Obfuscating Web User Search Queries via Generative Adversarial Privacy

EE 599 Final Project

Jiang Zhang, Zhongxuan Ruan, Mengwei Yang

2020.4.xx

AOL Dataset

UserID	Query	QueryTime	Rank	Link
81943	are people who have asthma prone to get lung cancer	3/7/06 23:26	2	http://kidshealth.org
81943	is pronounced lung cancer leaded into lung cancer	3/7/06 23:33		
81943	if you have asthma can it lead to lung cancer	3/7/06 23:35	6	http://www.lungusa.org

- Preprocessing:
  - Filter the dataset by keywords from two topic: cancer, pregnancy.
  - Classify the dataset into three categories: cancer related, pregnancy related, and other.
  - Each category contains 1000 users, with queries related to each topic.
- Goal:
  - Mutate word tokens in web search queries
  - Decrease its privacy risk: category inference accuracy
  - Maintain its stealthiness (utility metric1): the obfuscated query should be meaningful instead of random words
  - Reduce obfuscation cost (utility metric2): the maximum likelihood estimation loss

Motivation

- Baseline solution:
  - Delete words with high privacy risk (e.g. cancer, pregnancy)
  - Mutate words randomly
  - Mutate words with differential privacy
- Limitations:
  - May not capture the correlation among words in the query
  - Context-free, no adversary

Our approach: SeqGAN with multiple objectives

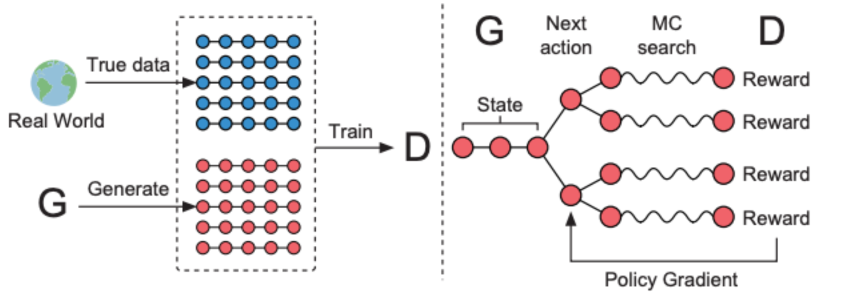


Figure 1: The illustration of SeqGAN. Left: D is trained over the real data and the generated data by G. Right: G is trained by policy gradient where the final reward signal is provided by D and is passed back to the intermediate action value via Monte Carlo search.

Fig. 1. Original SeqGAN (single objective).

* Discriminator: predict where a query is real, to make the obfuscated query meaningful.

* Adversary: predict the category of a query, to enhance the privacy of user query.

