On the Evaluation of Neural Code Translation: Taxonomy and Benchmark

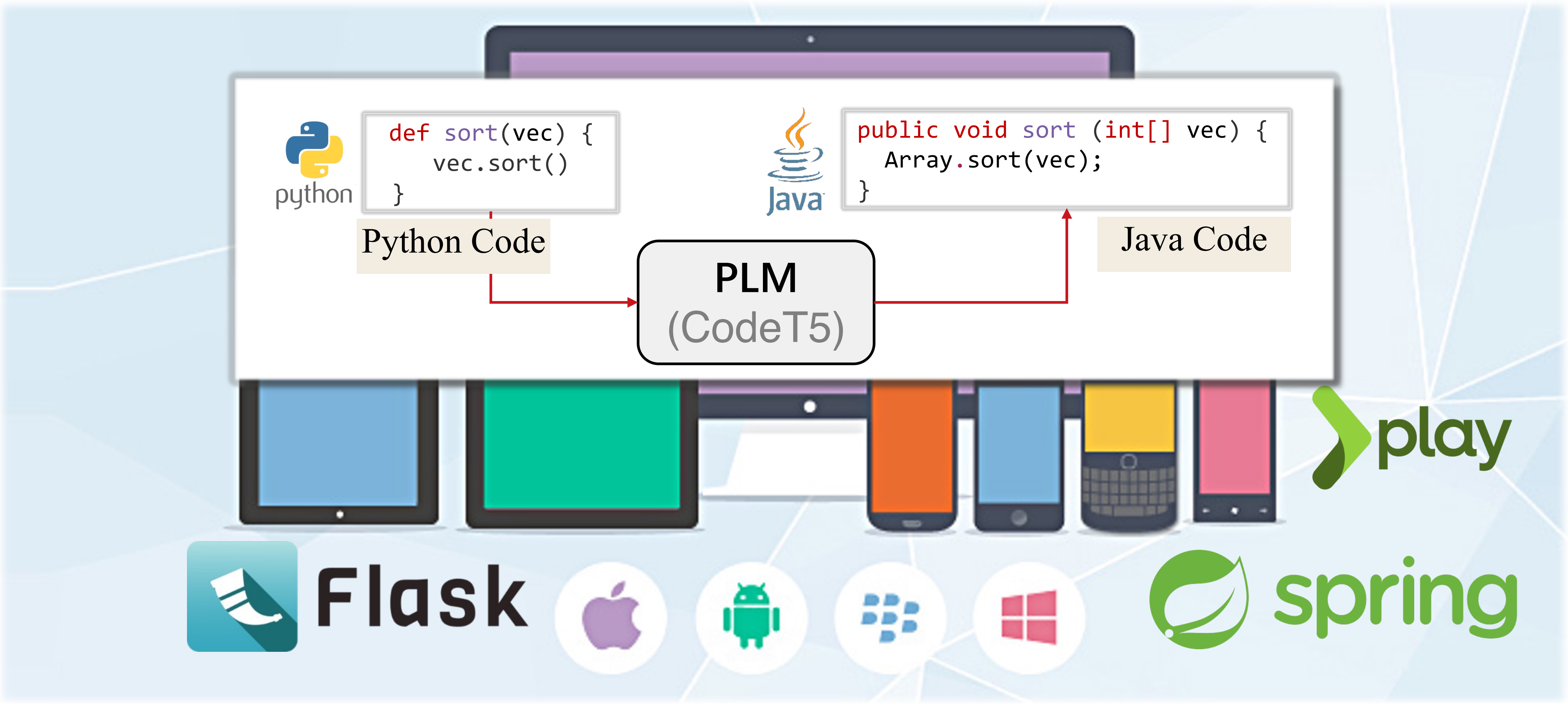
Mingsheng Jiao, Tingrui Yu, Xuan Li, Guanjie Qiu, Xiaodong Gu, Beijun Shen

School of Software, Shanghai Jiao Tong University

上海交通大學
SHANGHAI JIAO TONG UNIVERSITY

Neural Code Translation

Neural Code Translation



Motivation

python	Java	CodeBLEU
`def sort(vec) { vec.sort() return vec }`	`public sort(arr) { for(i=0;…) arr[i]=arr[j]; }`	98
`def split(str) { str.split() }`	`public seg(s) { String.split() }`	86
`def format(s) { s=s[1]+':'+s[2] print(s) }`	`public reform(str) { s.chatAt[1]+':".. system.out..(s) }`	91
		⋮

Motivation

python	Java	CodeBLEU
`def sort(vec) { vec.sort() return vec }`	`public sort(arr) { for(i=0;…) arr[i]=arr[j]; }`	98
`def split(str) { str.split() }`	`public seg(s) { String.split() }`	86
`def format(s) { s=s[1]+':'+s[2] print(s) }`	`public reform(str) { s.chatAt[1]+':".. system.out..(s) }`	91
		⋮

```
def sort(vec) {
    vec.sort()
    return vec
}
```

```
public sort(arr) {
    for(i=0;…)
    arr[i]=arr[j];
}
```

```
def split(str) {
    str.split()
}
```

```
public seg(s) {
    String.split()
}
```

```
def format(s) {
   s=s[1]+':'+s[2]
   print(s)
}
```

```
public reform(str) {
    s.chatAt[1]+':"..
    system.out..(s)
}
```

Are they good or not?

