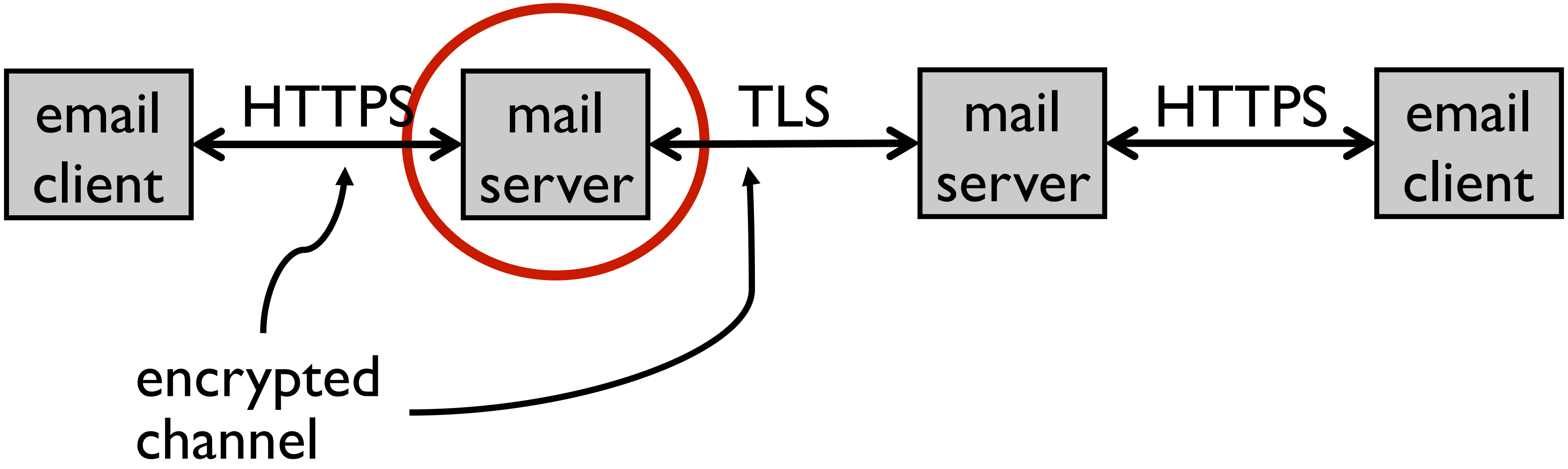
Email encryption is compatible with provider-supplied functions

Trinabh Gupta*†, Henrique Fingler*,
Lorenzo Alvisi*¶, and Michael Walfish†

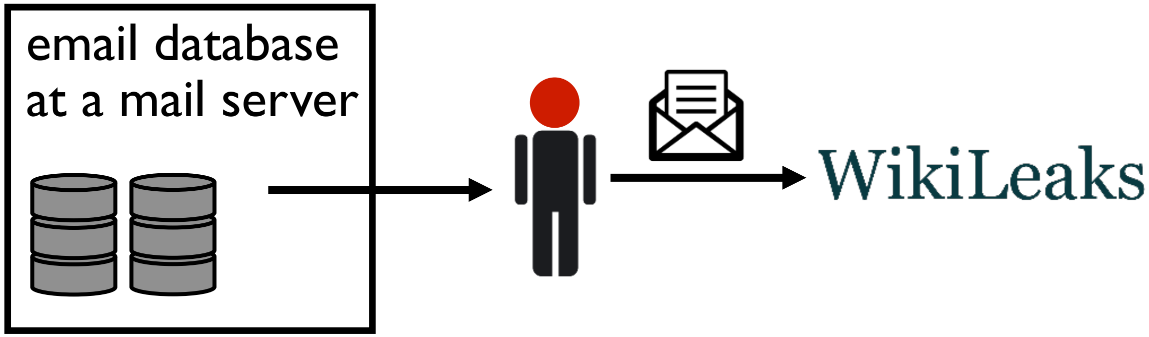
*The University of Texas at Austin
†New York University
¶Cornell



If a mail server can access email, then …

… rogue employees can access email.

