# DEEP LEARNING-BASED SOFTWARE ENGINEERING

Citation: Chen X P, Hu X, Huang Y, et al. Deep learning-based software engineering: progress, challenges, and opportunities. Sci China Inf Sci, 2025

Presented By: Rahul Kavati

08 May, 2025

# Introduction

- Rise of deep learning (DL) as a transformative force in software engineering (SE)
- Shift from manual, rule-based approaches to data-driven automation
- Overview of paper's goal: survey DL’s impact on 12 key SE tasks

# Objectives of the Study

- Map deep learning techniques to SE subareas
- Highlight research progress, challenges, and opportunities
- Provide a unified view to guide future AI4SE research

# Deep Learning Techniques

- CNNs, RNNs, LSTMs, Transformers
- Pre-trained models: BERT, CodeBERT, GraphCodeBERT, UnixCoder
- Application across code, text, graphs

# Software Engineering Tasks

01

Requirements engineering

02

Code generation

03

Code summarization

04

Defect prediction

05

Bug finding

06

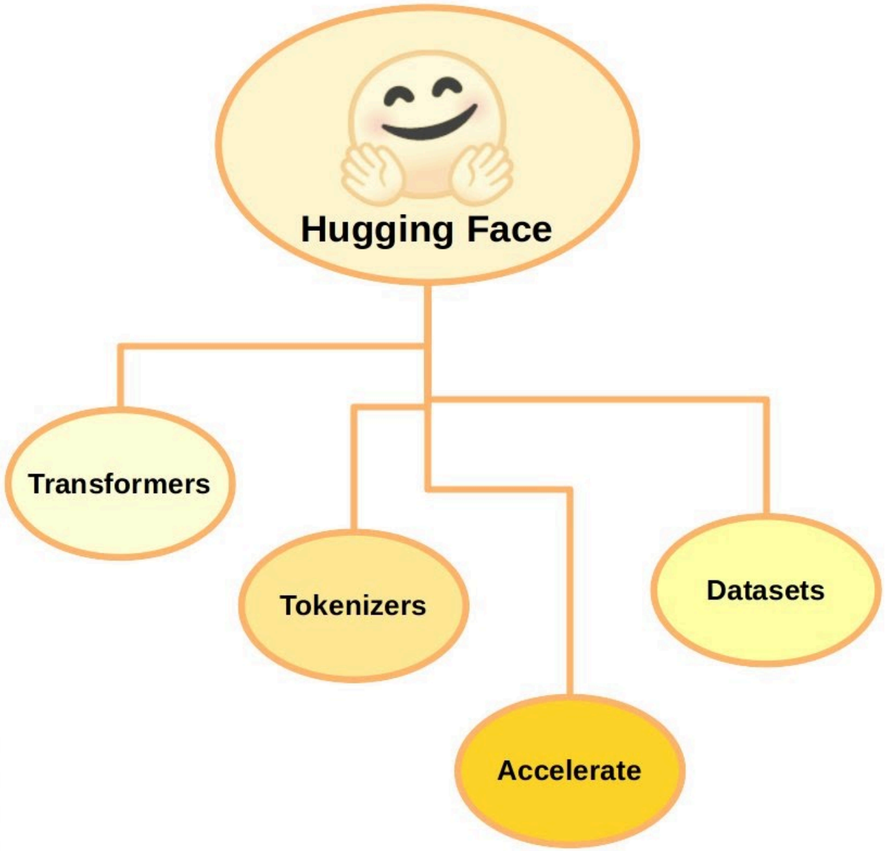
Fault localization

07

Bug report management

# Code Summarization

- Purpose: Generate natural language descriptions of code
- DL Techniques:
  - Seq2Seq (LSTM) models
  - Transformer-based models (e.g., CodeBERT, UnixCoder)
  - Multi-modal input: code tokens, ASTs, CFGs
- Benefits:
  - Enhances code comprehension and maintenance
  - Supports automatic documentation
- Challenges:
  - Quality and consistency of training data
  - Evaluating summary accuracy
  - Avoiding hallucinations or misleading summaries



# Contributions of the paper

- Task-oriented classification of DL applications in SE
- Analysis of over 600 papers
- Discussion of models, datasets, tools, and gaps
- Emphasis on cross-cutting themes (e.g., explainability, scalability)

# Research challenges Identified

- Dataset scarcity and labeling effort
- Generalization across projects/languages
- Integration with development workflows
- Trust, explainability, and reproducibility

# Proposed Framework

- Model-task alignment guidance
- Importance of benchmark datasets
- Advocates for explainable DL in SE
- Promotes reuse of knowledge via pre-trained models

# Conclusions

- DL is reshaping SE with substantial progress in many areas
- Code summarization among key areas of impact
- Still, major challenges remain in data, tools, and evaluation

# Future works

- Build diverse, high-quality datasets
- Improve interpretability of DL models
- Apply LLMs to collaborative and interactive SE tasks
- Study real-world adoption and trust in AI-generated artifacts

# THANK YOU

08 May, 2025