Motivation

Translation

```
int kthSmallest(int arr[], int k) {
  sort(arr, arr + n);
  return arr[k-1];
}
```

Undeclared !

BLEU=89.66

Target

```
int kthSmallest(int arr[], int n,
  int k) {
  sort(arr, arr+n);
  return arr[k-1];
}
```

Translation

```
int findReapting(int arr[], int n) {
  int sum = 0;
  for (int i = 0; i<n; i ++)
    sum += arr[i];
  return sum-(((n-1) * n)/2);
}
```

BLEU=41.56

Target

```
int findReapting(int arr[], int n) {
  return accumulate(arr, arr+n, 0)
         - ((n - 1) * n / 2);
}
```

An overall score may not capture the fine-grained capabilities of code translation models especially on difficult translations.

Outline

01

Empirical Study

02

Taxonomy

Fine-grained Evaluation of Code Translation Models

03

Benchmark

04

Experiments

Outline

01
Empirical Study

02
Taxonomy

03
Benchmark

04
Experiments

Empirical Study

RQ1: How is the **fine-grained performance** of state-of-the-art code translation models?

RQ2: Can existing benchmarks exhibit the **fine-grained capability** of code translation models?

Experimental Setup

- Models
  - CodeBERT
  - CodeT5
  - TransCoder
  - TransCoder-ST

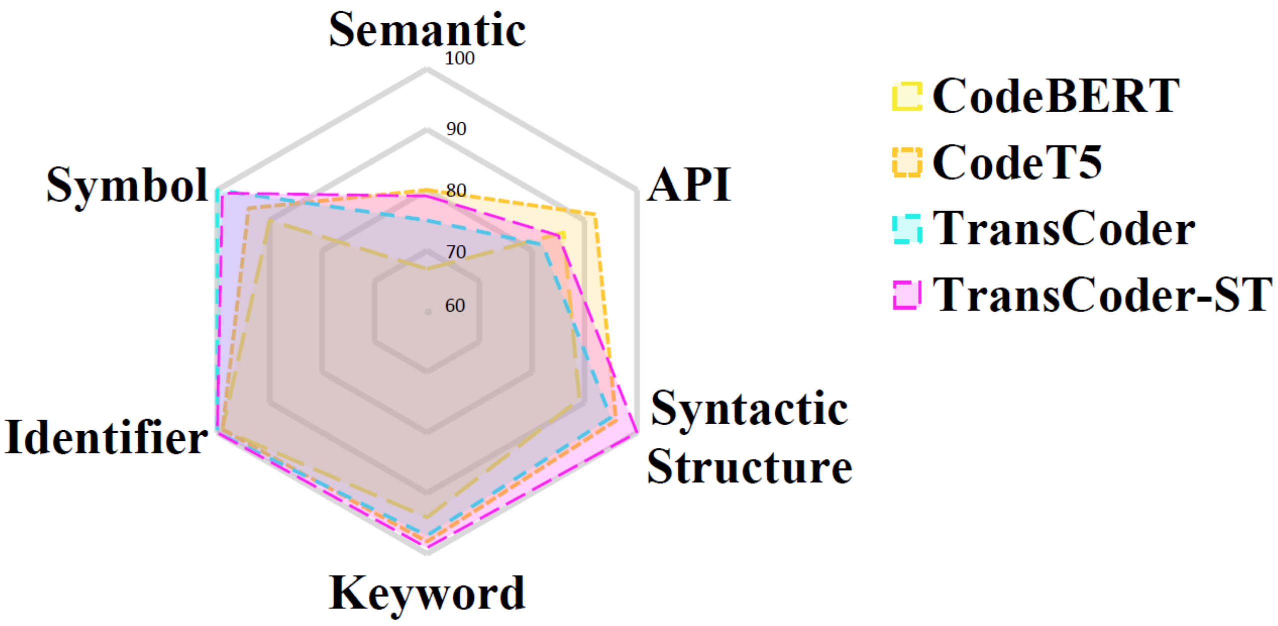
- Benchmarks
  - CodeXGLUE
  - TransCoder-test
  - XLCoST

- Metrics
  - BLEU
  - CodeBLEU
  - CA (Computational Accuracy)

RQ1: Fine-grained Performance of SOTA Code Translation Models

Fine-grained Aspects:

- **Easy**: Keyword & Identifier
- **Middle**: Syntactic Structure & Symbol
- **Difficult**: API & Semantic



Answer to RQ1: State-of-the-art code translation models exhibit varying translation capabilities in fine-grained aspects, with a greater proficiency in translating tokens, followed by syntax, APIs, and semantics.