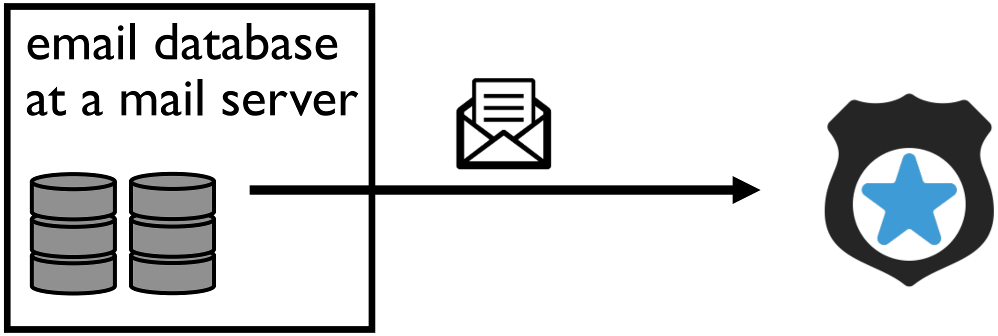
… hackers can access email.



email database at a mail server

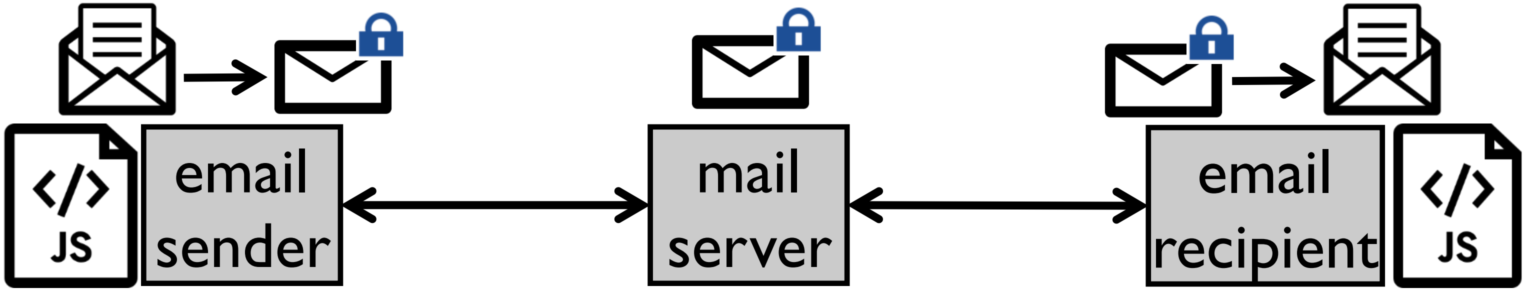
WikiLeaks

… law enforcement agencies can access email.



email database at a mail server

End-to-end encryption can prevent email leaks



WhatsApp and iMessage use end-to-end encryption.

So, why don't email service providers deploy end-to-end email encryption?

End-to-end encryption is in conflict with service providers' functions

"… we couldn't run our system if everything in it were encrypted because then we wouldn't know which ads to show you."

"So this is a system that was designed around a particular business model."

[Vint Cerf. Sixth Annual Meeting of the Internet Governance Forum. 2011]

We asked: can we build an email system that

a) supports end-to-end email encryption,

b) supports provider-supplied functions consistent with existing commercial regime, and

c) has low costs?

Pretzel demonstrates:

Email encryption is compatible with provider-supplied functions.

Pretzel requirements:

end-to-end email encryption

+

basic functions: spam filtering, topic extraction

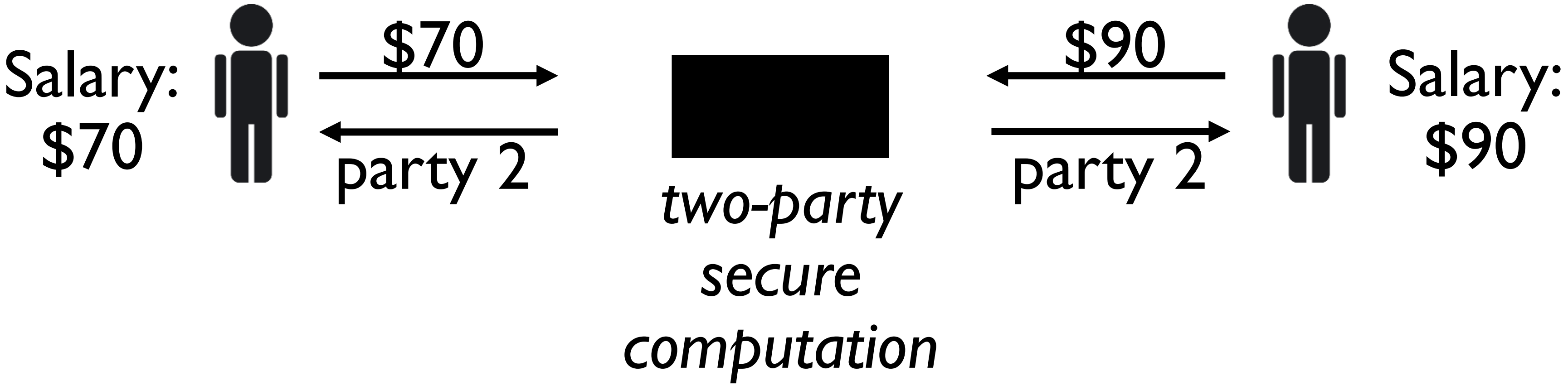
+

low resource cost

“[we cannot have end-to-end encryption and AI] until someone figures out how to do homomorphic machine learning.”

[Thai Duong, an engineer who co-leads Google’s product security team. 2011]

