train, which makes them more accessible if you have limited resources.

# HOW SMALL LANGUAGE MODELS WORK?

- The compact size of SLMs makes them process input and generate output quicker, which is super important for real-time applications like mobile keyboards or voice assistants.
- SLMs might not be as versatile or deeply understanding as large models, but they handle specific tasks well. For example, an SLM trained to analyze legal texts could do a better job than a general LLM in that area.

# HOW SLMS ARE CREATED: TECHNIQUES AND APPROACHES

- **Pruning:** Eliminates redundant or low-impact parameters to reduce model size.
- **Quantization:** Converts high-precision data (e.g., 32-bit) into lower-precision data (e.g., 8-bit) to enhance efficiency.
- **Knowledge distillation:** Transfers knowledge from a large pre-trained model to a smaller, efficient mode.

# PRUNING – OPTIMIZING SLMs FOR PERFORMANCE

- Pruning is kind of like trimming what's not needed.
- It removes unnecessary parameters (e.g., neuron connections, layers, parameters) that don't contribute much to the overall performance.
- This technique helps to shrink the model without significantly impacting its accuracy.
- Improves real-time efficiency, making AI models faster and more responsive.

A large model is fully trained.



Parameters that contribute least to the model's performance are identified.



These less important parameters are removed, "pruning" the model to a smaller size.



The pruned model is often fine-tuned to regain any lost performance.

# PRUNING – OPTIMIZING SLMs FOR PERFORMANCE

- Requires fine-tuning post-pruning to regain accuracy.
- Overpruning can degrade performance, so finding the right balance is crucial.
- Pruning can significantly reduce model size while maintaining much of the original performance, which makes it an effective technique for creating SLMs.

A large model is fully trained.



Parameters that contribute least to the model's performance are identified.



These less important parameters are removed, "pruning" the model to a smaller size.



The pruned model is often fine-tuned to regain any lost performance.

# QUANTIZATION COMPLEXITY

- Quantization involves using fewer bits to store the model's numbers.
- Normally, a model might use 32-bit numbers, but with this method, those numbers are reduced to 8-bit values, which are much smaller.
- Converts high-precision weights and activations into lower-precision representations.

# REDUCING

# MODEL

The original model uses high-precision floating-point numbers for its parameters.



These parameters are converted to lower-precision representations.



This reduction in precision leads to smaller model sizes and faster computation.



# QUANTIZATION – REDUCING MODEL COMPLEXITY

- Two methods:
  - *Post-training quantization (PTQ)*: Applied after model training, requires less compute power.
  - *Quantization-aware training (QAT)*: Integrated during training, yielding better accuracy.
- Speeds up inference and reduces storage needs.
- The best part is that, even though the numbers are less precise, the model still works well with only a small impact on its accuracy.

# QUANTIZATION – REDUCING MODEL COMPLEXITY

Imagine you're storing temperature values in a weather app. You'd store them with high precision (like 32-bit numbers), which is more than you need. By reducing the precision to 8-bit, you might lose details, but the app will still be useful while running faster and using less memory.

- This is particularly useful for deploying AI on devices with limited memory and computational power, like smartphones or edge devices.

# KNOWLEDGE DISTILLATION – LEARNING FROM LARGER MODELS

- A technique where a smaller model (student) learns from a larger, pre-trained model (teacher).
- The goal here is to take what the teacher model has learned and compress that into the student model without losing too much of its performance.
- This process makes SLMs retain much of the accuracy of larger models while being far more manageable in size and computational need.

The larger model (teacher) is trained on a dataset.



The smaller model (student) is trained to mimic the outputs of the teacher model.



A specialized loss function measures the difference between the outputs of the two models, guiding the student's learning.

