Can LLMs solve Fusion ?

Probably not, but lets entertain the argument

Vignesh Gopakumar
UKAEA / UCL / Simvue

SML Research Day
27th June, 2024

Transformers in Fusion

Two Approaches

Foundation Physics Models

Deploying large vision transformer based models to solve a wide range of partial differential equations collectively.

Computer Vision based.

LLM based Information-Retrieval

Using modern LLMs as data embeddings over Fusion related textual information.

Natural Language Processing based.

Why even bother with this ?

Foundation Physics Models

Research Landscape

DPOT: Auto-Regressive Denoising Operator Transformer for Large-Scale PDE Pre-Training

Zhongkai Hao [1 2] **Chang Su** [1] **Songming Liu** [1] **Julius Berner** [3] **Chengyang Ying** [1] **Hang Su** [1]
Anima Anandkumar [3] **Jian Song** [2] **Jun Zhu** [1 4]

Universal Physics Transformers: A Framework For Efficiently Scaling Neural Operators

Benedikt Alkin [1,2] **Andreas Fürst** [1] **Simon Schmid** [3] **Lukas Gruber** [1]
Markus Holzleitner [1] **Johannes Brandstetter** [1,2]

[1] ELLIS Unit Linz, Institute for Machine Learning, JKU Linz, Austria
[2] NXAI GmbH, Linz, Austria
[3] Software Competence Center Hagenberg GmbH, Hagenberg, Austria
{alkin, fuerst, brandstetter}@ml.jku.at

Pretraining Codomain Attention Neural Operators for Solving Multiphysics PDEs

Md Ashiqur Rahman [1] **Robert Joseph George** [2] **Mogab Elleithy** [2] **Daniel Leibovici** [2] **Zongyi Li** [2] **Boris Bonev** [3]
Colin White [3] **Julius Berner** [2] **Raymond A. Yeh** [1] **Jean Kossaifi** [3] **Kamyar Azizzadenesheli** [3]
Anima Anandkumar [2]

MULTIPLE PHYSICS PRETRAINING FOR PHYSICAL SURROGATE MODELS

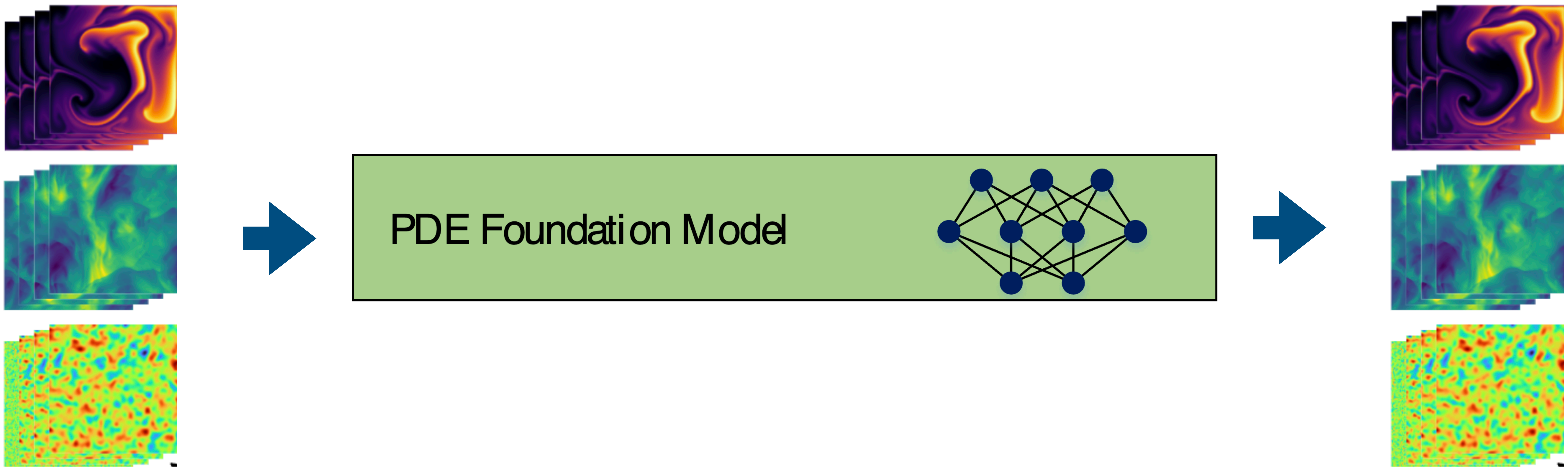
Michael McCabe [*,1,2] **Bruno Régaldo-Saint Blancard** [1] **Liam Parker** [1] **Ruben Ohana** [1]
Miles Cranmer [3] **Alberto Bietti** [1] **Michael Eickenberg** [1] **Siavash Golkar** [1]
Geraud Krawezik [1] **Francois Lanusse** [1,4] **Mariel Pettee** [1,5]
Tiberiu Tesileanu [1] **Kyunghyun Cho** [6,7,8] **Shirley Ho** [1,6,9]