RQ2: Distinguishing Ability of Existing Benchmarks

Features of code under different ranges of BLEU

BLEU	Percentage	Characteristics of Code
80-100	84.4 88.7 83.4	Basic data types Simple condition statements Arithmetic operations Simple function calls
50-80	13.4 10.9 12.0	Basic data structures Diverse conditions statements Fewer arithmetic operators Multiple API calls
0-50	2.2 0.4 4.6	Complex variable types Longer and informative identifiers Manipulation of complex variables Complex API calls Difference in algorithm

TransCoder-test
XLCoST
CodeXGLUE

Answer to RQ2: Existing benchmarks are biased towards trivial translations, such as token mapping and are limited in complex translations, such as library invocation and algorithm rewriting.

Outline

01
Empirical Study

02
Taxonomy

03
Benchmark

04
Experiments

Taxonomy on Code Translation

Taxonomy	Description	Definition
Type 1	Token-level translation	Map trivial tokens to their equivalent in the target
Type 2	Syntax-level translation	Migrate syntactic structures based on linguistic rules
Type 3	Library-level translation	Migrate library to their equivalent in the target language
Type 4	Algorithm-level translation	Reimplement the program in the target language using a different algorithm

Taxonomy – Examples

TYPE 1

C++

```cpp
int maxProductSubset(int a[], int n){
  if(n == 1) return a[0];
  int max_neg = INT_MIN;
  int prod = 1;
  for(int i = 0; i < n; i++) {
    //... Rest of the Code
  }
  return prod;
}
```

Java

```java
int maxProductSubset(int a[], int n) {
  if(n == 1) return a[0];
  int max_neg = Integer.MIN_VALUE;
  int prod = 1;
  for(int i = 0; i < n; i++)
    //... Rest of the Code
  }
  return prod;
}
```

TYPE 2

C++

```cpp
int countWays(string s) {
  int count[26] = {0};
  for(char x : s)
    count[x - 'a']++;
  count[s[0] - 'a'] = 1;
  int ans = 1;
  //... Rest of the Code
  return ans ;
}
```

Java

```java
int countWays(String s) {
  int count[] = new int[26];
  for(int i = 0; i < s.length(); i++)
    count[s.charAt(i)-'a']++;
  count[s.charAt(0)-'a']=1;
  int ans = 1;
  //... Rest of the Code
  return ans ;
}
```

TYPE 3

C++

```cpp
int removeConsecutiveSame(vector<string> v){
  stack<string> st;
  for(int i = 0; i < v.size(); i++){
    if(st.empty()) st.push(v[i]);
    else {
      string str = st.top();
      if(str.compare(v[i])==0) st.pop();
      else st.push(v[i]);
    }
  }
  return st.size();
}
```

Java

```java
int removeConsecutiveSame(Vector<String> v){
  Stack<String> st = new Stack<>();
  for(int i = 0; i < v.size(); i++){
    if(st.empty()) st.push(v.get(i));
    else {
      String str = st.peek();
      if(str.equals(v.get(i))) st.pop();
      else st.push(v.get(i));
    }
  }
  return st.size();
}
```

TYPE 4

C++

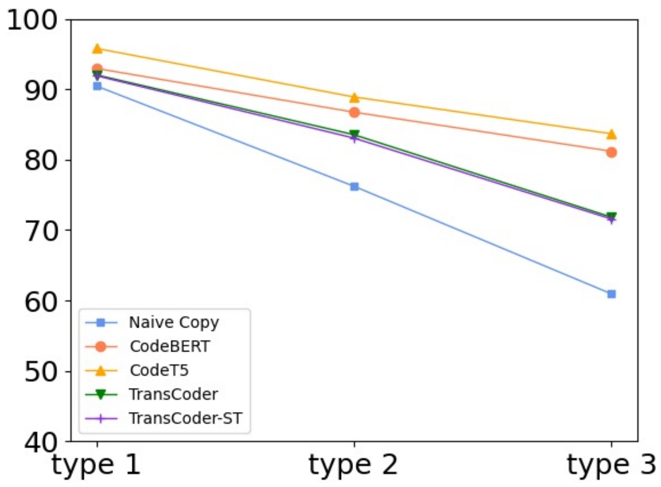
```cpp
int calFactorial (int n){
  int result = 1;
  for(int i=1; i<=n; ++i)
    result *= i;
  return result;
}
```

Java

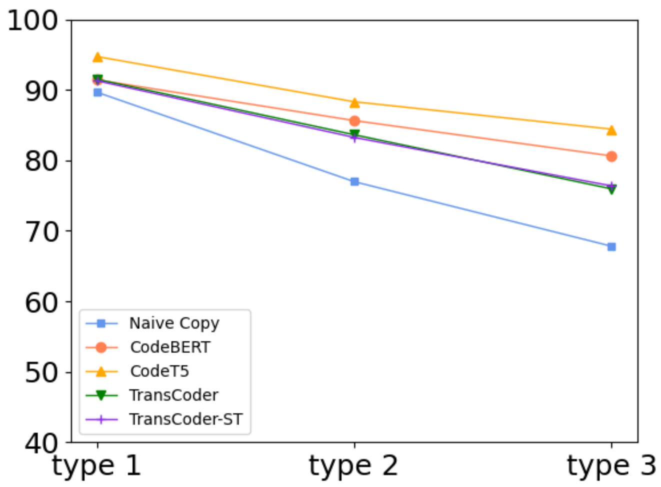
```java
int calFactorial (int n){
  if(n == 1 || n == 0) return 1;
  return n*calFactorial(n-1);
}
```