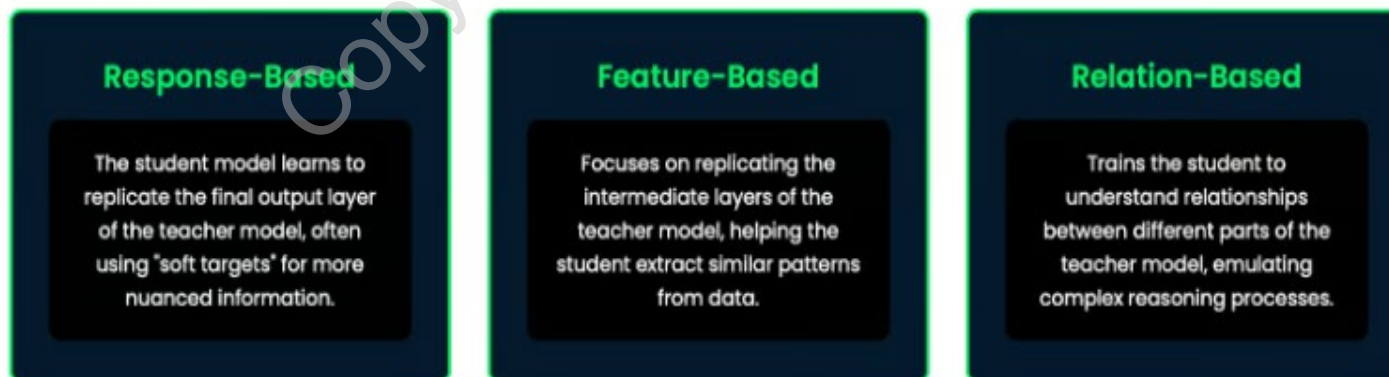
# KNOWLEDGE DISTILLATION – LEARNING FROM LARGER MODELS

- With this technique, the smaller model learns not just the final predictions of the teacher, but also the underlying patterns and nuances.
- **Process:**
  - **Training the student model:** The student replicates the outputs of the teacher, learning key patterns and decision-making behaviors.
  - **Squeezing knowledge:** The student distills the teacher's expertise, leveraging its knowledge while reducing computational demands.
  - **Performance retention:** Despite being smaller, the student model can achieve comparable performance by absorbing critical insights from the larger model.
- Enables small models to retain the accuracy and intelligence of LLMs while operating efficiently

# KNOWLEDGE DISTILLATION – LEARNING FROM LARGER MODELS

There are several methods of knowledge distillation:

- **Response-based:** The student model learns to replicate the final output layer of the teacher model, often using "soft targets" for more nuanced information.
- **Feature-based:** Focuses on replicating the intermediate layers of the teacher model, helping the student extract similar patterns from data.
- **Relation-based:** Trains the student to understand relationships between different parts of the teacher model, emulating complex reasoning processes.



# FINE-TUNING STRATEGIES FOR SLMS

## Transfer Learning

- Involves taking a pre-trained model and adapting it for a new task.
- Instead of training from scratch, transfer learning repurposes knowledge from a large, general dataset.
- Reduces training time and computational requirements while improving performance on domain-specific tasks.
- Example: A general NLP model pre-trained on Wikipedia text can be fine-tuned to analyze legal contracts, significantly improving its performance in that domain.

# FINE-TUNING STRATEGIES FOR SLMs

## Domain Adaptation

- Focuses on customizing models for industry-specific datasets.
- Helps SLMs understand specialized terminology and domain-specific patterns.
- Enhances accuracy in areas like finance, healthcare, and law by adjusting model behavior to fit specific linguistic structures and concepts.
- Example: Fine-tuning a medical chatbot on clinical texts to improve diagnosis recommendations.

# FINE-TUNING STRATEGIES FOR SLMS

## Parameter-Efficient Fine-Tuning (PEFT)

- Traditional fine-tuning involves adjusting all model parameters, which is computationally expensive.
- PEFT techniques allow modifying only a small subset of parameters while keeping the core model intact, reducing memory and processing requirements.
- Key PEFT Techniques:
  - ***LoRA (Low-Rank Adaptation)***: Instead of modifying all model weights, LoRA adds trainable low-rank matrices to frozen pre-trained parameters.
    - It maintains the generalization ability of the original model while learning task-specific nuances.
    - Use Case: Effective for domain-specific tasks like medical text processing or financial analytics without retraining the entire model.



# FINE-TUNING STRATEGIES FOR SLMS

- ***Adapter Layers:*** Introduces additional lightweight layers in the model, trained specifically for new tasks.
  - The main model remains unchanged, while adapter layers capture task-specific knowledge.
  - Use Case: Improves multi-task learning, allowing a single model to switch contexts efficiently.
- ***Prefix Tuning:*** Instead of fine-tuning the full model, prefix tuning modifies only the prompt embeddings.
  - It learns a small set of continuous task-specific prompts while keeping the underlying model unchanged.
  - Use Case: Particularly effective in dialog systems where prompts guide model responses without needing full fine-tuning.

# FINE-TUNING STRATEGIES FOR SLMs

## Contrastive Learning & Reinforcement Fine-Tuning

### Contrastive Learning:

- Helps SLMs learn better representations by distinguishing between similar and dissimilar text samples.
- Encourages the model to develop rich embeddings even with limited labelled data.
- Used in sentence embeddings, image-text alignments, and document clustering.