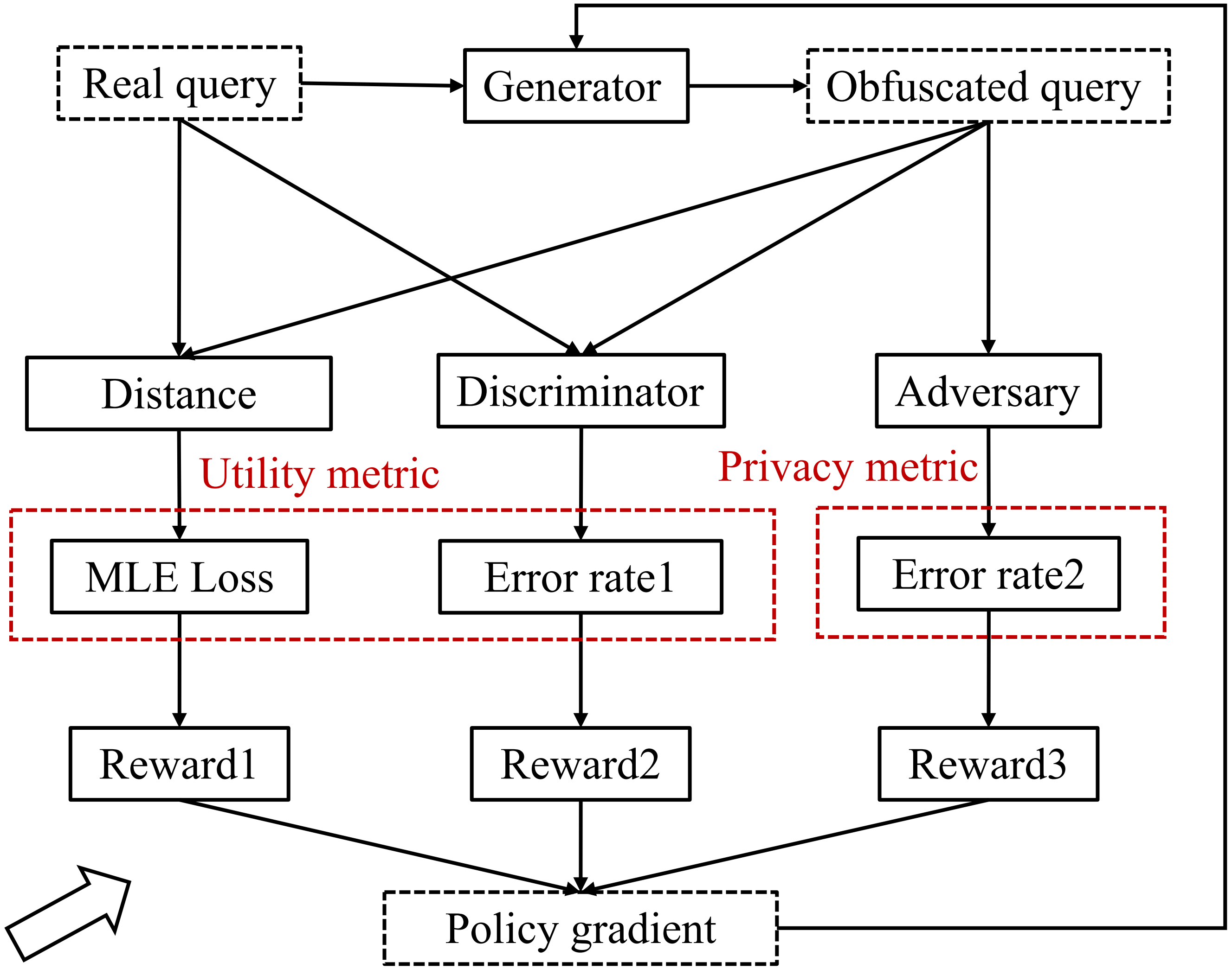


Fig. 2. Our SeqGAN with multiple objectives.

Policy gradient (theory part)

$$J(\theta) = \mathbb{E}[R_T|s_0, \theta] = \sum_{y_1 \in \mathcal{Y}} G_\theta(y_1|s_0) \cdot Q_{D_\phi}^{G_\theta}(s_0, y_1), \qquad (1)$$

$$Q_{D_\phi}^{G_\theta}(a = y_T, s = Y_{1:T-1}) = D_\phi(Y_{1:T}). \qquad (2)$$

$$\left\{Y_{1:T}^1, \ldots, Y_{1:T}^N\right\} = \mathbf{MC}^{G_\beta}(Y_{1:t}; N), \qquad (3)$$

$$Q_{D_\phi}^{G_\theta}(s = Y_{1:t-1}, a = y_t) = \qquad (4)$$
$$\begin{cases} \frac{1}{N}\sum_{n=1}^{N} D_\phi(Y_{1:T}^n),\ Y_{1:T}^n \in \mathbf{MC}^{G_\beta}(Y_{1:t}; N) & \text{for} \quad t < T \\ D_\phi(Y_{1:t}) & \text{for} \quad t = T, \end{cases}$$