Performance of models within our taxonomy

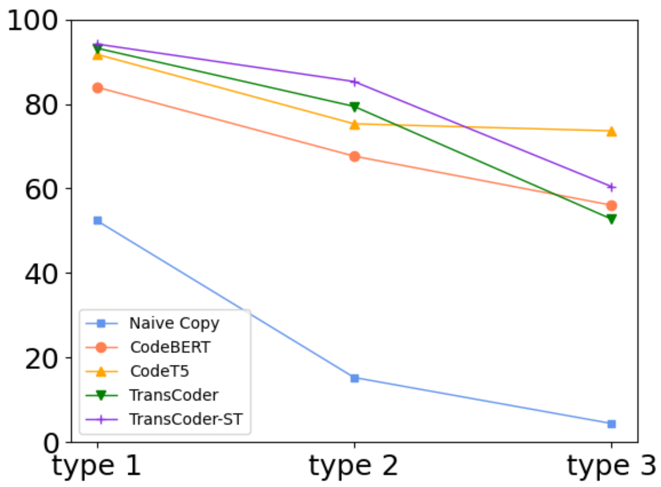
Performance of various models under different translation types of Java→C++ translations.



BLEU



CodeBLEU



CA

- Categories in our taxonomy indeed differentiate the increasing complexity of code translation
- More complex and diverse benchmarks are required for finer-grained model evaluation

Outline

01

Empirical Study

02

Taxonomy

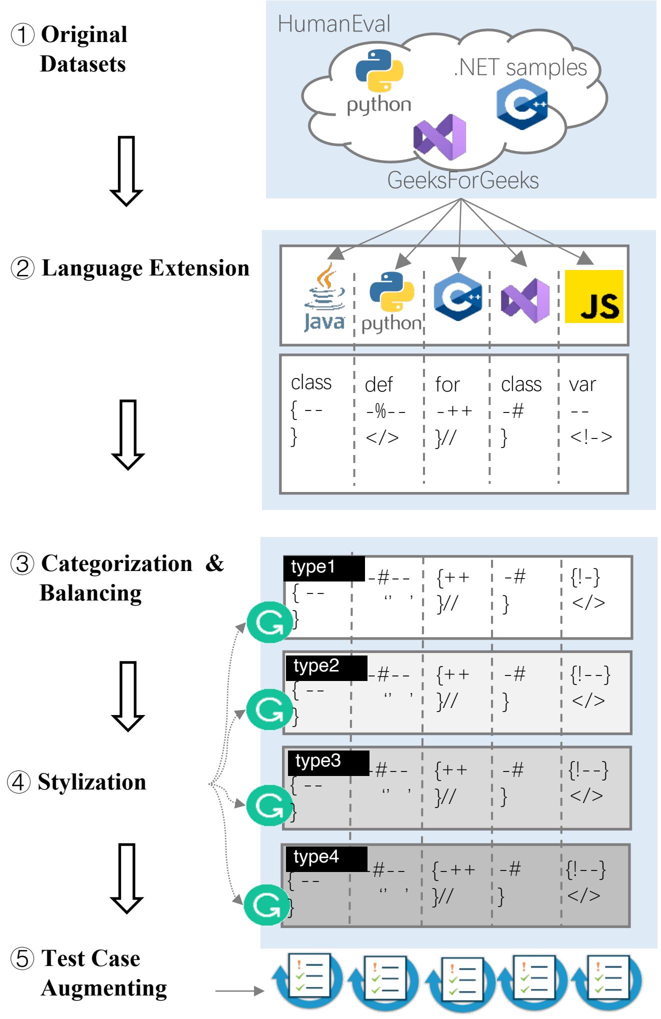
03

Benchmark

04

Experiments

Benchmark Construction: G-TransEval



- **Step1**: Collect sample codes from diverse sources
- **Step2**: Expand monolingual code to five programming languages
- **Step3**: Partition the dataset into four subsets based on the taxonomy
- **Step4**: Normalize coding stylization following Google style conventions
- **Step5**: Write test cases for each code sample

Comparison with Prior Benchmarks

Dataset	Source	Parallel Data Size (train/valid/test)	Languages	Categorized?	Style Normalized?	Unit Tests Included?	Golden Answer Verified?
CodeXGLUE	Lucune, POI, JGit, Antlr	10,253 / 499 / 1,000	Java, C#	✗	✗	✗	✗
XLCoST*	G4G	9450 / 490 / 901	C++, Java, C#, PHP, JavaScript, Python, C	✗	✗	✗	✗
TransCoder-test	G4G	- / 470 / 948	C++, Java, Python	✗	✗	Partial	✗
HumanEval-X	HumanEval	- / - / 164	C++, Java, Go, JavaScript, Python	✗	✓	✓	✓
G-TransEval	HumanEval, G4G, .NET samples	- / - / 400	C++, Java, C#, JavaScript, Python	✓	✓	✓	✓

