Indian Institute of Engineering, Science and Technology, Shibpur

Hand Written Mathematical Expression Recognition

Project Members

Aratrika Pal (2020CSB007) Riya Manna (2020CSB018)

Astik Gorai (2020CSB038) Indranil Bain (2020CSB039)

Junayet Islam Sarkar (2020CSB020) Munshi Fahimuzzaman (2020CSB047)

Palla Pawan (2020CSB055)

Under Guidance Of

Prof. Sekhar Kumar Mondal

Department of Computer Science and Technology

-: Introduction :-

1. Challenges in Handwritten Mathematical Expression Recognition:

- Offline recognition poses challenges due to lack of temporal data and diverse writing styles.

2. Novel Approach:

- Utilises encoder-decoder model with paired adversarial learning.

3. Architecture:

- Encoder: VIT (Vision Transformer).
- Decoder: GPT (Generative Pre-trained Transformer).
- Enables learning of semantic-invariant features and contextual information.

4. Strengths Utilization:

- Combines VIT's contextual understanding with GPT's coherent output generation.

5. Results:

- Achieved significant improvement in recognition rates over prior methods.
- Demonstrates effectiveness in tackling challenges of offline handwritten mathematical expression recognition.

-: Introduction :-

Input

hand written mathematical expression images

MODEL

Output

Latex Code of the corresponding mathematical expression

x = r \cos \θ

-: Related Works :-

Grammar-Based HMER Techniques

Components:

Symbol Segmentation → Symbol Recognition → Structural Analysis

Key Approaches:

Stochastic Context-Free Grammars
Relational Grammars
Definite Clause Grammars

Limitations:

Reliance on predefined rules
Struggles with complex or unusual symbol layouts

-: Related Works :-

Early Encoder-Decoder Models for HMER

Foundational Encoder-Decoder Approaches

Basic Principle:

Encoding handwritten text images into latent space

Decoding into a sequence of mathematical symbols

Developments:

Introduction of attention mechanisms

Outcomes:

Baseline models for further refinement

Cutting-Edge Developments and Hybrid Models

Latest Trends and Hybrid Approaches:

- Integration of Machine Learning Techniques
- Use of deep learning for feature enhancement
- Hybrid Models Combining Grammar and Encoder Techniques
- Merging strengths of both approaches for superior accuracy

-: Related Works :-

Right - To - Left Language Modelling

- **Introduction to R2L Modeling:**
  - Traditional models primarily focus on Left-to-Right (L2R) language modelling.
  - To enhance modelling capabilities, Right-to-Left (R2L) approaches are integrated.

- **Model Training and Interaction:**
  - Separate Training: Models for L2R and R2L are trained separately to capture linguistic patterns specific to each direction.
  - Re-ranking during Inference: Outputs from both L2R and R2L models are considered during inference, re-ranked to produce the best result.

- **Model Improvement and Bidirectional Approach:**
  - Integration of R2L with L2R: R2L models are used to regularize and improve the performance of L2R models during the training phase.
  - Bidirectional Language Modelling: Implements a concise approach using a single decoder capable of generating sequences bidirectionally, maintaining simplicity while enhancing model capabilities.

-: Dataset Collection :-

Training and Testing Data Distribution :

There the two folders: one for training and one for testing.

Training Data:

Total number of images for training: **8835**

Training Dataset contains two-column structure in the training data folder: image name and corresponding LaTeX code.

Testing Data:

Total number of images for testing: **986**. the variability in image sizes across the testing set.

