Transforming Generic Coder LLMs to Effective Binary Code Embedding Models for Similarity Detection

NeurIPS 2025

Litao Li*, Leo Song*, Steven H. H. Ding, Benjamin C. M. Fung, Philippe Charland

Queen's University, McGill University, Defence R&D Canada

*Equal contribution.

Background

- Binary Code Similarity Detection (BCSD) is crucial for:
  - Malware detection
  - Vulnerability identification and discovery
- Challenges:
  - Sparse, low-level, and architecture-dependent syntax
  - Compilation diversity (optimizations, architectures, obfuscation)
- Limitations of existing deep learning-based BCSD
  - Lack of generalizability: only a subset of compiler setting
  - Does not utilize pretrained knowledge of frontier models

Contributions – EBM (Effective Binary Matching)

- Proposed a multi-stage training framework to tackle generalizability and effectiveness of binary code matching
- Designed data augmentation processes and training objectives addressing specifically for the diverse compile options of binary code
- Significantly improves performance over the baselines with thorough experiments and ablation study

EBM Framework

- Four stages: data augmentation, translation-style causal training, LLM2Vec embedding, contrastive learning via cGTE loss
- Data Augmentation
  - De-noising and enabling structural and language awareness
- Causal translation training
  - Enhances generalization across architectures by training autoregressive model on concatenated function pairs
- LLM2Vec
  - Representation learning using masked next token prediction for semantic embedding
- Cumulative GTE Loss
  - Generalizes InfoNCE by all available in-batch contrasts for optimal resource usage

Evaluation – Cross-Optimization

Models	MRR						Recall@1					
	O0,O3	O0,O1	O0,O2	O1,O3	O2,O3	Avg.	O0,O3	O0,O1	O0,O2	O1,O3	O2,O3	Avg.
SAFE	0.189	0.189	0.200	0.218	0.171	0.193	0.063	0.000	0.063	0.063	0.000	0.038
PalmTree	0.023	0.020	0.019	0.314	0.878	0.251	0.008	0.006	0.007	0.184	0.676	0.176
Asm2Vec	0.444	0.494	0.460	0.535	0.563	0.499	0.234	0.290	0.252	0.343	0.376	0.299
OrderMatters	0.006	0.006	0.008	0.006	0.006	0.006	0.000	0.001	0.002	0.001	0.000	0.001
GraphCodeBERT (125M)	0.636	0.757	0.673	0.792	0.920	0.756	0.560	0.694	0.602	0.722	0.895	0.695
CodeT5+ (110M)	0.604	0.650	0.629	0.830	0.893	0.721	0.532	0.572	0.552	0.783	0.869	0.662
Qwen2.5-Emb (1.5B)	0.569	0.648	0.573	0.773	0.907	0.694	0.498	0.578	0.505	0.699	0.875	0.631
Qwen2.5-Coder (1.5B)	0.758	0.881	0.807	0.864	0.936	0.849	0.706	0.842	0.757	0.810	0.912	0.805
EBM (0.5B)	**0.850**	**0.942**	**0.902**	**0.933**	**0.955**	**0.916**	**0.793**	**0.903**	**0.850**	**0.887**	**0.929**	**0.872**

Table 1: Evaluation on cross-optimization settings (O0, O1, O2, and O3) with a pool size of 1,000.

Evaluation – Cross-Architecture

Models	MRR				Recall@1			
	Arm, x64	PowerPC, x64	MIPS, x64	Avg.	Arm, x64	PowerPC, x64	MIPS, x64	Avg.
SAFE	0.239	0.187	0.196	0.208	0.063	0.063	0.063	0.063
PalmTree	0.037	0.036	0.018	0.031	0.031	0.013	0.007	0.017
Asm2Vec	0.242	0.293	0.417	0.317	0.085	0.113	0.231	0.143
OrderMatters	0.007	0.007	0.007	0.007	0.002	0.000	0.001	0.001
GraphCodeBERT (125M)	0.067	0.269	0.495	0.277	0.037	0.204	0.419	0.220
CodeT5+ (110M)	0.056	0.303	0.462	0.274	0.035	0.227	0.392	0.218
Qwen2.5-Emb (1.5B)	0.039	0.059	0.409	0.169	0.031	0.035	0.331	0.132
Qwen2.5-Coder (1.5B)	0.256	0.481	0.548	0.428	0.179	0.380	0.442	0.334
EBM (0.5B)	**0.783**	**0.792**	**0.859**	**0.811**	**0.675**	**0.703**	**0.784**	**0.721**

Table 2: Evaluation on cross-architecture settings (Arm, x86-64, PowerPC, and MIPS) with a pool size of 1,000.