The Polymathic AI Collaboration

[1] Flatiron Institute [2] University of Colorado Boulder [3] University of Cambridge
[4] Université Paris-Saclay, Université Paris Cité, CEA, CNRS, AIM
[5] Physics Division, Lawrence Berkeley National Laboratory
[6] New York University [7] Prescient Design, Genentech
[8] CIFAR Fellow [9] Princeton University

Foundation Physics Models

Basic Structure



$t=0,\ldots,t_{in}$ $t=t_{in}+1,\ldots,t_{out}$

Foundation Fusion Physics Models ?

- Helping Jake and Dan with an improved Foundation Physics Model
- Neural Operator based approach that uses Attention within the Fourier space
- Will be implementing Fusion Magnetohydrodynamic and Gyrokinetic datasets

---

Issues:

Geometry Constraints, Physics Constraints, Significant Pre-processing

LLM-Based Information Retrieval

Using RAG over Fusion related textual information

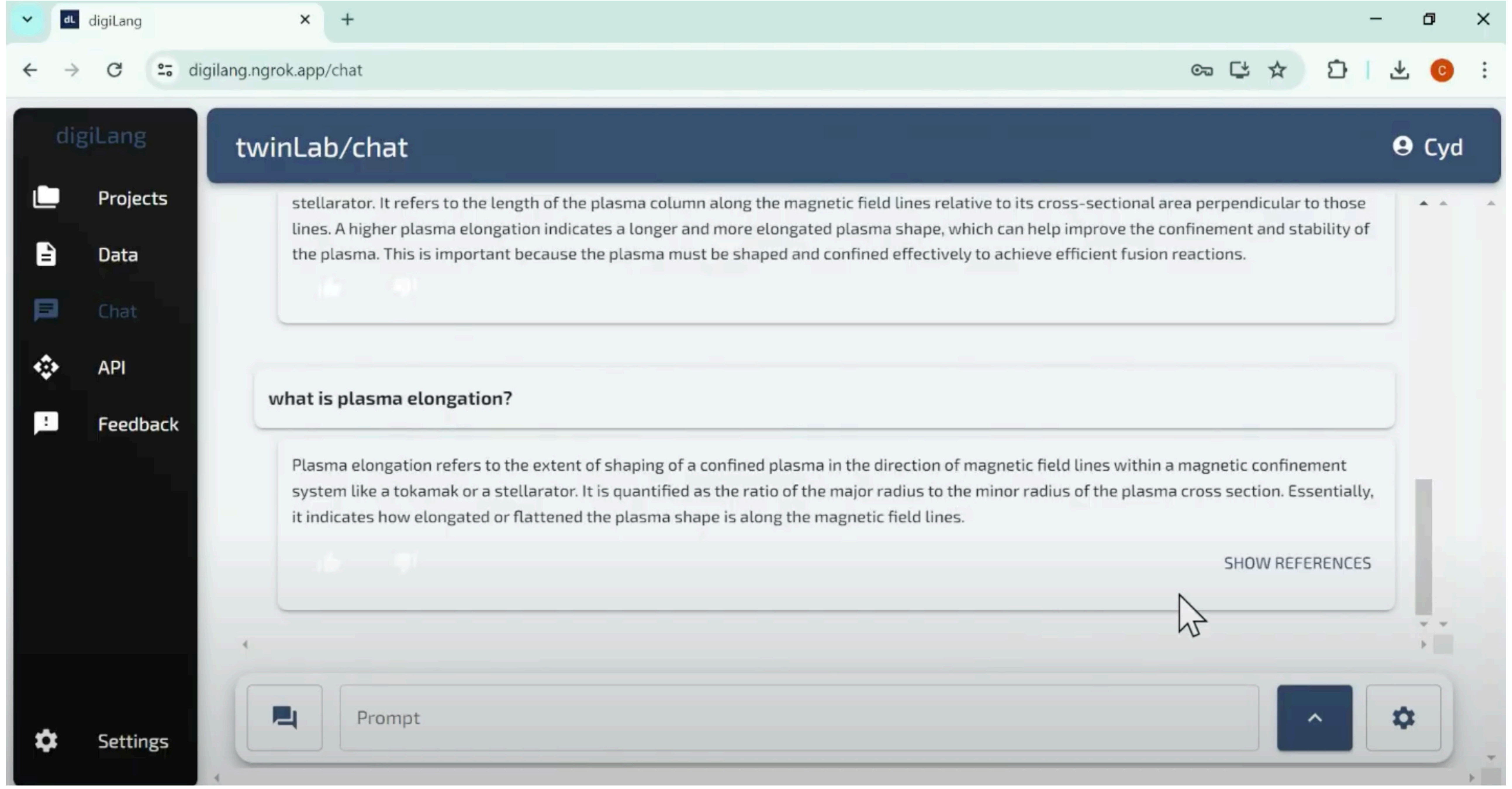
Shot-logs over Alcator C-Mod and DIII-D (Two Fusion Devices based in the US)