- **G-TransEval vs CodeXGLUE**: multilingual solutions
- **G-TransEval vs. XLCoST**: more complex and including unit test cases
- **G-TransEval vs. TransCoder-test**: including full test cases and verified solutions
- **G-TransEval vs. HumanEval-X**: categorized and more balanced distribution of four types

Outline

01
Empirical Study

02
Taxonomy

03
Benchmark

04
Experiments

SJTU

1) Effect of taxonomy

Comparison of model performance on the benchmark

Model	Java→Python			Python→Java			Java→C++			C++→Java			Java→JavaScript			JavaScript→Java		
	BLEU	CB	CA	BLEU	CB	CA	BLEU	CB	CA	BLEU	CB	CA	BLEU	CB	CA	BLEU	CB	CA
Type 1																		
CodeBERT	81.37	82.67	78.40	81.26	84.14	67.20	93.83	94.18	83.20	95.68	95.54	84.00	83.77	84.91	78.40	**93.47**	93.52	72.80
CodeT5	82.71	83.07	**88.00**	81.98	84.81	78.40	**94.14**	**94.41**	90.40	**97.39**	**97.35**	94.40	**84.67**	**85.25**	**85.60**	93.46	**93.81**	**76.80**
TransCoder	86.28	83.73	57.60	82.82	85.32	78.40	89.73	90.61	94.40	93.55	94.22	92.80	-	-	-	-	-	-
TransCoder-ST	**90.12**	**88.66**	80.80	**90.86**	**91.94**	**88.00**	87.95	88.78	**97.60**	94.50	95.15	**95.20**	-	-	-	-	-	-
Type 2																		
CodeBERT	77.52	77.70	53.60	**69.75**	68.04	33.60	89.19	89.17	62.40	77.83	76.30	55.20	80.22	79.57	55.20	77.02	74.01	42.40
CodeT5	78.90	79.11	74.40	69.57	**68.91**	50.40	**91.33**	**91.47**	**72.80**	**82.15**	**81.28**	**83.20**	**82.74**	**82.06**	**73.60**	**80.35**	**78.07**	**62.40**
TransCoder	84.39	84.37	60.80	65.13	65.18	46.40	83.76	84.95	69.60	69.50	68.77	57.60	-	-	-	-	-	-
TransCoder-ST	**87.38**	**87.60**	**76.00**	66.11	64.82	**51.20**	83.93	84.85	71.20	70.11	69.55	55.20	-	-	-	-	-	-
Type 3																		
CodeBERT	74.51	74.38	28.80	63.69	62.96	16.80	79.14	80.64	31.20	71.81	69.49	26.40	72.56	72.13	25.60	72.03	68.56	20.00
CodeT5	78.62	78.95	68.00	67.47	67.61	**44.80**	**83.66**	**84.67**	**48.00**	**74.52**	**74.26**	**58.40**	**75.18**	**75.71**	**52.80**	**74.26**	**72.04**	**37.60**
TransCoder	78.04	77.71	26.40	65.00	63.76	19.20	74.83	78.02	38.40	68.86	66.27	33.60	-	-	-	-	-	-
TransCoder-ST	**84.42**	**84.15**	**69.60**	**70.84**	**69.14**	38.40	76.20	79.26	40.80	68.45	67.69	36.80	-	-	-	-	-	-
Type 4																		
CodeBERT	35.89	37.10	0.00	26.32	30.22	0.00	25.18	33.53	0.00	20.38	28.05	0.00	34.05	36.48	0.00	22.88	27.93	0.00
CodeT5	37.08	39.82	0.00	21.79	28.13	0.00	**31.76**	**40.87**	0.00	**33.80**	45.19	0.00	**43.20**	**44.94**	**8.00**	**29.71**	**36.61**	0.00
TransCoder	37.71	39.35	0.00	**25.99**	**34.39**	0.00	19.91	30.09	0.00	31.75	**48.37**	0.00	-	-	-	-	-	-
TransCoder-ST	**50.99**	**49.50**	**4.00**	24.36	29.51	0.00	24.96	34.78	**4.00**	33.38	43.06	0.00	-	-	-	-	-	-