### Reinforcement Learning Fine-Tuning:

- Allows AI models to learn from rewards and penalties.
- Optimizes AI behavior based on feedback, making it more adaptive.
- Example: Chatbots that dynamically refine responses based on user satisfaction metrics.

Both techniques enhance SLMs' learning efficiency, enabling rapid adaptation to changing user needs.

# SMALL LANGUAGE MODELS EVOLUTION

- The development of SLMs from 2019 to 2024 has been fast, with many new models being created to meet the need for more efficient AI.
- It started with GPT-2 in 2019, and over the years, models have become more focused and faster.
- By 2022, models like Bloom and Galactica could handle multiple languages and scientific data, and in 2023, models like Pythia and Cerebras-GPT were designed for tasks like coding and logical thinking.
- In 2024, even more SLMs were released, such as LaMini-GPT, MobileLLaMA, and TinyLlama, which are made to work well on mobile devices and other low-power systems.
- Companies like Meta, Google, and Microsoft are leading the development of these models, with some being open to the public and others kept private.

# SLM EXAMPLES

Here are some of these models with their parameters and key features:

Model Name	Parameters	Open Source	Key Features
Qwen2	0.5B, 1B, 7B	Yes	Scalable, suitable for various tasks
Mistral Nemo 12B	12B	Yes	Complex NLP tasks, local deployment
Llama 3.1 8B	8B	Yes*	Balanced power and efficiency
Pythia	160M - 2.8B	Yes	Focused on reasoning and coding
Cerebras-GPT	111M - 2.7B	Yes	Compute-efficient, follows Chinchilla scaling laws
Phi-3.5	3.8B	Yes**	Long context length (128K tokens), multilingual
StableLM-zephyr	3B	Yes	Fast inference, efficient for edge systems
TinyLlama	1.1B	Yes	Efficient for mobile and edge devices

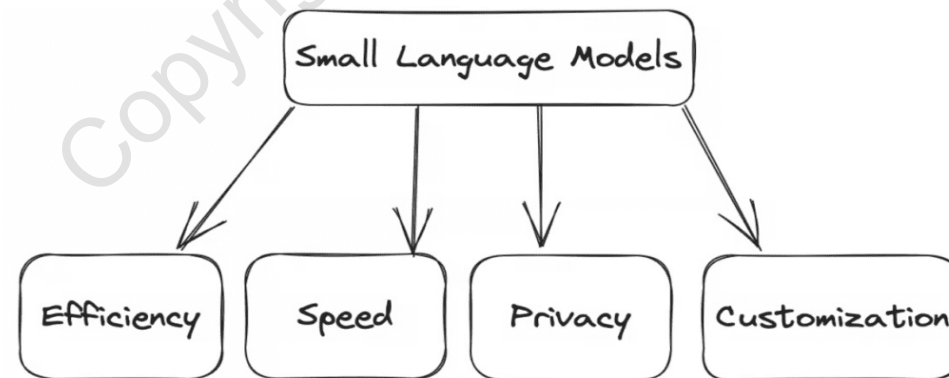
\*With usage restrictions; \*\*For research purposes only

# SLM EXAMPLES

Here are some of these models with their parameters and key features:

Model Name	Parameters	Open Source	Key Features
MobileLLaMA	1.4B	Yes	Optimized for mobile and low-power devices
LaMini-GPT	774M - 1.5B	Yes	Multilingual, instruction-following tasks
Gemma2	9B, 27B	Yes	Local deployment, real-time applications
MiniCPM	1B - 4B	Yes	Balanced performance, English and Chinese optimized
OpenELM	270M - 3B	Yes	Multitasking, low-latency, energy-efficient
DCLM	1B	Yes	Common-sense reasoning, logical deduction
Fox	1.6B	Yes	Speed-optimized for mobile applications

**THE BEAUTY OF SLMS LIES IN THEIR ABILITY TO DELIVER POWERFUL AI WITHOUT NEEDING MASSIVE INFRASTRUCTURE OR CONSTANT INTERNET CONNECTIVITY, WHICH OPENS UP SO MANY APPLICATIONS!**



# SMALL LANGUAGE MODEL USE CASES

- Enterprises can fine-tune SLMs on domain-specific datasets to customize them for their specific needs.
- This adaptability means small language models can be employed for a variety of real-world applications.



