

Performing Joint Analytics Requires Data Alignment

cust_name	city	zip_code
K. Ito	NYC	10012
A. Smith	BOS	02215

A data cleaning pipeline:
imputation & schema matching

custname	cty	zipcode
J. Doe	NULL	02446

A data cleaning pipeline:
imputation & schema matching

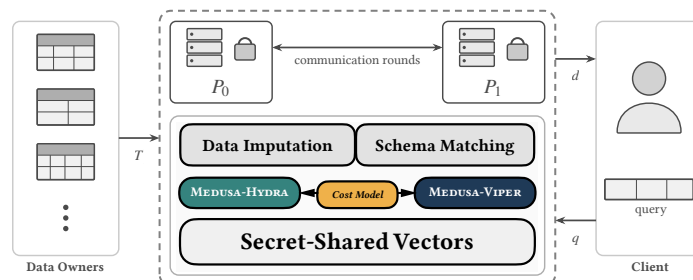
cust_name	city	zip_code
K. Ito	NYC	10012
A. Smith	BOS	02215
J. Doe	BOS	02446

Attribute to compare
Attribute to impute

Analytics systems based on secure multiparty computation (MPC) like CryptDough (SOSP'26), ORQ (SOSP'25), CoVault (USENIX Security'25), Piranha (USENIX Security'22) all assume aligned inputs.

Existing private edit distance systems make one of three concessions: they either approximate, only compute one-to-one edit distances, or rely on input assumptions primarily meant for genomic sequencing applications; the state-of-the-art system SecurED (PETS'25) assumes the distance is a small fraction of the input length.

MEDUSA makes none of these concessions and still outperforms SecurED.



Each server holds a random piece of the data; the pieces XOR to the actual value.

$0101 = 1100 \oplus 1001$ P_0 holds 1100, P_1 holds 1001

Boolean Sharing $\rightarrow$ MEDUSA-HYDRA

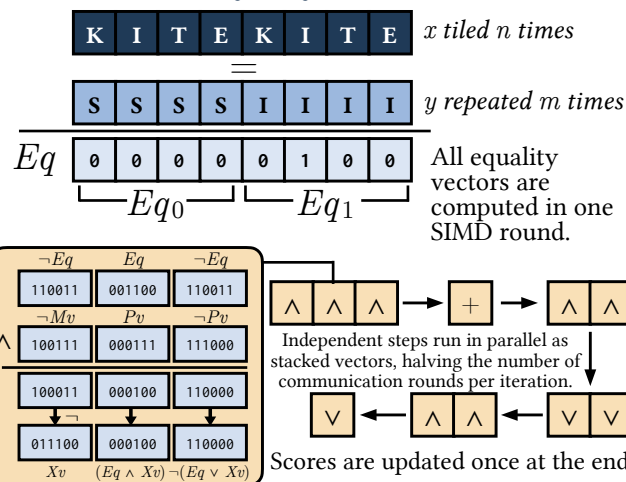
$P_0 \leftrightarrow P_1$ Many rounds, less data each

Yao Sharing $\rightarrow$ MEDUSA-VIPER

$P_0 \rightarrow P_1$ One round, more data each

MEDUSA Makes Private Edit Distance Computation Fast

MPC-Friendly Myers



We choose the bit-parallel Myers algorithm (JACM'99) as a first step to minimizing communication between parties. This raises several challenges that we solve to develop MEDUSA.

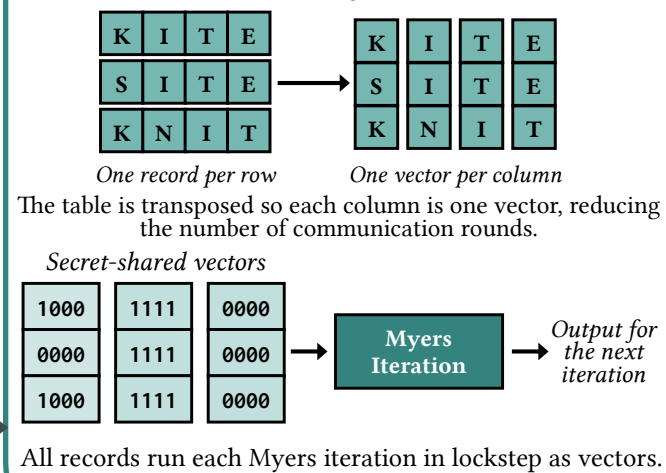
Making the Myers algorithm MPC-friendly

Amortizing costs across all records

Switching between sharing modes

Integrating with secure analytics engines

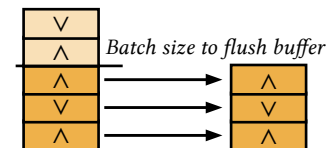
Record-Parallel Myers (in MEDUSA-HYDRA)



Engine Integration

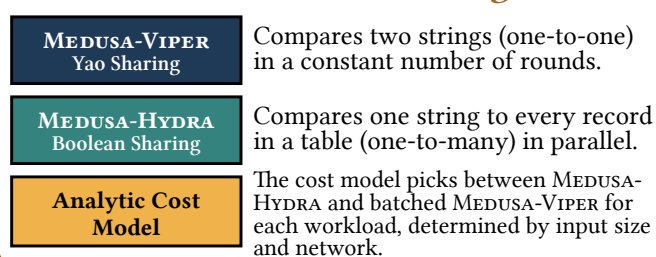
CryptDough (SOSP'26) has no garbled-circuit support, so MEDUSA integrates a vectorized garbling pipeline on it.

Every Boolean operation is processed in a buffer that flushes at set batch sizes.



We split sets of secure Boolean operations to send across several TCP connections, and the parties are pipelined to compute in parallel.

Cost-Based Mode Switching

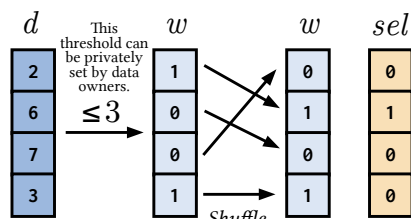


End-to-End Private Data Imputation and Schema Matching

Private Data Imputation

Goal: Select a uniformly random value under some private threshold.

Insight: Obviously select it by picking the first (using prefix-ANDs) with a shuffle.



elig	0	1	0	1	0
-PrefixOR	1	1	0	0	0
sel	0	1	0	0	0

We build a novel oblivious selection circuit to perform end-to-end private data imputation.

Up to **3.0x** faster than SecurED (PETS'25), the state-of-the-art private edit distance system, and faster on all input sizes.

Up to **457x** faster stable matching for schema matching than the state-of-the-art, and faster on all input sizes.

9.7x lower amortized cost for imputation.

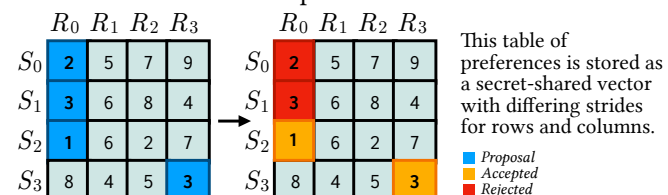
Under **2x** difference in all incorrect cost model selections.

No prior work supports end-to-end private data imputation or schema matching based on edit distance.

Private Schema Matching

Goal: We build off prior work on MPC stable matching to create a protocol to stable match edit distances.

Insight: Since the preferences are edit distances, they will be symmetric, so we can just run two oblivious argmin operations.



Our approach to stable matching edit distances outperforms the state-of-the-art private stable matching protocol (Mondal et al., arXiv'23).