-: Dataset Collection :-

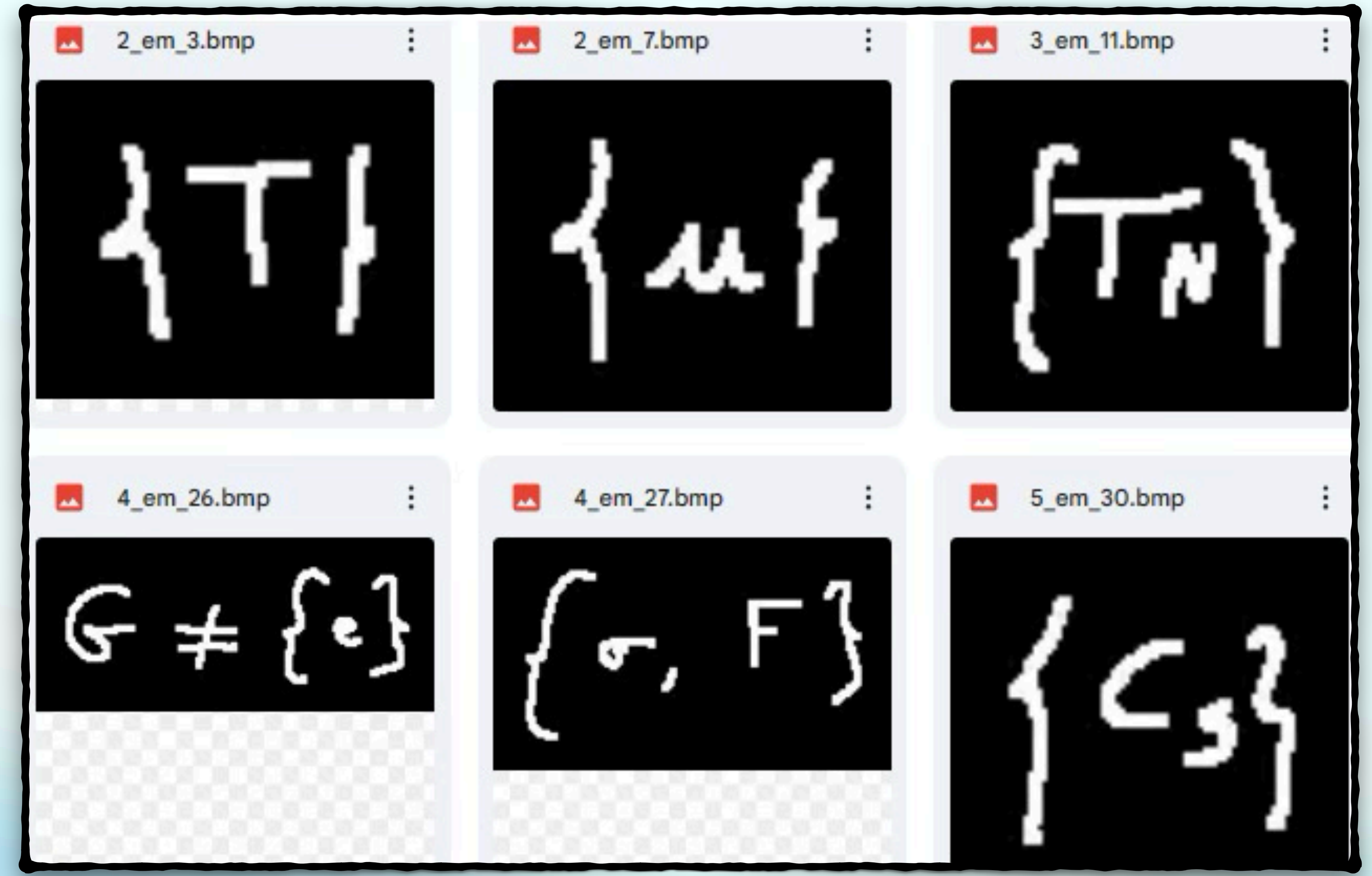


Fig 1

Mathematical expression images

2_em_3	\{ T \}
2_em_7	\{ u \}
3_em_11	\{ T _ { N } \}
3_em_18	\{ I _ { k } \}
4_em_22	\{ I , \sigma \}
4_em_26	G \neq \{ e \}
4_em_27	\{ \sigma , F \}
5_em_30	\{ c _ { s } \}
7_em_59	\forall g \in G
8_em_62	\sigma \in G
8_em_65	\forall i \in I
9_em_71	T ^ { \mu } _ { \mu }
9_em_76	\phi \in S
9_em_77	H \in P