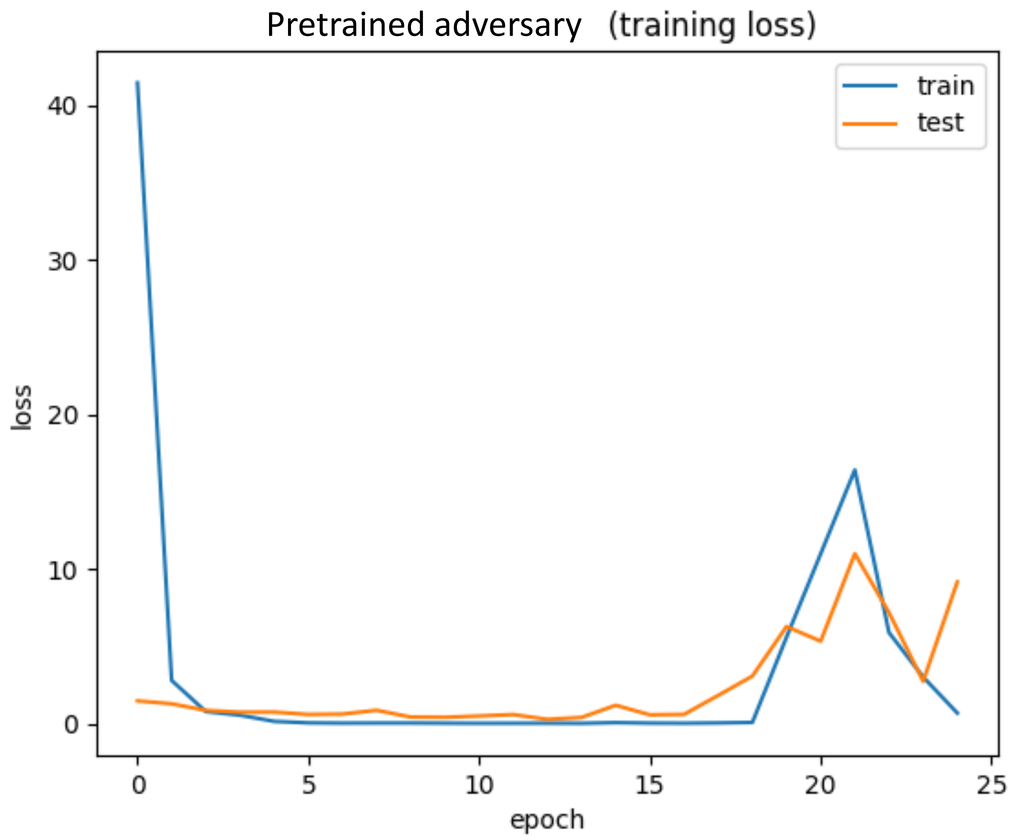
$$\min_\phi -\mathbb{E}_{Y\sim p_{\text{data}}}[\log D_\phi(Y)] - \mathbb{E}_{Y\sim G_\theta}[\log(1 - D_\phi(Y))]. \qquad (5)$$

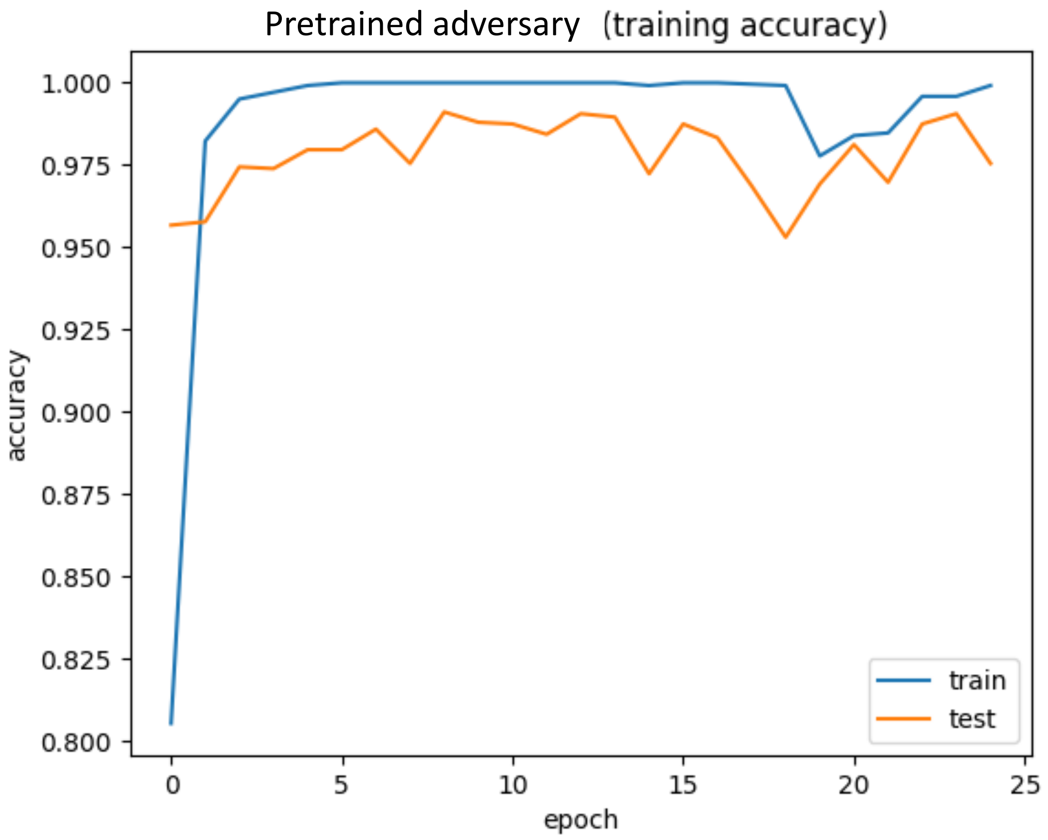
$$\nabla_\theta J(\theta) = \sum_{t=1}^{T} \mathbb{E}_{Y_{1:t-1}\sim G_\theta}\Big[\sum_{y_t \in \mathcal{Y}} \nabla_\theta G_\theta(y_t|Y_{1:t-1}) \cdot Q_{D_\phi}^{G_\theta}(Y_{1:t-1}, y_t)\Big]. \qquad (6)$$