1) Effect of taxonomy

Comparison of model performance on the benchmark

Model	Java→Python			Python→Java			Java→C++			C++→Java			Java→JavaScript			JavaScript→Java		
	BLEU	CB	CA	BLEU	CB	CA	BLEU	CB	CA	BLEU	CB	CA	BLEU	CB	CA	BLEU	CB	CA
Type 1																		
CodeBERT	81.37	82.67	78.40	81.26	84.14	67.20	93.83	94.18	83.20	95.68	95.54	84.00	83.77	84.91	78.40	**93.47**	93.52	72.80
CodeT5	82.71	83.07	**88.00**	81.98	84.81	78.40	**94.14**	**94.41**	90.40	**97.39**	**97.35**	94.40	**84.67**	**85.25**	**85.60**	93.46	**93.81**	**76.80**
TransCoder	86.28	83.73	57.60	82.82	85.32	78.40	89.73	90.61	94.40	93.55	94.22	92.80	-	-	-	-	-	-
TransCoder-ST	**90.12**	**88.66**	80.80	**90.86**	**91.94**	**88.00**	87.95	88.78	**97.60**	94.50	95.15	**95.20**	-	-	-	-	-	-
Type 2																		
CodeBERT	77.52	77.70	53.60	**69.75**	68.04	33.60	89.19	89.17	62.40	77.83	76.30	55.20	80.22	79.57	55.20	77.02	74.01	42.40
CodeT5	78.90	79.11	74.40	69.57	**68.91**	50.40	**91.33**	**91.47**	**72.80**	**82.15**	**81.28**	**83.20**	**82.74**	**82.06**	**73.60**	**80.35**	**78.07**	**62.40**
TransCoder	84.39	84.37	60.80	65.13	65.18	46.40	83.76	84.95	69.60	69.50	68.77	57.60	-	-	-	-	-	-
TransCoder-ST	**87.38**	**87.60**	**76.00**	66.11	64.82	**51.20**	83.93	84.85	71.20	70.11	69.55	55.20	-	-	-	-	-	-
Type 3																		
CodeBERT	74.51	74.38	28.80	63.69	62.96	16.80	79.14	80.64	31.20	71.81	69.49	26.40	72.56	72.13	25.60	72.03	68.56	20.00
CodeT5	78.62	78.95	68.00	67.47	67.61	**44.80**	**83.66**	**84.67**	**48.00**	**74.52**	**74.26**	**58.40**	**75.18**	**75.71**	**52.80**	**74.26**	**72.04**	**37.60**
TransCoder	78.04	77.71	26.40	65.00	63.76	19.20	74.83	78.02	38.40	68.86	66.27	33.60	-	-	-	-	-	-
TransCoder-ST	**84.42**	**84.15**	**69.60**	**70.84**	**69.14**	38.40	76.20	79.26	40.80	68.45	67.69	36.80	-	-	-	-	-	-
Type 4																		
CodeBERT	35.89	37.10	0.00	26.32	30.22	0.00	25.18	33.53	0.00	20.38	28.05	0.00	34.05	36.48	0.00	22.88	27.93	0.00
CodeT5	37.08	39.82	0.00	21.79	28.13	0.00	**31.76**	**40.87**	0.00	**33.80**	45.19	0.00	**43.20**	**44.94**	**8.00**	**29.71**	**36.61**	0.00
TransCoder	37.71	39.35	0.00	**25.99**	**34.39**	0.00	19.91	30.09	0.00	31.75	**48.37**	0.00	-	-	-	-	-	-
TransCoder-ST	**50.99**	**49.50**	**4.00**	24.36	29.51	0.00	24.96	34.78	**4.00**	33.38	43.06	0.00	-	-	-	-	-	-

1) Effect of taxonomy

Comparison of model performance on the benchmark

Model	Java→Python			Python→Java			Java→C++			C++→Java			Java→JavaScript			JavaScript→Java		
	BLEU	CB	CA	BLEU	CB	CA	BLEU	CB	CA	BLEU	CB	CA	BLEU	CB	CA	BLEU	CB	CA
Type 1																		
CodeBERT	81.37	82.67	78.40	81.26	84.14	67.20	93.83	94.18	83.20	95.68	95.54	84.00	83.77	84.91	78.40	**93.47**	93.52	72.80
CodeT5	82.71	83.07	**88.00**	81.98	84.81	78.40	**94.14**	**94.41**	90.40	**97.39**	**97.35**	94.40	**84.67**	**85.25**	**85.60**	93.46	**93.81**	**76.80**
TransCoder	86.28	83.73	57.60	82.82	85.32	78.40	89.73	90.61	94.40	93.55	94.22	92.80	-	-	-	-	-	-
TransCoder-ST	**90.12**	**88.66**	80.80	**90.86**	**91.94**	**88.00**	87.95	88.78	**97.60**	94.50	95.15	**95.20**	-	-	-	-	-	-
Type 2																		
CodeBERT	77.52	77.70	53.60	**69.75**	68.04	33.60	89.19	89.17	62.40	77.83	76.30	55.20	80.22	79.57	55.20	77.02	74.01	42.40
CodeT5	78.90	79.11	74.40	69.57	**68.91**	50.40	**91.33**	**91.47**	**72.80**	**82.15**	**81.28**	**83.20**	**82.74**	**82.06**	**73.60**	**80.35**	**78.07**	**62.40**
TransCoder	84.39	84.37	60.80	65.13	65.18	46.40	83.76	84.95	69.60	69.50	68.77	57.60	-	-	-	-	-	-
TransCoder-ST	**87.38**	**87.60**	**76.00**	66.11	64.82	**51.20**	83.93	84.85	71.20	70.11	69.55	55.20	-	-	-	-	-	-
Type 3																		
CodeBERT	74.51	74.38	28.80	63.69	62.96	16.80	79.14	80.64	31.20	71.81	69.49	26.40	72.56	72.13	25.60	72.03	68.56	20.00

Finding 1:

- ✓ G-TransEval with the taxonomy is effective in differentiating between various levels of translations.
- ✓ As the translation level increases, the task becomes more rigorous.
- ✓ Unsupervised approaches exhibit better performance on lower levels, while supervised approaches demonstrate better performance on higher levels.