Two-party secure computation (2PC) from 10,000 feet



Salary: $70

$70

party 2

two-party secure computation

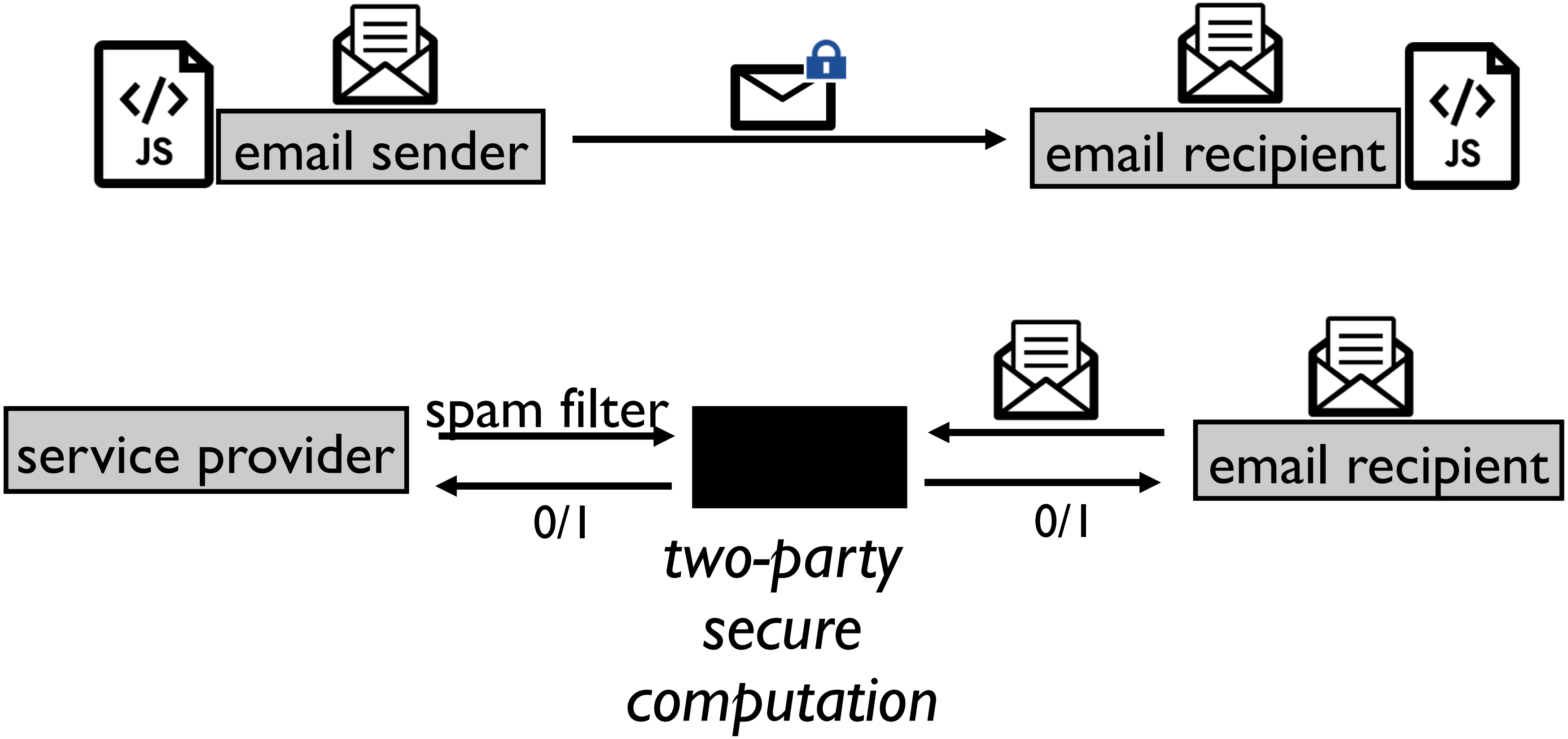
$90

party 2

Salary: $90

- can handle arbitrary computations

Two-party secure computation (2PC) crypto protocols can enable encryption and functions



but have huge resource (CPU, network, etc.) costs.

Pretzel:

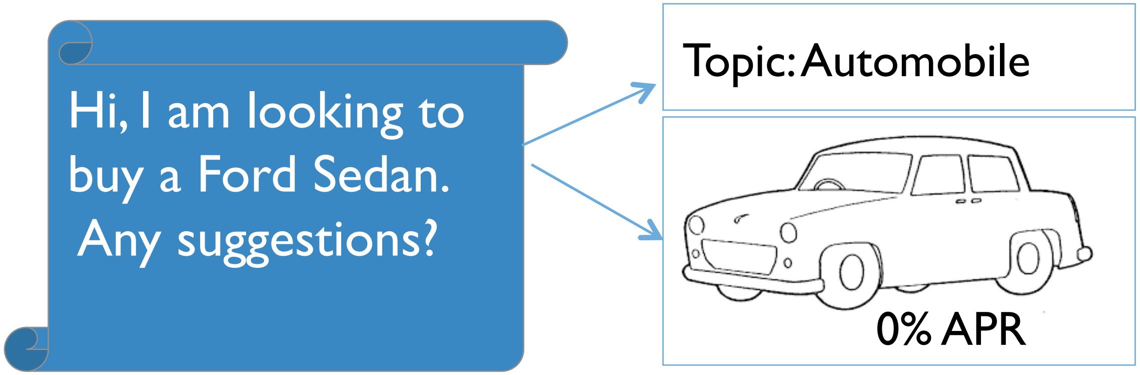
reduces costs of 2PC by up to 100x,
by refining 2PC for specific functions.

Rest of this talk

- Two example functions.
- Background on 2PC (Yao+GLLM) that can implement these functions.
- Refinement of 2PC.

Pretzel supports two functions: spam filtering and topic extraction.

Topic extraction:



Hi, I am looking to buy a Ford Sedan. Any suggestions?

Topic: Automobile

0% APR

Linear classifiers (for both spam filtering, topic extraction)

categories

words in dictionary

	networks		OS		security
BGP	0.4		0.0		0.1
route	0.3	…	0.1	…	0.0
cloud	0.1		0.7		0.3
encrypt	0.2		0.2		0.6

model

BGP may be
used for routing.

words in email:
{BGP, routing}

Part 1: Add probabilities corresponding to words in email.

