Adversarial Training for Weakly Supervised Event Detection

Xiaozhi Wang[1], Xu Han[1], Zhiyuan Liu[1], Maosong Sun[1], Peng Li[2]

[1]Department of Computer Science and Technology, Tsinghua University
[2]Pattern Recognition Center, WeChat, Tencent Inc.

July 22, 2019

Introduction

- Event Detection: Detect event triggers and identify event types.

Mark Twain and Olivia Langdon *married* in 1870

Event Type: **Marry**

- First stage of the Event Extraction.
- Important for downstream NLP applications.

Challenge: data sparsity

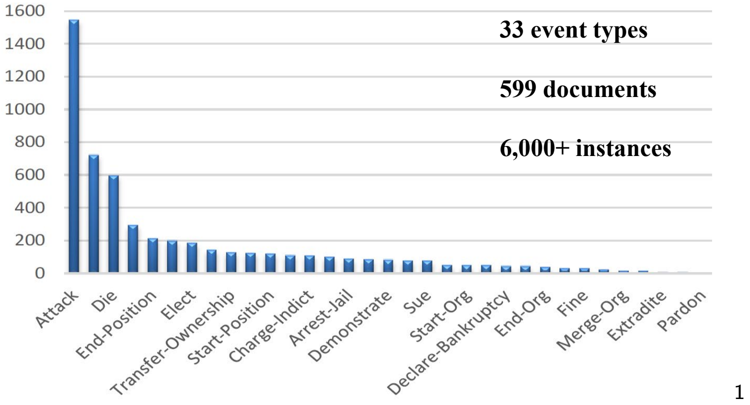
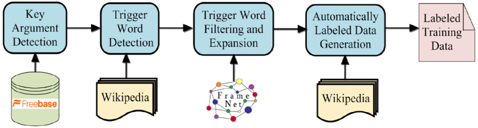
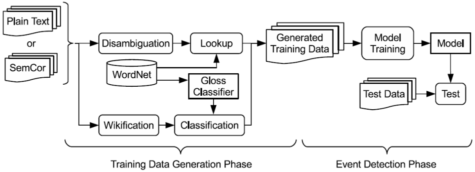


Figure 1: Statistics of ACE 2005 English Data. Thanks Chen et al., 2017.

Related Work: Distant Supervision



(a) Automatically Labeled Data Generation for Large Scale Event Extraction (Chen et al., 2017)



(b) Open-Domain Event Detection using Distant Supervision (Araki et al., 2018)

Related Work: Semi-supervision

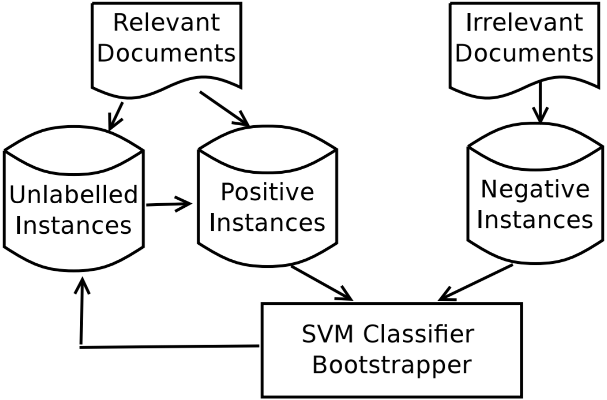


Figure 2: Bootstrapped Training of Event Extraction Classifiers (Huang et al., 2012)

Related Work: Weakness

- Sophisticated pre-defined rules: topic bias.
- Existing instances in knowledge bases: low coverage.

Our Model

- **Adversarial Training** to unsupervisedly denoise data.
- **Trigger-based latent instance discovery strategy** to automatically construct large-scale candidate set with good coverage.

Overall architecture

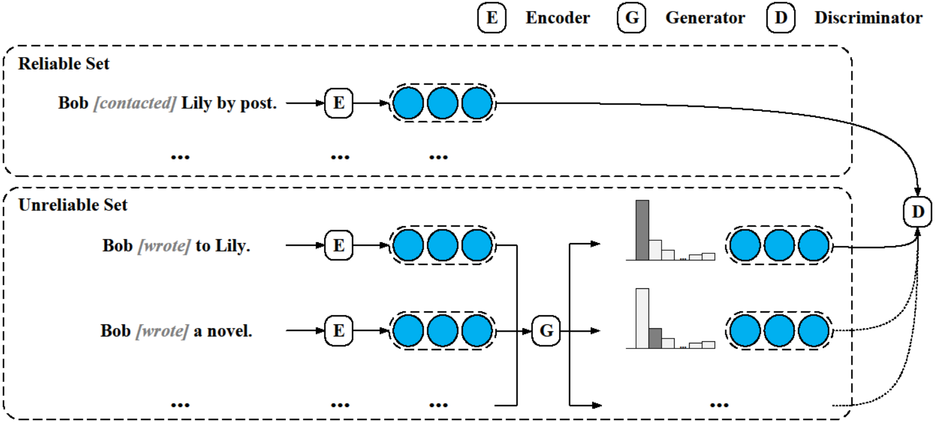


Figure 3: The overall architecture. The event type is Contact.