Fig 2

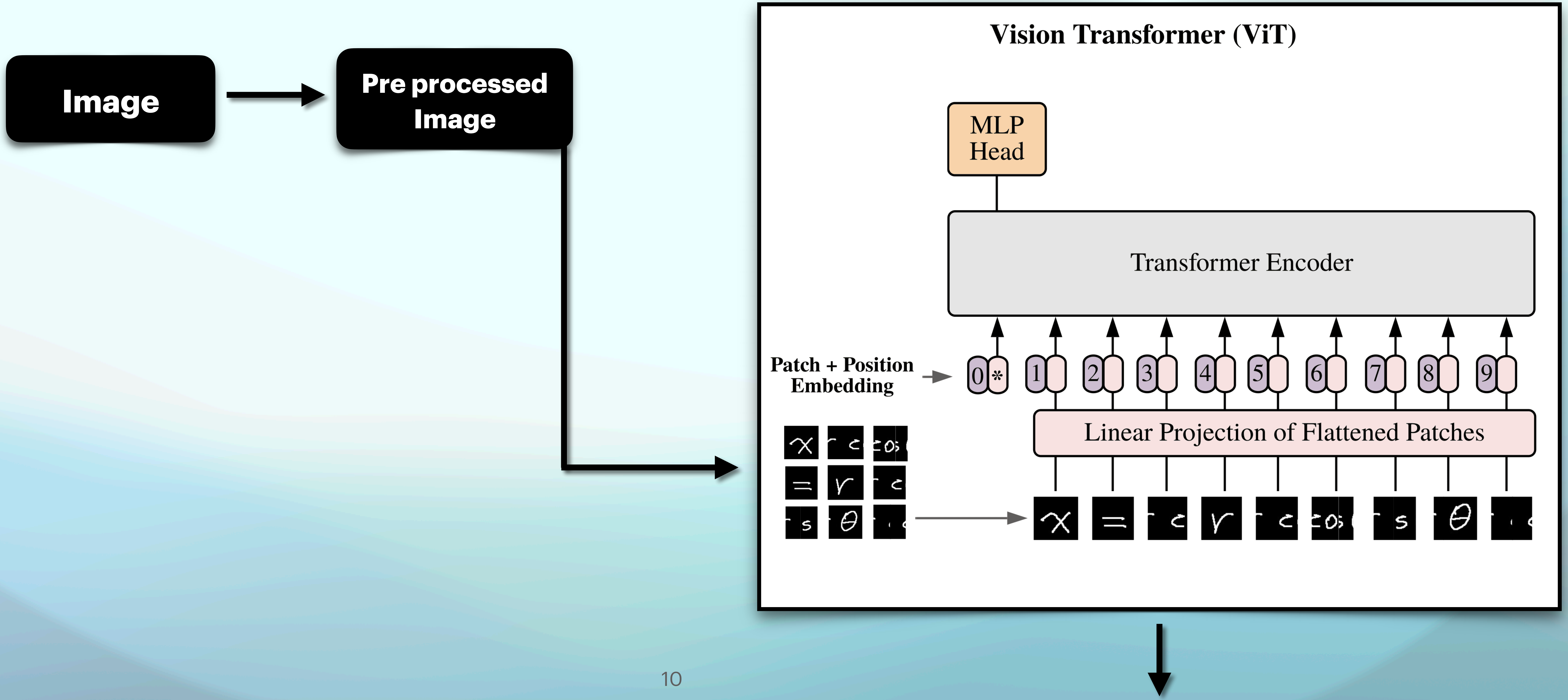
Corresponding Latex Code

-: Preprocessing Stages :-

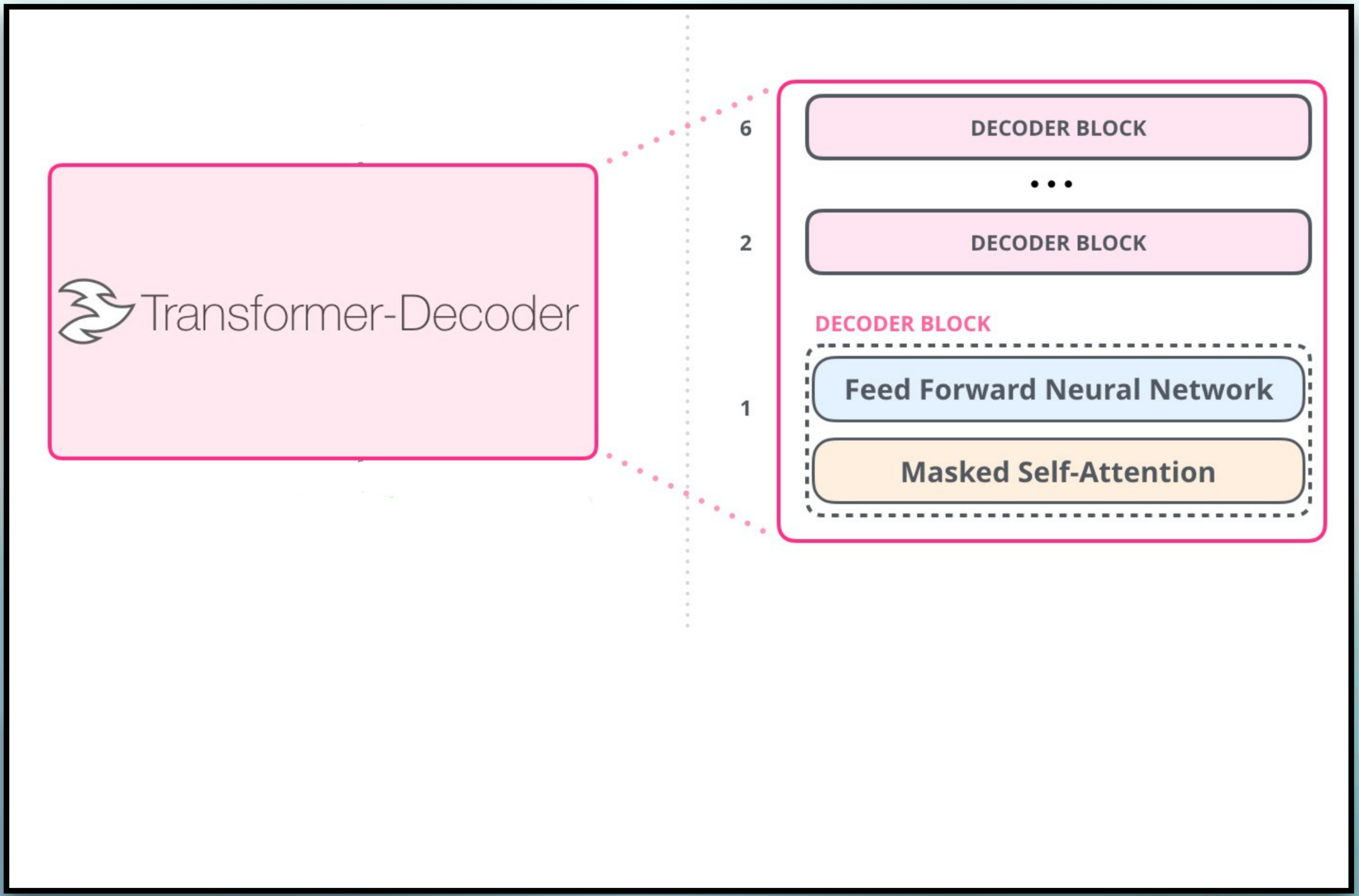
$$\text{IMAGE} \xrightarrow[1]{\text{Normalise}} \left\{ \text{Normalised Image} \right\} \xrightarrow[2]{\text{Resize}} \left[\text{Normalised + Resized Image} \right] \xrightarrow[3]{\text{Grayscale Conversion}} \text{Preprocessed Image}$$