$$\nabla_\theta J(\theta) \simeq \sum_{t=1}^{T} \sum_{y_t \in \mathcal{Y}} \nabla_\theta G_\theta(y_t|Y_{1:t-1}) \cdot Q_{D_\phi}^{G_\theta}(Y_{1:t-1}, y_t) \qquad (7)$$
$$= \sum_{t=1}^{T} \sum_{y_t \in \mathcal{Y}} G_\theta(y_t|Y_{1:t-1}) \nabla_\theta \log G_\theta(y_t|Y_{1:t-1}) \cdot Q_{D_\phi}^{G_\theta}(Y_{1:t-1}, y_t)$$
$$= \sum_{t=1}^{T} \mathbb{E}_{y_t\sim G_\theta(y_t|Y_{1:t-1})}[\nabla_\theta \log G_\theta(y_t|Y_{1:t-1}) \cdot Q_{D_\phi}^{G_\theta}(Y_{1:t-1}, y_t)],$$

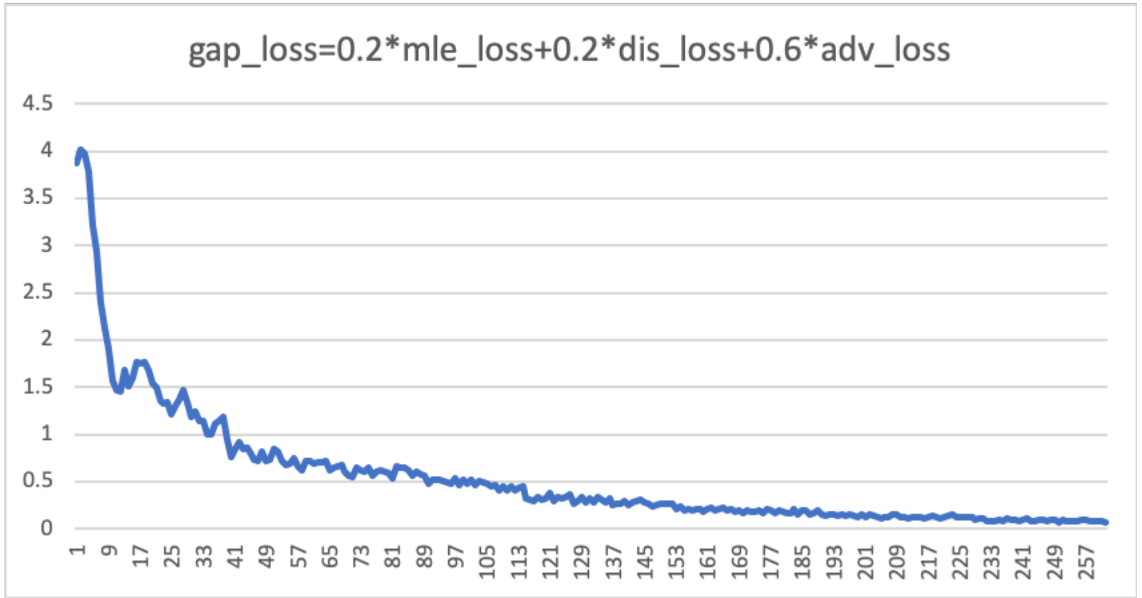
$$\theta \leftarrow \theta + \alpha_h \nabla_\theta J(\theta), \qquad (8)$$