Adversarial Training

- **Discriminator**
  - To detect events correctly.
  - Should resist noise.
- **Generator**
  - To confuse the discriminators.

Overall architecture

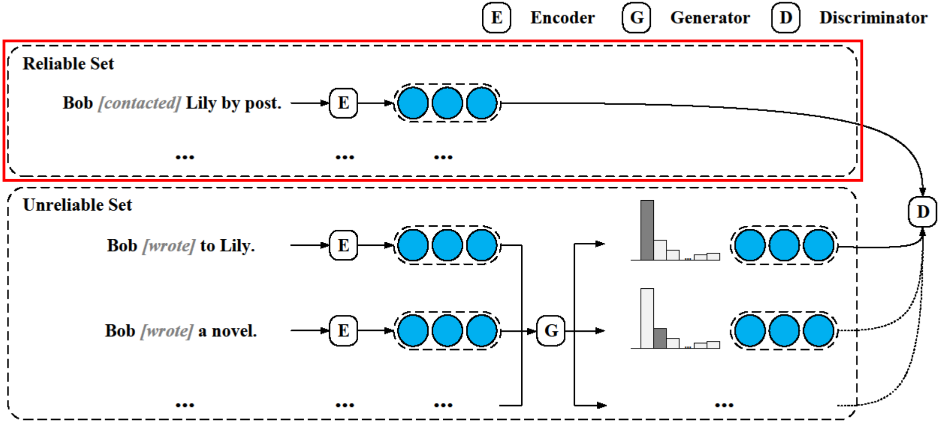


Figure 4: The overall architecture. The event type is Contact.

Overall architecture

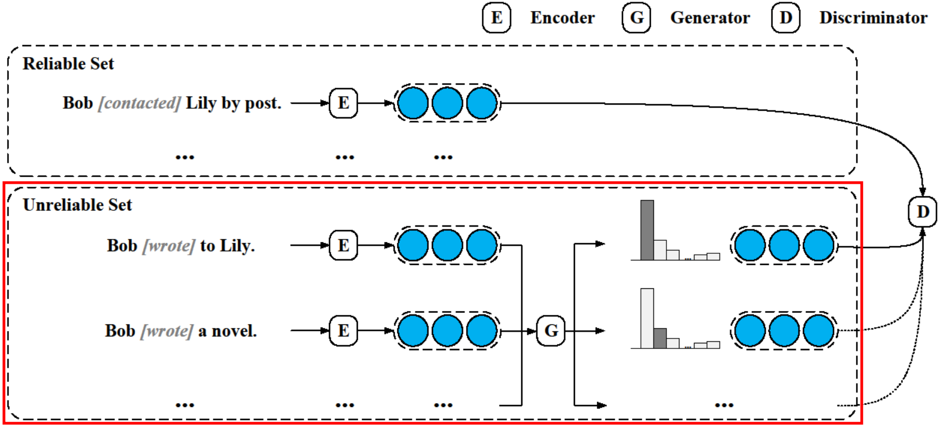


Figure 5: The overall architecture. The event type is Contact.

Adversarial Training

- **Discriminator**
  - $x \in \mathcal{R}$ as positive instances and $x \in \mathcal{U}$ as negative instances.
  - $\phi_D = \max \left(E_{x \sim P_R} \left[\log \left(P(e|x,t) \right) \right] + E_{x \sim P_{\mathcal{U}}} \left[\log \left(1 - P(e|x,t) \right) \right] \right)$.
- **Generator**
  - Select most confusing $x \in \mathcal{U}$ to fool the discriminator.
  - $\phi_G = \max E_{x \sim P_{\mathcal{U}}} \left[\log \left(P(e|x,t) \right) \right]$.