-: Model Architecture :-

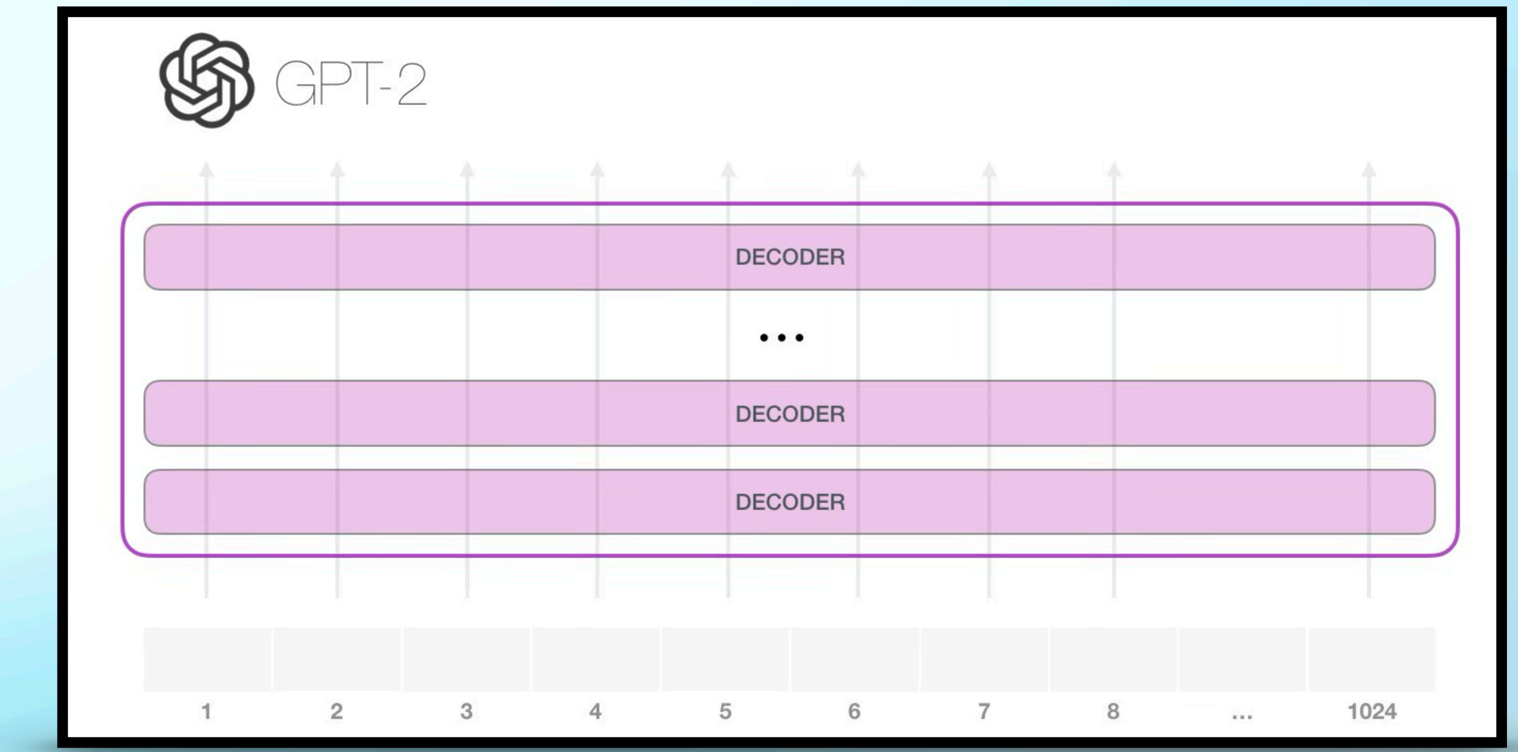
VIT Model



-: Model Architecture contd.. :-



Processed VIT Output + **Latex Code**



Predicted Latex code

-: Experimental Analysis :-

Bleu Score:

- The BLEU score measures the similarity between machine-generated translations and human reference translations, with 1 representing a perfect match.

- It's a metric commonly used in evaluating the quality of machine translation systems.