Towards LLMs as Operational Copilots for Fusion Reactors

Viraj Mehta[1], Joe Abbate[2], Allen M. Wang[3], Andy Rothstein[2,4], Ian Char[1], Jeff Schneider[1], Egemen Kolemen[2,4], Cristina Rea[3], and Darren T. Garnier[3]

[1]Carnegie Mellon University, Pittsburgh, PA, USA
[2]Princeton Plasma Physics Laboratory, Princeton, NJ, USA
[3]MIT Plasma Science and Fusion Center, Cambridge, MA, USA
[4]Department of Mechanical and Aerospace Engineering, Princeton University, Princeton, NJ, USA

Nuclear Fusion publications over the past 50 years (Journal under IOP)



Transformers in Fusion

Two Approaches

Foundation Physics Models

LLM based Information-Retrieval

Embeddings

Can LLMs solve Fusion ?

Can LLMs solve PDEs ?

Vignesh Gopakumar
UKAEA / UCL / Simvue

SML Research Day
27th June, 2024

Solving PDEs the Numerical Way

FD, FEM, FVM solvers mainly written in Fortran or C/C++

OpenFOAM

MOOSE

NEKTAR++
SPECTRAL/HP ELEMENT FRAMEWORK

fenics project

PETSc

Basic Structure of a Numerical Code

Geometry

```
/*--------------------------------*- C++ -*----------------------------------
  =========                 |
  \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox
   \\    /   O peration     | Website:  https://openfoam.org
    \\  /    A nd           | Version:  7
     \\/     M anipulation  |
\*---------------------------------------------------------------------------
FoamFile
{
    version     2.0;
    format      ascii;
    class       cellList;
    object      cells;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *

10720
(
6(0 1 21626 21627 2 21548)
6(1 3 21628 21629 4 21549)
6(3 5 21630 21631 6 21550)
6(5 7 21632 21633 8 21551)
6(7 9 21634 21635 10 21552)
6(9 11 21636 21637 12 21553)
6(11 13 21638 21639 14 21554)
6(13 15 21640 21641 16 21555)
6(15 17 21642 21643 18 21556)
6(17 19 21644 21645 20 21557)
6(19 21 21646 21647 22 21558)
```