# ON-DEVICE AI – AI IN MOBILE AND EDGE COMPUTING

- **Voice Assistants:** SLMs power real-time voice recognition for faster and privacy-focused interactions.
- **Text Prediction:** Apps like SwiftKey and Gboard enhance text typing through SLM-driven context awareness.
- **Offline AI Processing:** Google Translate uses SLMs for offline language translation, benefiting areas with limited internet access.



# PERSONALIZED AI – TAILORING AI FOR INDIVIDUALS AND BUSINESSES

- **Healthcare:** SLMs enhance medical text analysis, providing real-time health monitoring on smart wearables.
- **Smart Home Devices:** AI-powered assistants learn user preferences for lighting, temperature, and energy efficiency.
- **Education:** Adaptive learning platforms use SLMs to personalize instruction, adjusting to students' learning pace.

# INTERNET OF THINGS (IOT) – MAKING DEVICES SMARTER

- SLMs **quietly run in the background**, enabling smart home systems and connected devices to operate efficiently without constant cloud communication.
- Improves **privacy and security** by keeping AI interactions local.
- **Example:** AI-driven refrigerators that suggest groceries based on usage patterns.

# REAL-TIME LANGUAGE TRANSLATION

- SLMs enable instant translation for global communication.
- **Travel Apps:** Real-time translation of signs, menus, and conversations.
- **Multilingual Communication:** Enhances accessibility in business and tourism industries.

Copyright © Dr. Ashikumar

# AUTOMOTIVE SYSTEMS – AI IN SMART VEHICLES

- **Navigation Optimization:** Real-time traffic updates and route suggestions.
- **Voice-Controlled Commands:** Enables hands-free operation for music, calls, and smart car systems.
- **Safety Enhancements:** AI-based emergency detection and response.

# ENTERTAINMENT & GAMING – PERSONALIZED AI FOR USERS

- **Smart TVs and Streaming Platforms:** AI-powered content recommendations based on watch history.
- **Gaming Consoles:** Adaptive gaming AI for personalized user experiences.
- **Voice Control:** AI assistants for controlling entertainment devices hands-free.

# CUSTOMER SERVICE – AI IN BUSINESS INTERACTIONS

- **Retail AI Chatbots:** SLM-powered virtual assistants help customers with product recommendations, order tracking, and FAQs.
- **Financial AI Advisors:** SLMs analyze market trends for personalized investment insights.
- **E-commerce:** AI chatbots handle returns, refunds, and support inquiries efficiently.

# REAL WORLD EXAMPLE OF SLM

- A notable real-world example of Small Language Models (SLMs) in action is Microsoft's internal application for cloud supply chain fulfillment.
- In this case, Microsoft developed an SLM tailored to facilitate natural language interactions within their supply chain processes.
- The results demonstrated that these smaller models not only outperformed larger ones in accuracy but also offered reduced running times, even when fine-tuned on limited datasets.
- This showcases the potential of SLMs to deliver efficient and effective solutions in specialized domains.

Li, B., Zhang, Y., Bubeck, S., Pathuri, J., & Menache, I. (2024). [Small language models for application interactions: A case study.](#) *arXiv preprint arXiv:2405.20347.*

# DEMO OF SLMs LOCALLY USING OLLAMA





# DEMO OF SLMs LOCALLY USING OLLAMA



# DEMO OF SLM LOCALLY USING OLLAMA



# DEMO OF SLMs LOCALLY USING OLLAMA



# SLM ADVANTAGE: TRAINING EFFICIENCY – OPTIMIZING LEARNING

- **Efficient Training Methods:** Using techniques like progressive learning, layer freezing, and batch size optimization to reduce computation.
- **Data Efficiency:** Training with smaller, high-quality datasets to improve model accuracy without excessive resource consumption.
- **Adaptive Learning Rates:** Implementing dynamic learning rates to fine-tune SLMs more efficiently.
- **Hardware Optimization:** Leveraging TPUs and optimized GPUs for cost-effective training.

# ENERGY CONSUMPTION & SUSTAINABILITY

## Why Energy Efficiency Matters in AI?

- AI models consume vast amounts of computational power, contributing to high energy use.
- LLMs require extensive GPU clusters, leading to increased carbon emissions.
- SLMs, being lightweight, use fewer computational resources, making them a more sustainable alternative.
- **Green AI Movement:** Growing emphasis on developing energy-efficient models for reducing environmental impact.