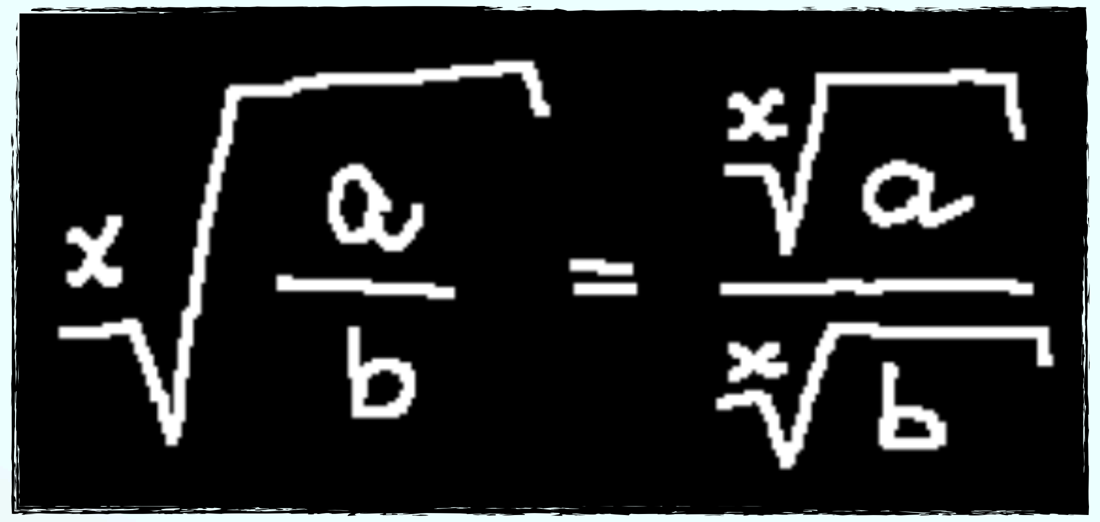
The brevity penalty (BP) accounts for the difference in length between the candidate translation and the reference translation(s), penalising shorter translations.

$$\mathrm{BLEU} = \mathrm{BP} \times \exp\left(\sum_{n=1}^{N} w_n \cdot \log(\mathrm{precision}_n)\right)$$

- BP is the brevity penalty,
- N is the maximum n-gram order considered (usually 4),
- w_n is the weight for the precision of n-grams (typically $\frac{1}{N}$),
- $\mathrm{precision}_n$ is the precision for n-grams.

-: Experimental Analysis :-

For example, for one of our dataset images we got the following output



Output

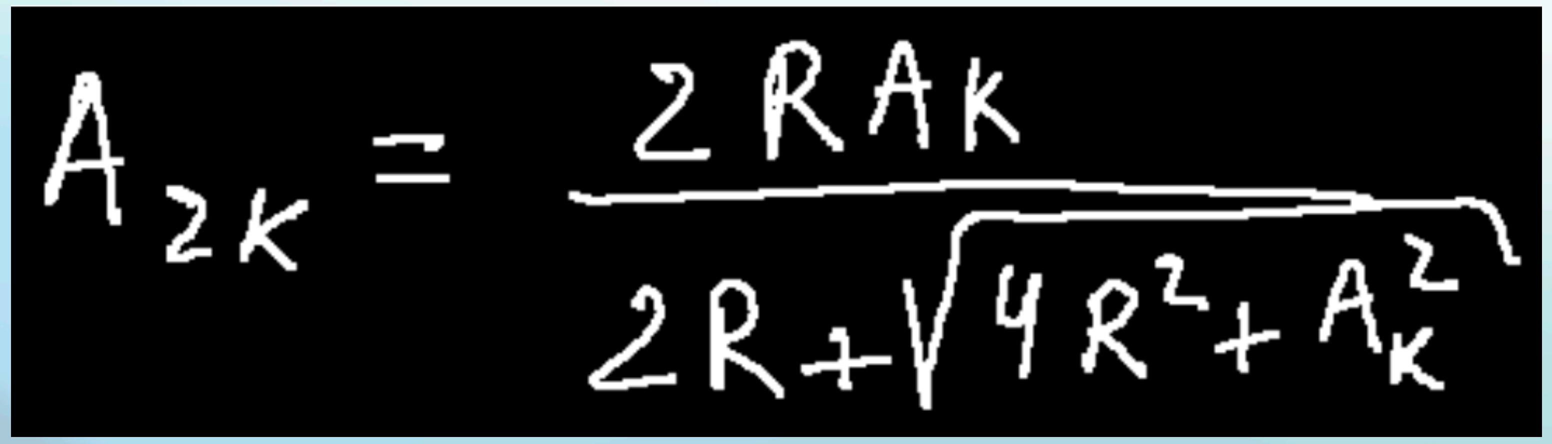
```
The predicted latex code for the image is:

xV \frac {a} {b} = \frac {\sqrt[11]{a}}{\sqrt[x]{b}}

Process finished with exit code 0
```