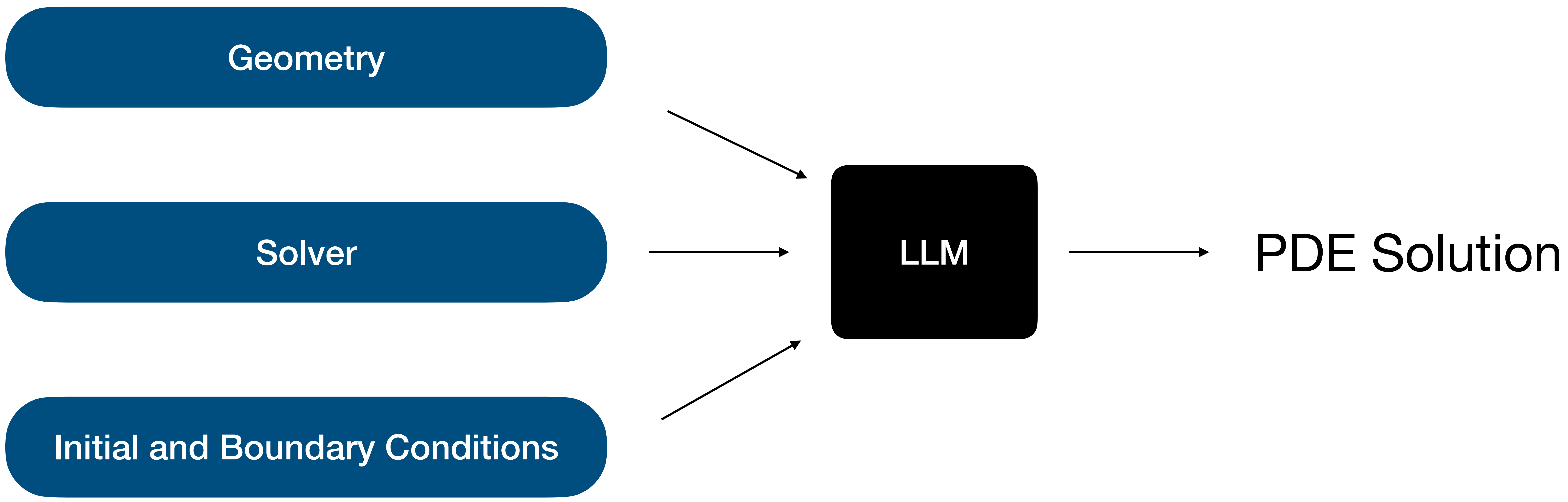
Solver

```
/*--------------------------------*- C++ -*----------------------------------
  =========                 |
  \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox
   \\    /   O peration     | Website:  https://openfoam.org
    \\  /    A nd           | Version:  7
     \\/     M anipulation  |
\*---------------------------------------------------------------------------
FoamFile
{
    version     2.0;
    format      ascii;
    class       dictionary;
    location    "system";
    object      fvSchemes;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *

ddtSchemes
{
    default         steadyState;
}

gradSchemes
{
    default         Gauss linear;
}

divSchemes
{
    default         none;
    div(phi,U)      bounded Gauss linearUpwind grad(U);
    div(phi,nuTilda) bounded Gauss linearUpwind grad(nuTilda);
    div((nuEff*dev2(T(grad(U))))) Gauss linear;
}

laplacianSchemes
```

Initial and Boundary Conditions

```
/*--------------------------------*- C++ -*----------------------------------
  =========                 |
  \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox
   \\    /   O peration     | Website:  https://openfoam.org
    \\  /    A nd           | Version:  7
     \\/     M anipulation  |
\*---------------------------------------------------------------------------
FoamFile
{
    version     2.0;
    format      ascii;
    class       volVectorField;
    object      U;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *

dimensions      [0 1 -1 0 0 0 0];

internalField   uniform (25.75 3.62 0);

boundaryField
{
    inlet
    {
        type            freestreamVelocity;
        freestreamValue $internalField;
    }

    outlet
    {
        type            freestreamVelocity;
        freestreamValue $internalField;
    }

    walls
    {
        type            noSlip;
    }

    frontAndBack
    {
        type            empty;
    }
}
```