# HOW SLMs CONTRIBUTE TO SUSTAINABILITY?

- **Lower Carbon Footprint:** Requires fewer computational resources, reducing data center emissions.
- **Edge Computing Friendly:** Can be deployed on low-power devices, eliminating the need for constant cloud access.
- **Energy-Efficient Training:** Uses model compression techniques like pruning and quantization to reduce energy costs.

## Comparison of Energy Usage: LLMs vs. SLMs

Model Type	Training Energy Consumption	Inference Energy Usage
LLMs	High (Weeks to Months)	High (Requires Cloud)
SLMs	Low (Hours to Days)	Low (Can Run on Edge Devices)

# ETHICAL CONSIDERATIONS – BIAS & FAIRNESS IN SLMS

## Understanding Bias in AI Models

- **What is Algorithmic Bias?**
  - AI models can **learn biases** from imbalanced datasets.
  - If a dataset contains **historical or societal biases**, the AI can reinforce them.
- **Fairness in AI Decision-Making:**
  - AI should provide **equitable results across different demographics**.
  - **Bias Detection & Fairness Audits** help ensure that SLMs do not reinforce discrimination.
- **Common Types of Bias in AI**

Bias Type	Example
Sampling Bias	Dataset favors one demographic group.
Labelling Bias	Annotators unknowingly introduce subjective opinions.
Automation Bias	Users trust AI decisions even if flawed.

# ETHICAL CONSIDERATIONS – BIAS & FAIRNESS IN SLMS

## Mitigating Bias in SLMs

- **Data Balancing:** Ensuring diverse and representative datasets for training.
- **Bias Detection Tools:** Using frameworks like IBM AI Fairness 360 to detect and reduce biases.
- **Adversarial Debiasing:** Training models to actively reduce discriminatory tendencies.
- **Ethical AI Regulations:** Ensuring compliance with laws like GDPR and AI ethics guidelines.



# EXPLAINABILITY & INTERPRETABILITY IN SMALL MODELS

## Why Explainability Matters in AI

- AI decisions should be transparent and understandable.
- Black-box models like LLMs make it difficult to trace decision-making.
- SLMs, being smaller, offer more explainable outputs, making them ideal for regulated industries (e.g., healthcare, finance).
- **Regulatory Demands:** Many industries require AI models to be interpretable for compliance and trust.

# EXPLAINABILITY & INTERPRETABILITY IN SMALL MODELS

## Techniques for Explainable AI (XAI) in SLMs

Method	Explanation
SHAP (SHapley Additive Explanations)	Identifies which inputs impact model predictions the most.
LIME (Local Interpretable Model-agnostic Explanations)	Generates interpretable approximations for AI decisions.
Attention Visualization	Highlights key words/phrases that influenced an AI response.

# CHALLENGES OF SLMs

## 1. Limited Knowledge Capacity (Inherited from LLMs)

- Like LLMs, SLMs still struggle with factual accuracy and hallucinations.
- Smaller models lack broad general knowledge, making them more reliant on fine-tuning.
- Example: An SLM trained on financial documents may fail to answer legal-related queries accurately.

## 2. Performance Trade-offs (Partially Resolved in SLMs)

- LLMs require massive computational resources, while SLMs optimize efficiency.
- SLMs trade off some performance for speed and lower resource usage.
- Example: While an LLM can generate highly creative content, an SLM is better suited for real-time applications like chatbots.

# CHALLENGES OF SLMs

## 3. Limited Generalization Ability (Still a Challenge)

- SLMs excel in domain-specific tasks but struggle with general knowledge.
- Unlike LLMs, they need frequent fine-tuning to remain relevant.
- Example: An SLM trained for customer support might fail when handling out-of-domain questions.

## 4. Adaptability to New Contexts (Improved but Not Fully Solved)

- SLMs require fine-tuning but can be adapted faster than LLMs.
- Frequent retraining is needed, but with fewer computational resources compared to LLMs.
- Example: A product recommendation model in e-commerce needs updates as user preferences change

# CHALLENGES OF SLMs

## 5. Ethical and Bias Concerns (A Persistent Issue in Both LLMs & SLMs)

- SLMs inherit biases from their training data, just like LLMs.
- Bias detection and mitigation techniques remain crucial for both models.
- Example: A hiring AI model trained on historical job applications may exhibit gender or racial bias if not carefully balanced.