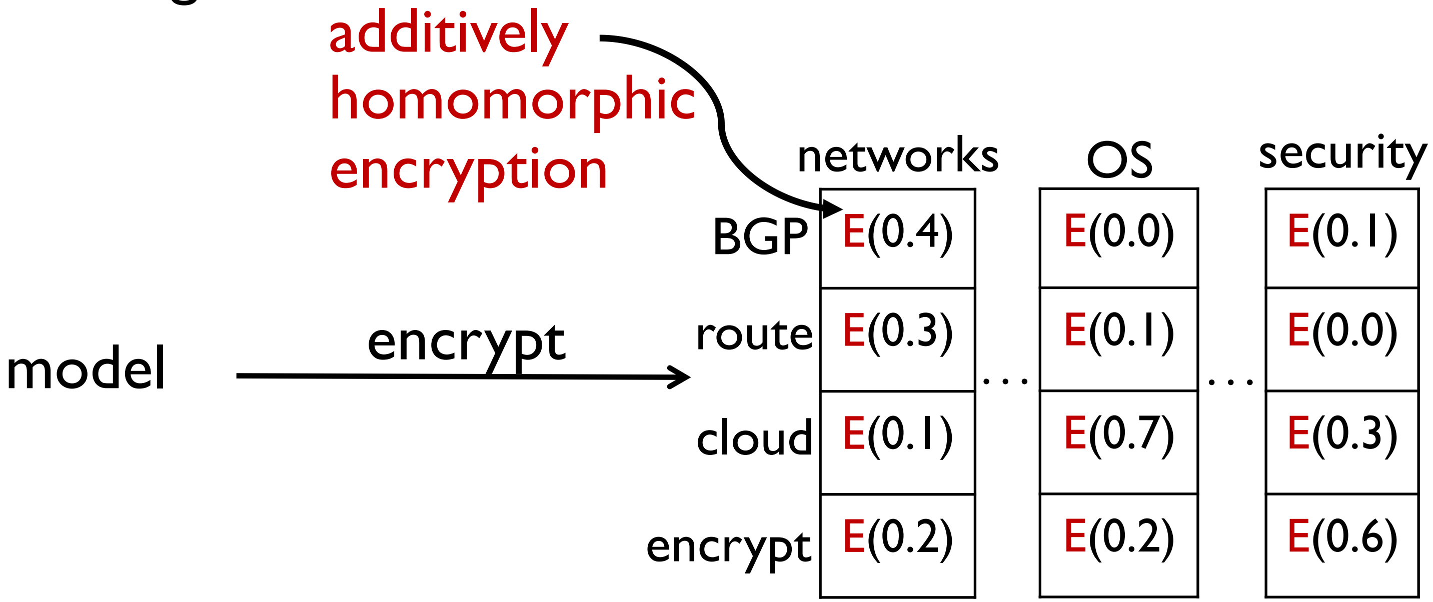
Example: networks: 0.7

Part 2: Compare outputs from part 1.

Category is “networks”.

Background on Yao+GLLM 2PC

Provider does
the following:



additively
homomorphic
encryption

model

encrypt

	networks		OS		security
BGP	E(0.4)		E(0.0)		E(0.1)
route	E(0.3)	…	E(0.1)	…	E(0.0)
cloud	E(0.1)		E(0.7)		E(0.3)
encrypt	E(0.2)		E(0.2)		E(0.6)

Background on Yao+GLLM 2PC

Client does the following:

	networks		OS		security
BGP	E(0.4)		E(0.0)		E(0.1)
route	E(0.3)	. . .	E(0.1)	. . .	E(0.0)
cloud	E(0.1)		E(0.7)		E(0.3)
encrypt	E(0.2)		E(0.2)		E(0.6)

encrypted model

BGP may be used for routing.

words in email:
{BGP, routing}

Add encrypted probabilities using additive homomorphism.

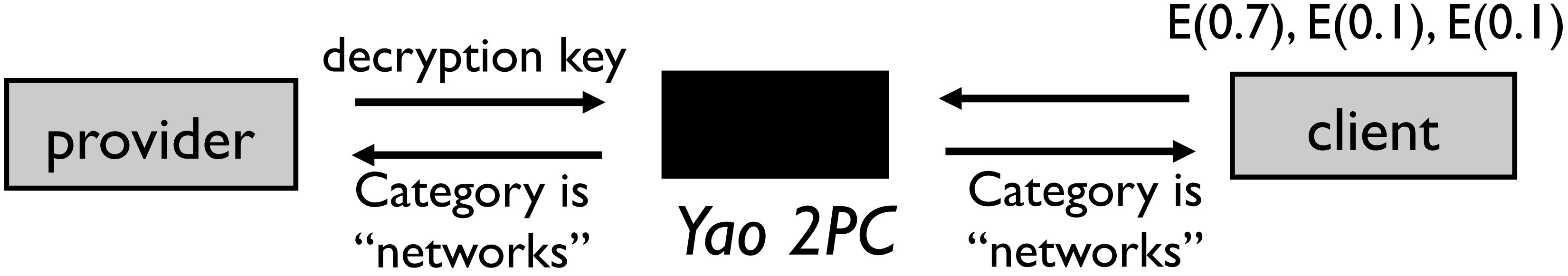
Example:

some operation

networks: $E(0.4) \circ E(0.3) = E(0.4 + 0.3) = E(0.7)$

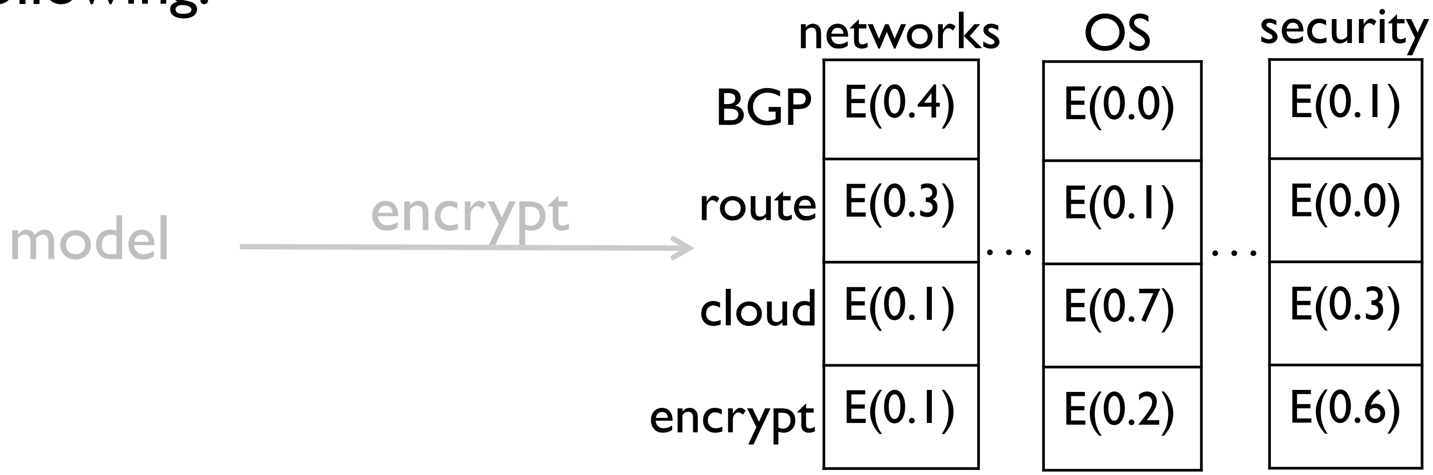
Background on Yao+GLLM 2PC

Client and provider do the following:



Cost issues in Yao+GLLM 2PC

Provider does
the following:



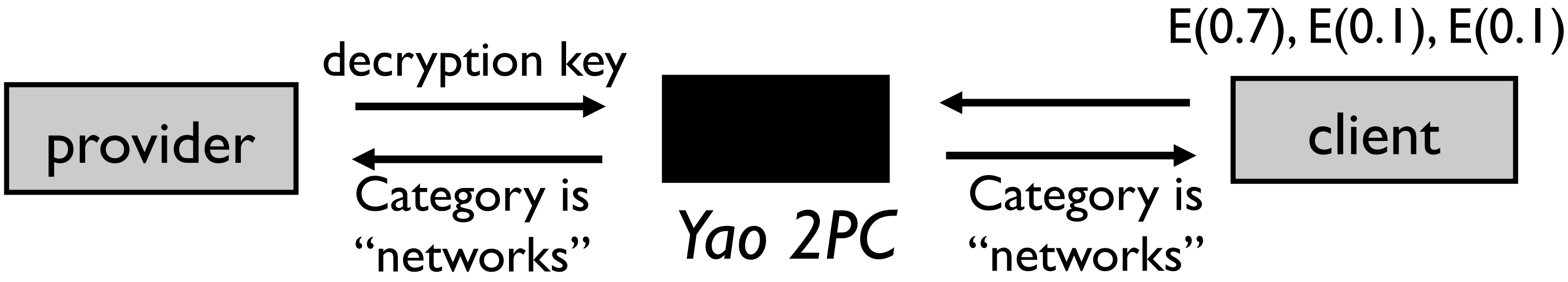
	networks	OS	security
BGP	E(0.4)	E(0.0)	E(0.1)
route	E(0.3)	E(0.1)	E(0.0)
cloud	E(0.1)	E(0.7)	E(0.3)
encrypt	E(0.1)	E(0.2)	E(0.6)