LLMs to solve PDEs



LLMs to solve PDEs

Token Issue

Inputs

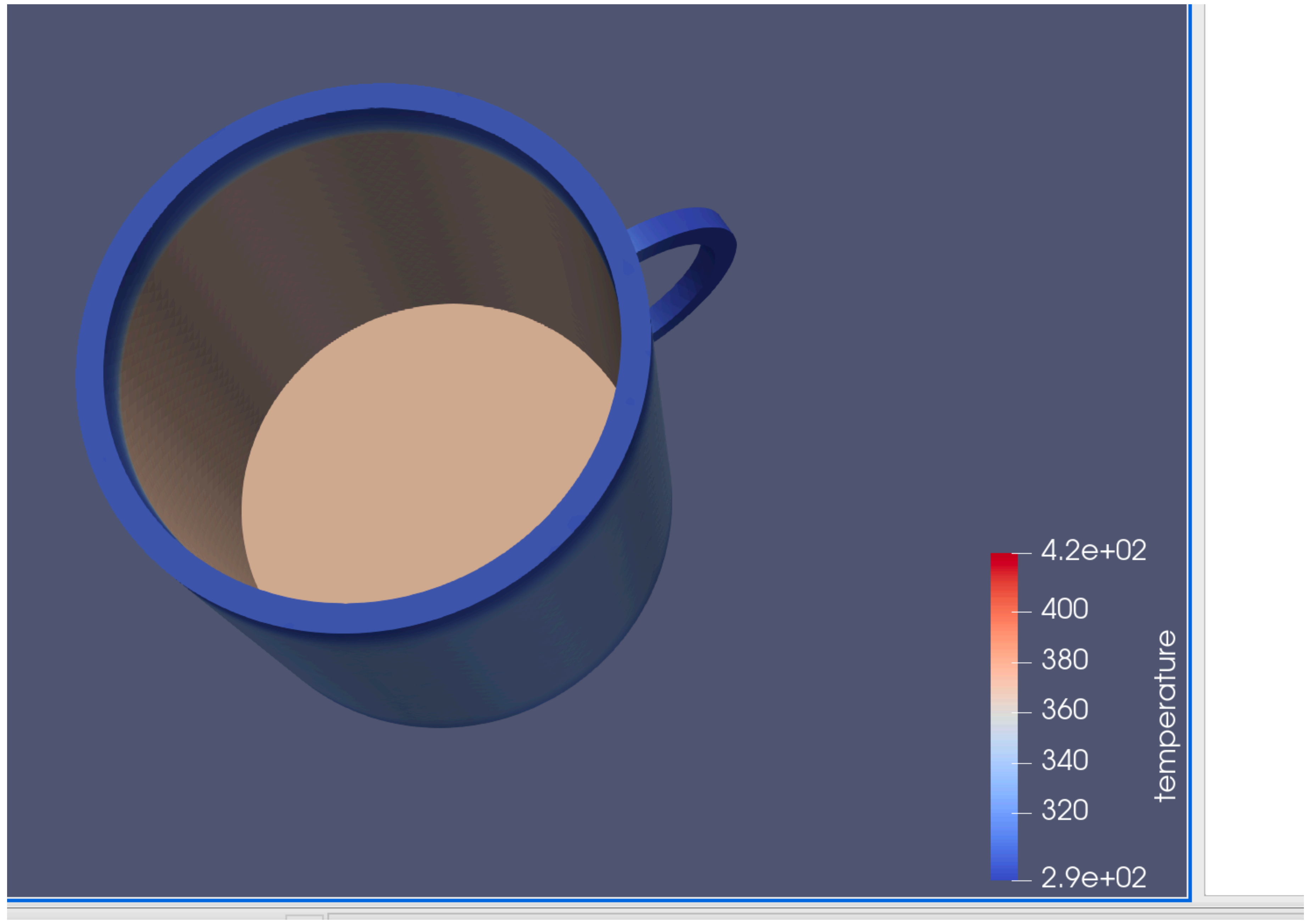
Input files ranging from 2-6 MB ~ (400K - 1200K tokens)

Outputs

Output files are a big problem.

Simulation output could range from several MBs to tons of GBs.

(I don't even want to know how many tokens that is).



```
/*--------------------------------*- C++ -*----------------------------------*\
| =========                 |                                                 |
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox           |
|  \\    /   O peration     | Version:  2406                                  |
|   \\  /    A nd           | Website:  www.openfoam.com                      |
|    \\/     M anipulation  |                                                 |
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    arch        "LSB;label=32;scalar=64";
    class       volVectorField;
    location    "150";
    object      U;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

dimensions      [0 1 -1 0 0 0 0];

internalField   nonuniform List<vector>
10720
(
(0.413217 28.1372 7.29312e-18)
(2.21028 30.4123 -5.75307e-17)
(4.45055 32.601 2.15392e-16)
(7.0167 34.287 -4.09916e-16)
(9.84621 35.5632 3.14852e-16)
(12.8727 36.3859 1.98827e-16)
(15.9396 36.6518 -3.57752e-16)
(18.954 36.402 1.22346e-15)
(21.8377 35.6988 -8.22691e-16)
(24.5129 34.581 -5.67868e-16)
(26.951 33.1302 2.90718e-16)
(29.1393 31.4337 1.62379e-15)
(31.0533 29.5212 6.26745e-16)
(32.6827 27.4237 5.67626e-16)
(34.0153 25.1706 1.33937e-15)
(35.0632 22.8073 1.14002e-15)
(35.814 20.3526 -2.26369e-15)
(36.2711 17.8579 -7.75469e-15)
```