Copyright @DANIEL KUMAR

# COMBINING LLMS AND SLMS

Advances in AI development have led to optimization approaches that maximize the joint power of LLMs and SLMs:

- **Hybrid AI pattern:** A hybrid AI model can have smaller models running on premises and accessing LLMs in the public cloud when a larger corpus of data is required to respond to a prompt.
- **Intelligent routing:** Intelligent routing can be applied to more efficiently distribute AI workloads. A routing module can be created to accept queries, evaluate them and choose the most appropriate model to direct queries to. Small language models can handle basic requests, while large language models can tackle more complicated ones.

# THE FUTURE OF SLMs IN THE NEXT 5 YEARS

## 1. Expansion Across Industries

- **Healthcare:** AI-driven diagnostics, personalized treatment recommendations.
- **Finance:** Fraud detection, real-time customer interactions with reduced latency.
- **Retail & E-Commerce:** AI-powered shopping assistants and recommendation engines.

## 2. Refinements in Model Efficiency & Performance

- **Breakthroughs in compression techniques** will enable even smaller, more powerful models.
- **Adaptive learning algorithms** will allow SLMs to retain information better without frequent retraining.
- **Greater autonomy in decision-making** for real-world applications.

# THE FUTURE OF SLMs IN THE NEXT 5 YEARS

## 3. SLMs as the Future of Responsible AI

- **Privacy-first AI:** More applications will leverage SLMs on-device to ensure data security.
- **Low-energy AI:** SLMs will dominate where energy efficiency is key, such as edge devices and IoT.
- **Ethically aware models:** Enhanced frameworks for bias detection and explainability.

 *SLMs are not just a smaller version of AI—they are the next step in making AI more accessible, responsible, and efficient!* 





# UNDERSTANDING SLM & LLM – BEYOND JUST THE ACRONYMS

- SLM – Small Language Model
  - Smart, Lightweight, Mobile
  - Optimized for efficiency, designed for fast inference and low computational cost.
  - Ideal for on-device AI, privacy-sensitive applications, and real-time decision-making.
  - Think of it as a tactical warrior—precise, agile, and cost-effective.
- LLM – Large Language Model
  - Large-scale, Logically-rich, Multifunctional
  - Deep contextual understanding, capable of handling complex and creative tasks.
  - Requires massive datasets and high computational power.
  - Think of it as a grand general—powerful, strategic, but resource-heavy.

# DECISION MATRIX – CHOOSING BETWEEN SLM VS. LLM?

The following matrix provides a quick way to determine whether an **SLM (Small Language Model)** or **LLM (Large Language Model)** is best suited for your use case based on key criteria:

Criteria	SLM 	LLM 
Computational Cost	Low	High
Inference Speed	Fast	Slower
Scalability	Limited	High
Data Requirement	Small, domain-specific	Massive, general-purpose
Privacy & Security	High (on-device processing)	Lower (cloud-based)
Deployment	Mobile, Edge, IoT	Cloud, Enterprise Servers
Contextual Understanding	Basic to Moderate	Deep, Complex Reasoning
Creative Text Generation	Simple responses	Advanced, human-like text
Real-Time Applications	Excellent	Moderate to High Latency
Energy Efficiency	High	Low
Use Case Complexity	Low to Medium	High

# HOW TO USE THE DECISION MATRIX?

- If your requirements align more with the SLM column, a Small Language Model is the better choice.
- If your needs fit the LLM column, then an LLM is preferable for advanced, high-performance tasks.
- If your needs are mixed, consider a hybrid approach, leveraging SLMs for efficiency and LLMs for complex tasks.

 *Choosing between an SLM and an LLM is not about which is superior—it's about what fits best for your task!* 

# SUMMING UP

- “Simpler is better” rings truer than ever!
- The focus of AI is on practicality, reduced costs, and optimized efficiency.
- While large, general-purpose models were initially favoured, the focus is now on efficiency and cost-effectiveness. The future of AI is shifting towards smaller, specialized models.
- Therefore, businesses seek models that excel at specific domains rather than a one-size-fits-all approach.
- *SLMs and LLMs serve different purposes—choosing the right model depends on your needs for speed, scale, and intelligence!*

# CONCLUSION-STRATEGY OVER SIZE!

- History has shown us that **strategy outweighs brute force**. The shift from LLMs to SLMs reflects this principle.
- Choosing between an SLM and an LLM isn't about which is superior—it's about selecting the right tool for the right task.



*“From the battlefields of history to the war of algorithms—strategy, not size, decides the victor!”*





**Thank You for Your Time!**



**For Queries & Collaborations:**

**Email:** *Akshi.Kumar@gold.ac.uk*