Issue 1: encrypted
model is large

Provider sends encrypted
model to the client.

Cost issues in Yao+GLLM 2PC

Client and provider do the following:


provider

decryption key

Category is "networks"

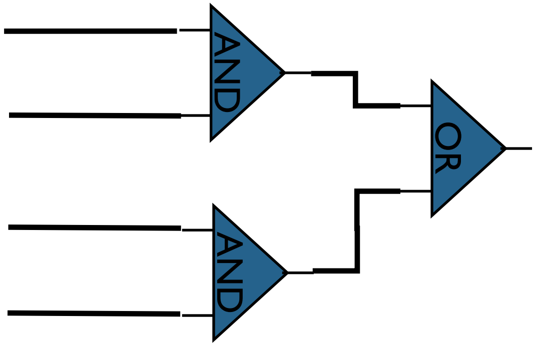
Yao 2PC

E(0.7), E(0.1), E(0.1)

Category is "networks"

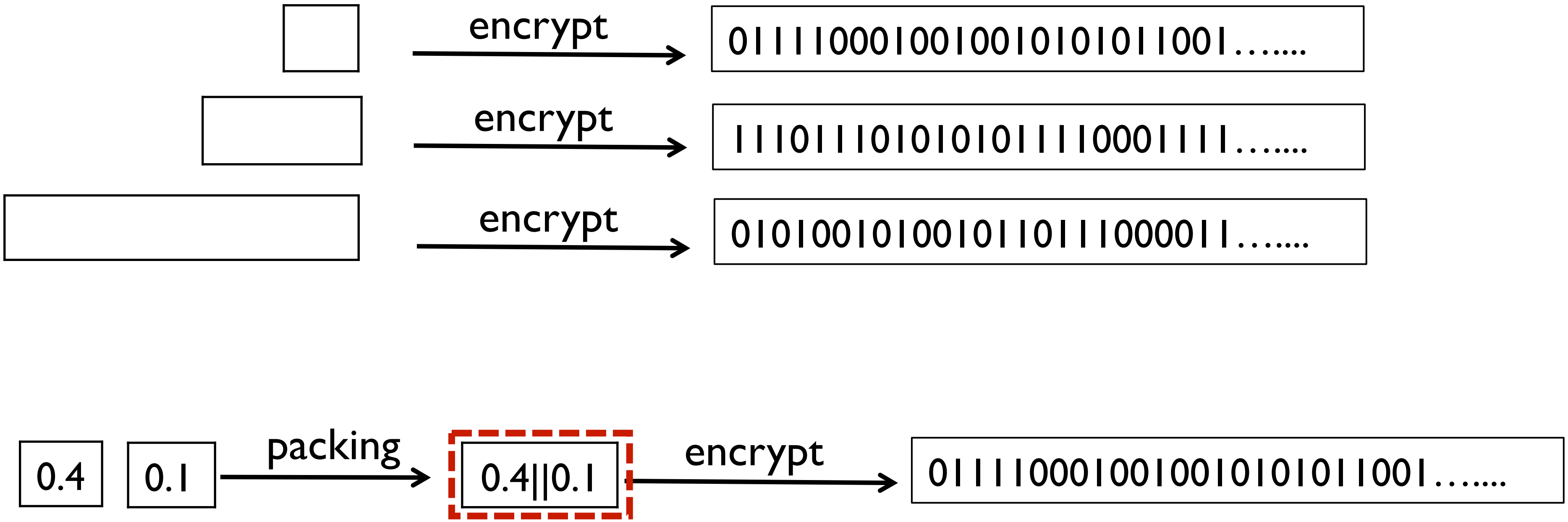
client

Issue 2: CPU and network costs of Yao part grow with the number of categories.