2) Effect of programming languages

Comparison of model performance on the benchmark

Model	Java→Python			Python→Java			Java→C++			C++→Java			Java→JavaScript			JavaScript→Java		
	BLEU	CB	CA	BLEU	CB	CA	BLEU	CB	CA	BLEU	CB	CA	BLEU	CB	CA	BLEU	CB	CA
Type 1																		
CodeBERT	81.37	82.67	78.40	81.26	84.14	67.20	93.83	94.18	83.20	95.68	95.54	84.00	83.77	84.91	78.40	**93.47**	93.52	72.80
CodeT5	82.71	83.07	**88.00**	81.98	84.81	78.40	**94.14**	**94.41**	90.40	**97.39**	**97.35**	94.40	**84.67**	**85.25**	**85.60**	93.46	**93.81**	**76.80**
TransCoder	86.28	83.73	57.60	82.82	85.32	78.40	89.73	90.61	94.40	93.55	94.22	92.80	-	-	-	-	-	-
TransCoder-ST	**90.12**	**88.66**	80.80	**90.86**	**91.94**	**88.00**	87.95	88.78	**97.60**	94.50	95.15	**95.20**	-	-	-	-	-	-
Type 2																		
CodeBERT	77.52	77.70	53.60	**69.75**	68.04	33.60	89.19	89.17	62.40	77.83	76.30	55.20	80.22	79.57	55.20	77.02	74.01	42.40
CodeT5	78.90	79.11	74.40	69.57	**68.91**	50.40	**91.33**	**91.47**	**72.80**	**82.15**	**81.28**	**83.20**	**82.74**	**82.06**	**73.60**	**80.35**	**78.07**	**62.40**
TransCoder	84.39	84.37	60.80	65.13	65.18	46.40	83.76	84.95	69.60	69.50	68.77	57.60	-	-	-	-	-	-
TransCoder-ST	**87.38**	**87.60**	**76.00**	66.11	64.82	**51.20**	83.93	84.85	71.20	70.11	69.55	55.20	-	-	-	-	-	-
Type 3																		
CodeBERT	74.51	74.38	28.80	63.69	62.96	16.80	79.14	80.64	31.20	71.81	69.49	26.40	72.56	72.13	25.60	72.03	68.56	20.00
CodeT5	78.62	78.95	68.00	67.47	67.61	**44.80**	**83.66**	**84.67**	**48.00**	**74.52**	**74.26**	**58.40**	**75.18**	**75.71**	**52.80**	**74.26**	**72.04**	**37.60**
TransCoder	78.04	77.71	26.40	65.00	63.76	19.20	74.83	78.02	38.40	68.86	66.27	33.60	-	-	-	-	-	-
TransCoder-ST	**84.42**	**84.15**	**69.60**	**70.84**	**69.14**	38.40	76.20	79.26	40.80	68.45	67.69	36.80	-	-	-	-	-	-
Type 4																		
CodeBERT	35.89	37.10	0.00	26.32	30.22	0.00	25.18	33.53	0.00	20.38	28.05	0.00	34.05	36.48	0.00	22.88	27.93	0.00
CodeT5	37.08	39.82	0.00	21.79	28.13	0.00	**31.76**	**40.87**	0.00	**33.80**	45.19	0.00	**43.20**	**44.94**	**8.00**	**29.71**	**36.61**	0.00
TransCoder	37.71	39.35	0.00	**25.99**	**34.39**	0.00	19.91	30.09	0.00	31.75	**48.37**	0.00	-	-	-	-	-	-
TransCoder-ST	**50.99**	**49.50**	**4.00**	24.36	29.51	0.00	24.96	34.78	**4.00**	33.38	43.06	0.00	-	-	-	-	-	-