Output latex code:

xV \frac{a}{b} = \frac{\sqrt[11]{a}} {\sqrt[x]{b}}



```
The predicted latex code for the image is:

A_{2k} = \frac{2P \Delta k}{2R + \sqrt{9R^2 + A_k^3}}

Process finished with exit code 0
```

Output latex code:

A_{2k} = \frac{2R \Delta k}2R+\sqrt{9R^2 + A_K^2}}

**Using this metric we got bleu score as 0.7542*

-: Applications :-

- ✦ Educational Platforms
- ✦ Scientific Research
- ✦ Engineering and Technical Documentation
- ✦ Data Analysis and Visualisation
- ✦ Accessibility Tools
- ✦ Publishing and Digital Media
- ✦ Mathematical Software and Libraries
- ✦ User Experience and Interface Design

-: *Drawbacks* :-

- Complexity of Mathematical Expressions
- Limited Dataset variability
- Dependency on Image Quality
- Interpretability and Error Handling
- Integration and Compatibility

-:Conclusion:-

- **Marks a critical turning point** in the field of mathematical expression extraction from images.
- Utilises cutting-edge technologies such as **Vision Transformer (ViT) and Generative Pre-trained Transformer 2 (GPT-2)**.
- **Demonstrates effectiveness and dependability** in translating mathematical equations into LaTeX code.
- **Exhibits adaptability for various fields** including publishing, data analysis, engineering, research, teaching, and mathematical software development.
- **Emphasises inclusivity and accessibility features for users** with visual impairments and compatibility with assistive devices.
- **Streamlines workflows, enhances efficiency, and opens avenues** for future enhancements and integration into real-world applications.

-:Challenges:-

- **Computational Resource Requirement**
  - Processing handwritten mathematical expressions demands significant computational resources.
  - Resource-intensive algorithms are required for accurate recognition.
  - Efficient hardware and optimised software are vital for real-time processing.
- **Accuracy Calculation Challenges**
  - Traditional measures like precision, F1 score, and confusion matrix are insufficient.
  - The complexity of mathematical expressions poses unique challenges.
  - Novel approaches for accuracy assessment are necessary for meaningful evaluation.
- **Availability of large labeled datasets is limited.**
  - Training models with small datasets requires careful handling.
  - Techniques such as data augmentation and transfer learning are crucial for performance enhancement.

-: References :-

1. **The Illustrated Transformer - Jay alamar**
    - https://jalammar.github.io/illustrated-transformer/
2. **The Illustrated GPT-2 (Visualizing Transformer Language Models)**
    - https://jalammar.github.io/illustrated-gpt2/
3. **Handwritten Mathematical Expression Recognition with Bidirectionally Trained Transformer**

- https://arxiv.org/pdf/2105.02412

4. **A global learning approach for an online handwritten mathematical expression recognition system**

- https://www.sciencedirect.com/science/article/abs/pii/S0167865512003546?casa_token=oLM-X2eYiMoAAAAA:7hWsu1_dJeM7TW86YjeLVNwZ5uvago4Ot6Jht7i_gmTCeUrSfpDLfv8qIPdeYal8zFYXMJpf3oo

Thank you :)