Pros and Cons

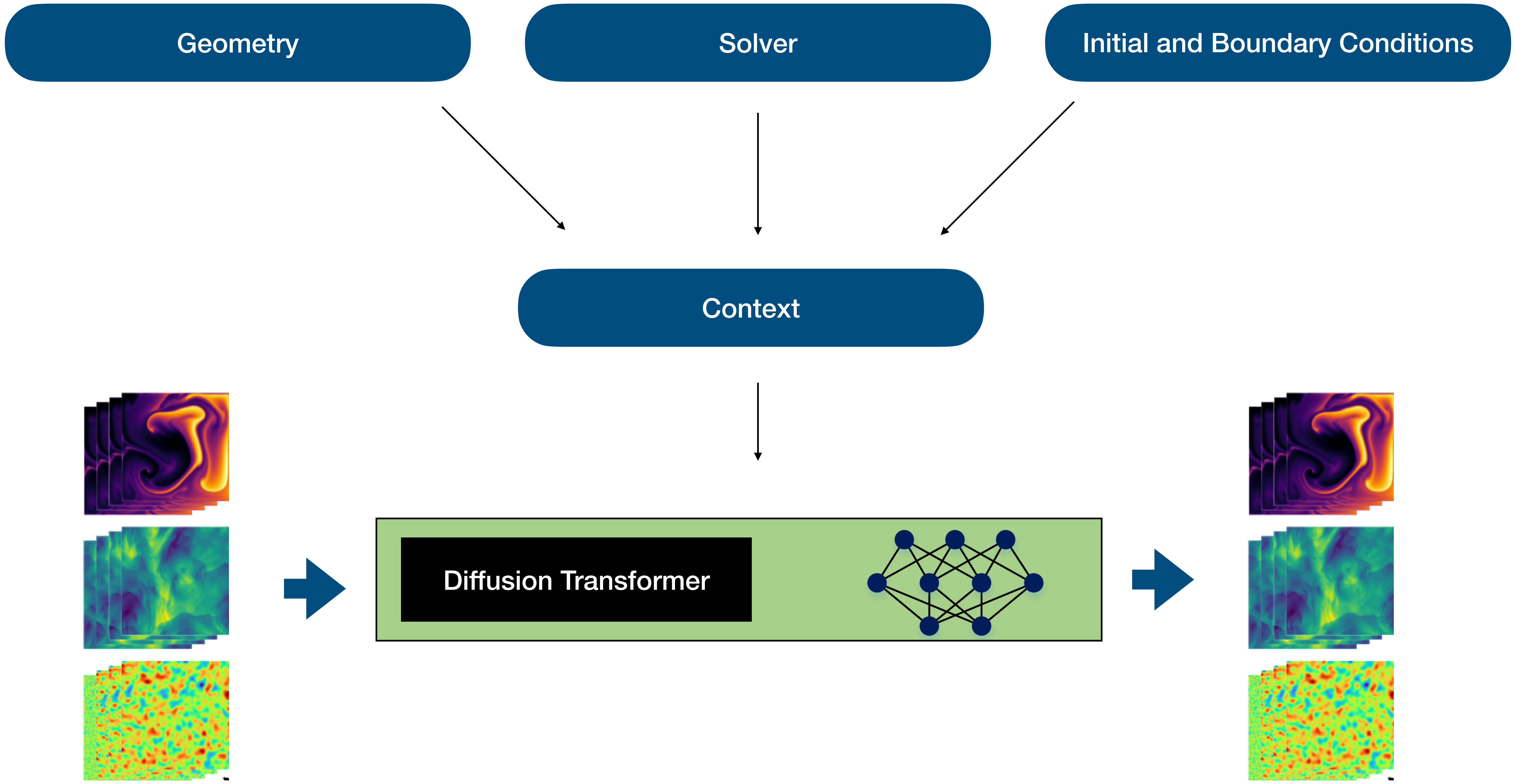
Advantages

- Integrates with existing solver
- Pre-processing pipeline is made simpler
- Complex geometry and multi-physics
- Works on top of existing models.
- Takes advantage of explicit PDE definitions.

Disadvantages

- Context Window Size
- Throwing out all inductive bias
- EXPENSIVE
- Accuracy ??

Text to Video then ?


Geometry
Solver
Initial and Boundary Conditions
Context
Diffusion Transformer

Can LLMs solve Fusion ?

Can Humans ?

Vignesh Gopakumar
UKAEA / UCL / Simvue

SML Research Day
27th June, 2024