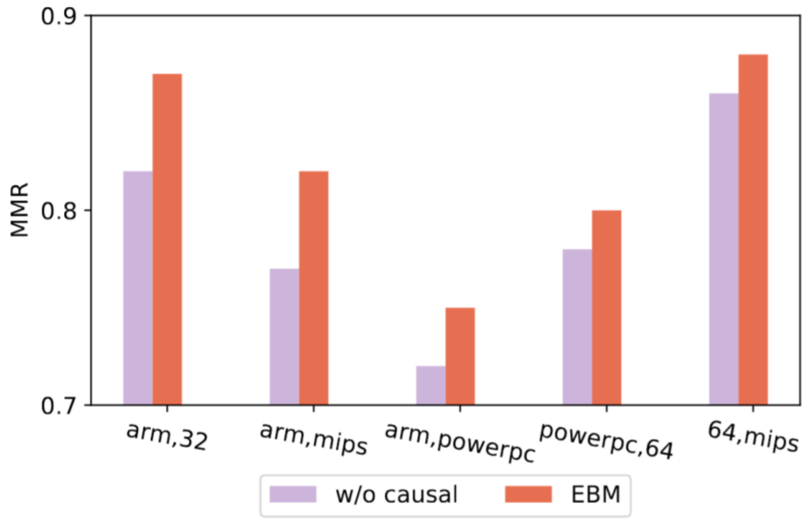
Evaluation – Cross-Obfuscation

Models	**MRR**				**Recall@1**			
	all, none	none, bcf	sub, fla	Avg.	all, none	none, bcf	sub, fla	Avg.
SAFE	0.256	0.181	0.264	0.234	0.0625	0.0625	0.125	0.083
PalmTree	0.122	0.289	0.215	0.209	0.060	0.200	0.083	0.114
Asm2Vec	0.200	0.181	0.264	0.215	0.069	0.063	0.125	0.086
OrderMatters	0.008	0.006	0.007	0.007	0.001	0.001	0.001	0.001
GraphCodeBERT (125M)	0.230	0.648	0.479	0.452	0.163	0.557	0.391	0.370
CodeT5+ (110M)	0.176	0.619	0.372	0.389	0.118	0.539	0.291	0.316
Qwen2.5-Emb (1.5B)	0.288	0.630	0.466	0.461	0.213	0.538	0.375	0.375
Qwen2.5-Coder (1.5B)	0.391	0.719	0.580	0.563	0.301	0.637	0.491	0.476
EBM (0.5B)	**0.531**	**0.815**	**0.784**	**0.710**	**0.454**	**0.738**	**0.713**	**0.635**

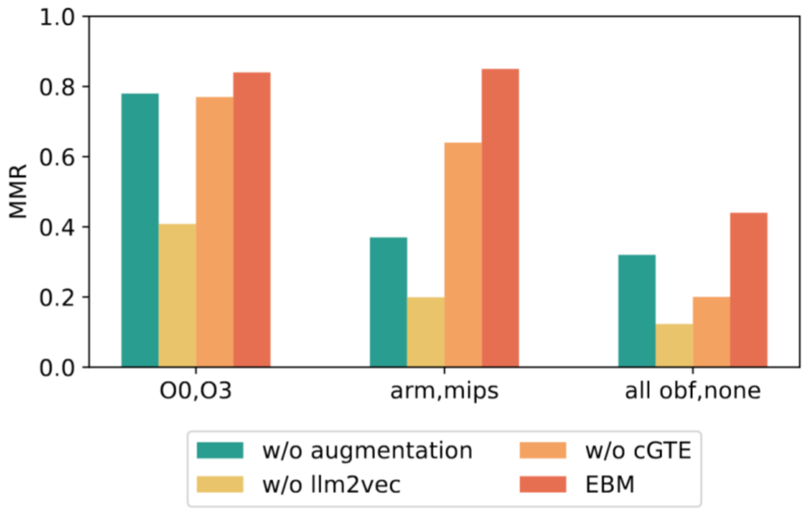
Table 3: Evaluation on cross-obfuscation settings (all obfuscations, bogus control flow, flattened, and substitution) with a pool size of 1,000. The proposed EBM model outperforms all baselines by an absolute margin of over 15% in both MRR and Recall@1 metrics.

Ablation Study

- **Data Augmentation**: Key for handling obfuscation
- **Causal Training**: Crucial for cross-arch tasks
- **LLM2Vec**: 2x+ MRR boost
- **cGTE**: Adds rich contrastive signals under low resources



(a) Causal training ablation study



(b) Ablation study for data augmentation, LLM2Vec, and cGTE.

Conclusions

- EBM effectively transforms generic LLMs into BCSD experts
- Small models, high performance
- Scalable, open-source, no reliance on closed APIs
- Paves the way for secure, effective, and practical binary analysis