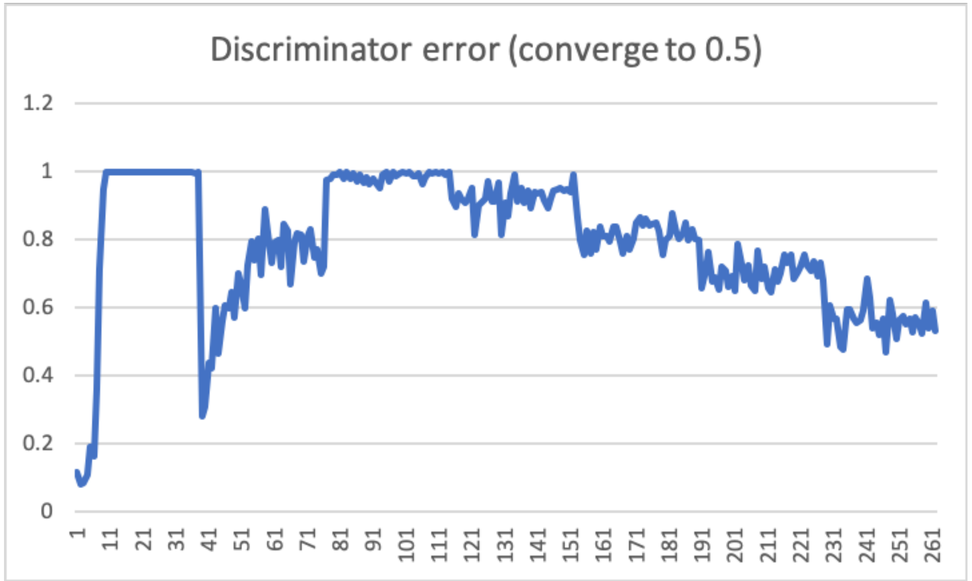
Pretrained Result (Adversary)



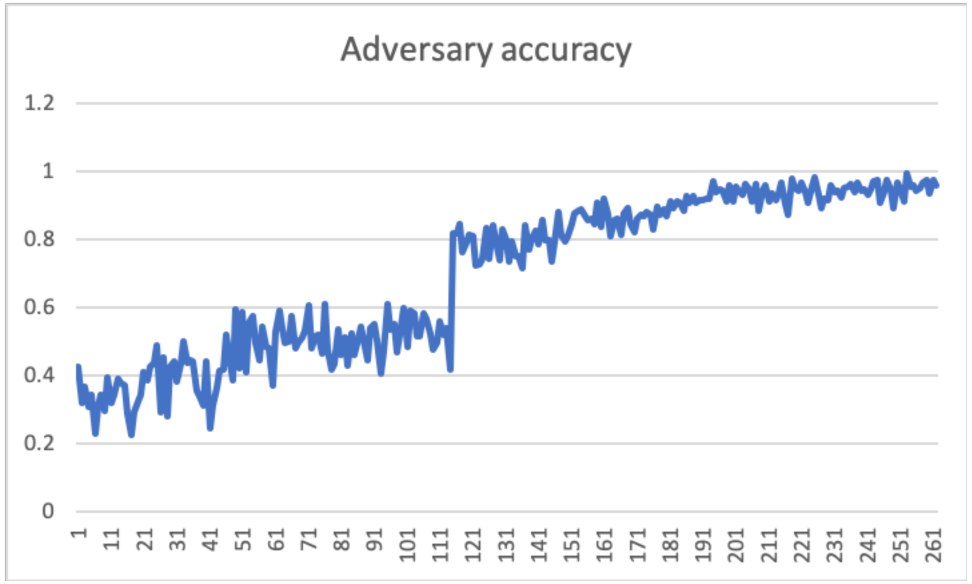


Evaluation






```
[INFO] Target query:  ['<SOS>', 'baby', 'modeling', '<POS>', 'baby', 'modeling', 'agency', '<POS>', 'baby', 'photo'
, 'contest', '<POS>', 'baby', 'photo', 'contest', '<EOS>', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '
*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '
*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*']
[INFO] Predicted query:  ['<SOS>', 'baby', 'modeling', '<POS>', 'baby', 'modeling', 'agency', '<POS>', 'baby', 'pho
to', 'contest', '<POS>', 'baby', 'photo', 'contest', '<EOS>', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*'
, '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*'
, '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*', '*']
```



Explanation: MLE loss -> 0 (Identity transformation)

To do:

- Change the weights of each part in loss function

Content

- Introduction
- Related work
- Approach
- Evaluation
- Conclusion

Introduction

Conclusion