2) Effect of programming languages

Comparison of model performance on the benchmark

Model	Java→Python			Python→Java			Java→C++			C++→Java			Java→JavaScript			JavaScript→Java		
	BLEU	CB	CA	BLEU	CB	CA	BLEU	CB	CA	BLEU	CB	CA	BLEU	CB	CA	BLEU	CB	CA
Type 1																		
CodeBERT	81.37	82.67	78.40	81.26	84.14	67.20	93.83	94.18	83.20	95.68	95.54	84.00	83.77	84.91	78.40	**93.47**	93.52	72.80
CodeT5	82.71	83.07	**88.00**	81.98	84.81	78.40	**94.14**	**94.41**	90.40	**97.39**	**97.35**	94.40	**84.67**	**85.25**	**85.60**	93.46	**93.81**	**76.80**
TransCoder	86.28	83.73	57.60	82.82	85.32	78.40	89.73	90.61	94.40	93.55	94.22	92.80	-	-	-	-	-	-
TransCoder-ST	**90.12**	**88.66**	80.80	**90.86**	**91.94**	**88.00**	87.95	88.78	**97.60**	94.50	95.15	**95.20**	-	-	-	-	-	-
Type 2																		
CodeBERT	77.52	77.70	53.60	**69.75**	68.04	33.60	89.19	89.17	62.40	77.83	76.30	55.20	80.22	79.57	55.20	77.02	74.01	42.40
CodeT5	78.90	79.11	74.40	69.57	**68.91**	50.40	**91.33**	**91.47**	**72.80**	**82.15**	**81.28**	**83.20**	**82.74**	**82.06**	**73.60**	**80.35**	**78.07**	**62.40**
TransCoder	84.39	84.37	60.80	65.13	65.18	46.40	83.76	84.95	69.60	69.50	68.77	57.60	-	-	-	-	-	-
TransCoder-ST	**87.38**	**87.60**	**76.00**	66.11	64.82	**51.20**	83.93	84.85	71.20	70.11	69.55	55.20	-	-	-	-	-	-
Type 3																		
CodeBERT	74.51	74.38	28.80	63.69	62.96	16.80	79.14	80.64	31.20	71.81	69.49	26.40	72.56	72.13	25.60	72.03	68.56	20.00
CodeT5	78.62	78.95	68.00	67.47	67.61	**44.80**	**83.66**	**84.67**	**48.00**	**74.52**	**74.26**	**58.40**	**75.18**	**75.71**	**52.80**	**74.26**	**72.04**	**37.60**
TransCoder	78.04	77.71	26.40	65.00	63.76	19.20	74.83	78.02	38.40	68.86	66.27	33.60	-	-	-	-	-	-

Finding 2:

- ✓ Translations between syntactically dissimilar languages yield lower CA scores for type-1 translations.
- ✓ Translations from dynamically- to statically-typed languages are more challenging than other language pairs.

3) Results of LLMs

Model	Java→Python			Python→Java			Java→C++			C++→Java			Java→JavaScript			JavaScript→Java		
	BLEU	CB	CA	BLEU	CB	CA	BLEU	CB	CA	BLEU	CB	CA	BLEU	CB	CA	BLEU	CB	CA
Type 1																		
gpt-3.5-turbo	55.27	69.39	**96.00**	86.65	89.69	**88.80**	90.04	91.53	**99.20**	89.93	92.66	**98.40**	**83.67**	**83.23**	**96.00**	**92.36**	**93.60**	**91.20**
StarCoderBase	**61.11**	**73.84**	85.60	**92.96**	**93.02**	84.80	**94.64**	**94.49**	96.00	**93.76**	**95.40**	91.20	81.66	81.74	78.40	92.05	92.30	64.80
Type 2																		
gpt-3.5-turbo	54.78	67.08	**93.60**	74.72	72.20	**84.80**	83.10	83.72	87.20	84.06	82.68	**88.80**	**85.86**	**84.31**	**91.20**	76.41	73.59	**80.80**
StarCoderBase	**61.48**	**72.61**	83.20	**87.13**	**84.62**	79.20	**91.96**	**91.03**	**93.60**	**90.35**	**88.75**	79.20	84.51	83.32	88.00	**88.54**	**85.53**	63.20
Type 3																		
gpt-3.5-turbo	53.76	65.10	**94.40**	74.48	74.00	**85.60**	80.77	80.28	**81.60**	83.48	80.18	**91.20**	**81.53**	**81.25**	**91.20**	73.91	73.07	**87.20**
StarCoderBase	**62.51**	**74.21**	88.80	**85.08**	**82.97**	73.60	**87.71**	**85.83**	78.40	**88.29**	**86.41**	84.00	81.09	80.32	80.00	**86.07**	**84.49**	60.80
Type 4																		
gpt-3.5-turbo	27.59	40.42	**72.00**	37.60	47.81	64.00	35.57	43.08	**68.00**	44.73	55.14	**68.00**	**61.67**	**61.99**	**76.00**	35.62	47.78	**88.00**
StarCoderBase	**41.14**	**47.91**	44.00	**54.09**	**59.55**	**72.00**	**40.61**	**44.95**	48.00	**60.73**	**64.25**	**68.00**	49.29	47.52	56.00	**55.72**	**62.60**	64.00

Finding 3:

✓ LLMs alleviate the knowledge gap of higher level translations through the substantial number of parameters and training data, hence yielding competitive results in type-2 and type-3 translations.

Conclusion

- Empirical Study
- Taxonomy on code translation
- G-TransEval: a benchmark of code translation
- Experiments on G-TransEval

Benchmark and code released: https://github.com/PolyEval/G-TransEval

Thank You!

Q&A