Issues in Yao+GLLM	Pretzel's refinements
encrypted model is large	adapt packing from other domains
CPU and network costs of Yao part grow with the number of categories	decomposed classification

Pretzel uses packing to reduce client-side storage cost



- Packing can reduce the size of model by #elements packed
- Caution: Must preserve addition operation in cipherspace

Issues in Yao+GLLM	Pretzel’s refinements
encrypted model is large	adapt packing from other domains
CPU and network costs of Yao part grow with the number of categories	decomposed classification

Pretzel's decomposed classification at a high level

What we want:

{network, OS, security, algo} → {network}

set of all topics → chosen topic

step 1: *performed at client using a public classifier*

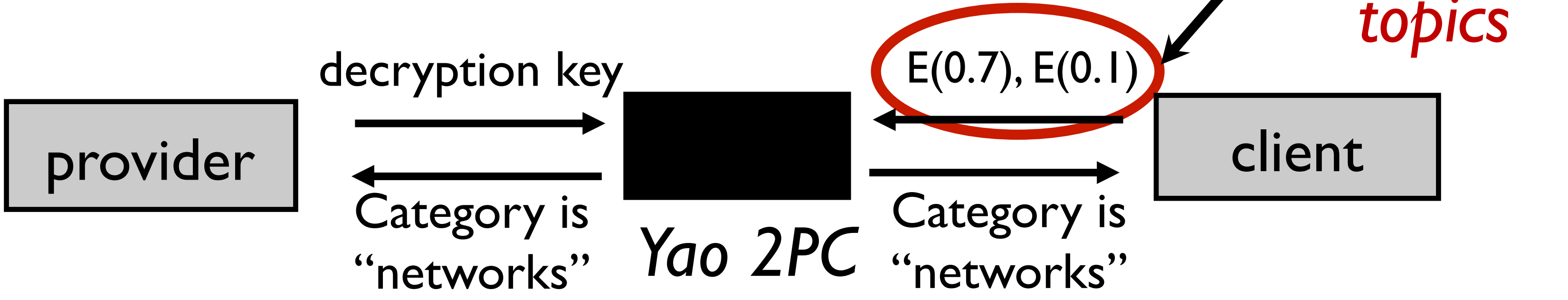
{network, OS, security, algo} → {network, algo}

set of all topics → candidate topics

step 2: *performed using 2PC*

{network, algo} → {network}

candidate topics → chosen topic



only for candidate topics

Outline

✓Background on 2PC (Yao+GLLM).

✓Design of Pretzel.

- Evaluation of Pretzel

Experiment method

Baselines:

- Non-private system
- Yao+GLLM (with Paillier cryptosystem and GLLM packing)

Functions:

- Spam filtering (5M features)
- Topic extraction (20K features, 2048 topics, 20 candidate topics)

Measure CPU time, network transfers, and storage space

Overheads for spam filtering (relative to status quo)

	Yao+GLLM	Pretzel
provider-side CPU time:	15.9x	2.7x
network transfers:	1.05x	1.26x
client-side storage:	1.3GB	183MB

Overheads for topic extraction (relative to status quo)

	Yao+GLLM	Pretzel
provider-side CPU time:	110x	1.8x
network transfers:	109x	5.4x
client-side storage:	288MB	720MB

Related work

- Improving performance of general purpose 2PC [SEC11, CCS12, NDSS12, S&P12, SEC12, S&P14, EUROCRYPT15]

- Secure dot-product 2PC protocols [CSFW01, ACSAC01, KDD02, AusDM07, PAKDD14, NSPW02, ICISC04, HICSS10, WiCOM10, CollaborateCom15]

- Privacy preserving data mining [CRYPTO00, SDM04, KDD05, ESORICS05, CCS15, ICDM03, VLDB Journal 08, SIAM05, Information Systems 09]

Take-away points from this talk

Pretzel :

email encryption + provider-supplied functions + low cost

So, why don't email service providers deploy end-to-end email encryption?