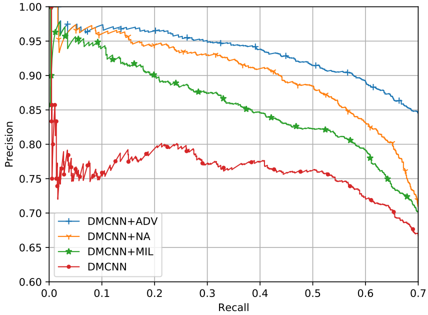
Adversarial Training

- **Discriminator**
  - $x \in \mathcal{R}$ as positive instances and $x \in \mathcal{U}$ as negative instances.
  - $\mathcal{L}_D = -\sum_{x \in \mathcal{R}} \frac{1}{|\mathcal{R}|} \log\left(P(e|x,t)\right) - \sum_{x \in \mathcal{U}} P_{\mathcal{U}}(x) \log\left(1 - P(e|x,t)\right)$.
- **Generator**
  - Select most confusing $x \in \mathcal{U}$ to fool the discriminator.
  - Confusing score: $P_{\mathcal{U}}(x) = \frac{\exp\left(f(x)\right)}{\sum_{\hat{x} \in \mathcal{U}} \exp\left(f(\hat{x})\right)}$.
  - $\mathcal{L}_G = -\sum_{x \in \mathcal{U}} P_{\mathcal{U}}(x) \log\left(P(e|x,t)\right)$.

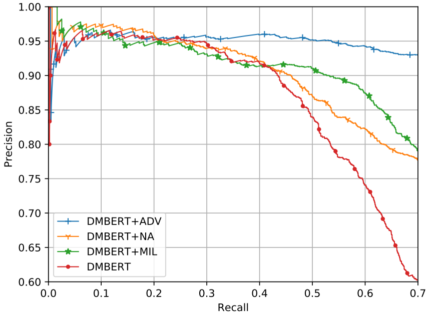
Method

- Pre-train a normal model in the noisy dataset, and set a threshold for the confidence scores of the model.
- Reliable Set $\mathcal{R}$: instances with higher confidence.
- Unreliable Set $\mathcal{U}$: instances with lower confidence.
- Initialize the encoders with the pre-trained model, then conduct adversarial training.

Experiments



(a) Precision-Recall Curves for the CNN models.



(b) Precision-Recall Curves for the BERT models.

Method

- Pre-train a model on the small high-quality dataset.
- Retrieve candidate instances from a large-scale raw dataset to construct a large candidate set.
- Automatically label the candidate set with a pre-trained model.
- Reliable Set $\mathcal{R}$: Small-scale human-annotated data.
- Unreliable Set $\mathcal{U}$: Large-scale auto-labeled data.
- Adversarial training, then the instances recommend by the generator will be trusted.

Trigger-based latent instance discovery strategy

- Intuition: If a word serves as the trigger in a known instance, the raw sentences mentioning it may also express an event.
- Retrieve the sentences in NYT corpus which contains triggers in ACE 2005.
- Simple but effective.

Experiments

Method	Trigger Identification +Classification		
	P	R	F1
Li's Joint	73.7	62.3	67.5
JRNN	66.0	**73.0**	69.3
ANN-FN	77.6	65.2	70.7
DLRNN	77.2	64.9	70.5
GMLATT	78.9	66.9	72.4
DMCNN+Chen's DS	75.7	66.0	70.5
Bi-LSTM+GAN	71.3	74.7	73.0
GCN-ED	**77.9**	68.8	73.1
DMCNN	75.6	63.6	69.1
DMCNN+Boot	77.7	65.1	70.8
DMBERT	77.6	71.8	74.6
DMBERT+Boot	**77.9**	72.5	**75.1**

Table 1: The overall performance (%) of different models on ACE-2005.

Manual Evaluation

Method	Average Precision	Fleiss's Kappa
chen2017automatically	88.9	–
zeng2018scale	91.0	–
Our First Iteration	91.7	61.3
Our Second Iteration	87.5	52.0

Table 2: The human evaluation results (%) of auto-labeled data.

Case Study

Event-Type: `Justice` Subtype: `Sue`	
In ACE-2005	Dell **sued** for "bait and switch" and false promises.
Discovered	1. The lawyers for the four former state officials who have been **sued** told the jurors ... 2. But **litigation** held up the project until ....

Table 3: The examples with highlighting triggers.

Conclusion and Future work

- An effective adversarial training method for weakly supervised event detection.
  - Denoise and enhance distantly supervised models.
  - Automatically collect more diverse and accurate training data.
- Future work
  - Extract event arguments.
  - A large-scale dataset.

The End

Thanks for listening.
Questions are welcome.