Adversarial Training with Contrastive Learning in NLP

Procesamiento de Lenguaje Natural

Francisco Javier Sáez Maldonado

14 de marzo de 2022

Máster en Ciencia de Datos

Escuela Politécnica Superior
Universidad Autónoma de Madrid

Índice

Introducción

- **Tarea**: Modelado del lenguage (LM) y Traducción automática (NMT)

Introducción

- **Tarea**: Modelado del lenguage (LM) y Traducción automática (NMT)
- **Objetivo**: Conseguir modelos que sean más robustos semánticamente:

$$\text{Inputs parecidos} \implies \text{Outputs parecidos}$$

Herramientas

Herramientas

Adversarial Training

Adversarial Training

Definición (Adversarial Learning)

Técnica usada en el aprendizaje automático para, usando información sobre un modelo, crear ataques maliciosos para causar fallos en el modelo

Adversarial Training

Definición (Adversarial Learning)

Técnica usada en el aprendizaje automático para, usando información sobre un modelo, crear ataques maliciosos para causar fallos en el modelo

Definición (Adversarial example)

Ejemplo diseñado para engañar al modelo, creado introduciendo una perturbación en un ejemplo original.

¿ Cómo ayuda el aprendizaje adversario a nuestros modelos ?

Técnicas de creación ejemplos adversarios en NLP

- Visuales: (Morris u. a., 2020).

Original Input	This film has a special place in my heart	Positive
Adversarial example	This film has a special plcae in my herat	Negative

Cuadro 1: Ejemplo extraído de (Gao u. a., 2018)

- Semánticas:(Jin u. a., 2019)

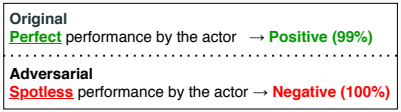

Figura 1: Ejemplo extraído de (Jin u. a., 2019)

Ejemplos adversarios

Dada una secuencia $s = \{x_1, \ldots, x_T\}$ de tokens

1. Creamos una representación embebida en un espacio continuo

$$\mathbf{E}x_i = e_i.$$

2. Añadimos una pequeña perturbación en el embedding

$$e_i' = e_i - \epsilon \frac{g}{\|g\|_2},$$

siendo $g = \nabla_{e_i} J(s, \theta)$ y J la función de coste.

Función de coste actual:

$$\mathcal{J}(\theta) = \sum_s \mathcal{L}(s, \theta) + \alpha \sum_{s'} \mathcal{L}_{adv}(s', \theta), \quad \alpha \in [0, 1].$$

Herramientas

Contrastive Learning

Contrastive Learning

Idea: Acercar las representaciones de ejemplos positivos (de la misma clase) y alejar las de los ejemplos negativos (resto de ejemplos).

Ejemplo

Original: Elefante. Positivo: Hipopótamo. Negativo: Pistola.

Definición (Pérdida contrastiva)

Sean a_i las entradas originales, p_{a_i} ejemplos positivos y n_a ejemplos negativos. Se define la pérdida contrastiva como:

$$\mathcal{L}_{cont} = -\sum_{a_i \in A} \log \frac{\exp(a_i \cdot p_{a_i}/\tau)}{\sum_{n_a \in A - \{a_i\}} \exp(a_i \cdot n_a/\tau)}$$

Framework

Elementos

Consideramos :

- $\mathbf{S} = \{s_1, \ldots, s_B\}$ un conjunto de frases, donde cada frase $s_k = \{x_{k1}, \ldots, x_{kN}\}$ tiene N tokens.

Elementos

Consideramos :

- $\mathbf{S} = \{s_1, \ldots, s_B\}$ un conjunto de frases, donde cada frase $s_k = \{x_{k1}, \ldots, x_{kN}\}$ tiene N tokens.
- $\mathbf{E}s_k = \{\mathbf{E}x_{k1}, \ldots, \mathbf{E}x_{kN}\} = \{e_{k1}, \ldots e_{kN}\}$ el embedding al espacio continuo de cada frase.

Elementos

Consideramos :

- $\mathbf{S} = \{s_1, \ldots, s_B\}$ un conjunto de frases, donde cada frase $s_k = \{x_{k1}, \ldots, x_{kN}\}$ tiene N tokens.
- $\mathbf{E}s_k = \{\mathbf{E}x_{k1}, \ldots, \mathbf{E}x_{kN}\} = \{e_{k1}, \ldots e_{kN}\}$ el embedding al espacio continuo de cada frase.
- El vocabulario $\mathcal{V}$ y un subconjunto **restringido** del mismo del que excluimos palabras incompletas (caracteres individuales o símbolos) $\mathcal{V}_R$

Elementos

Consideramos :

- $\mathbf{S} = \{s_1, \ldots, s_B\}$ un conjunto de frases, donde cada frase $s_k = \{x_{k1}, \ldots, x_{kN}\}$ tiene N tokens.
- $\mathbf{E}s_k = \{\mathbf{E}x_{k1}, \ldots, \mathbf{E}x_{kN}\} = \{e_{k1}, \ldots e_{kN}\}$ el embedding al espacio continuo de cada frase.
- El vocabulario $\mathcal{V}$ y un subconjunto **restringido** del mismo del que excluimos palabras incompletas (caracteres individuales o símbolos) $\mathcal{V}_R$
- La función que nos da la restricción a $\mathcal{V}_R$:

$$\mathcal{M}(\mathbf{E}x_{ki}) = \begin{cases} 1 & \text{if } x_{ki} \in \mathcal{V}_R \\ 0 & \text{otherwise} \end{cases}$$

(evita tomar candidatos a modificación adversaria sin sentido).

Elementos

Consideramos :

- $\mathbf{S} = \{s_1, \ldots, s_B\}$ un conjunto de frases, donde cada frase $s_k = \{x_{k1}, \ldots, x_{kN}\}$ tiene N tokens.
- $\mathbf{E}s_k = \{\mathbf{E}x_{k1}, \ldots, \mathbf{E}x_{kN}\} = \{e_{k1}, \ldots e_{kN}\}$ el embedding al espacio continuo de cada frase.
- El vocabulario $\mathcal{V}$ y un subconjunto **restringido** del mismo del que excluimos palabras incompletas (caracteres individuales o símbolos) $\mathcal{V}_R$
- La función que nos da la restricción a $\mathcal{V}_R$:

$$\mathcal{M}(\mathbf{E}x_{ki}) = \begin{cases} 1 & \text{if } x_{ki} \in \mathcal{V}_R \\ 0 & \text{otherwise} \end{cases}$$

  (evita tomar candidatos a modificación adversaria sin sentido).
- Llamaremos h_{kj} a la representación obtenida de la frase s_k.

Adversarial Training with Contrastive Learning (ATCL)

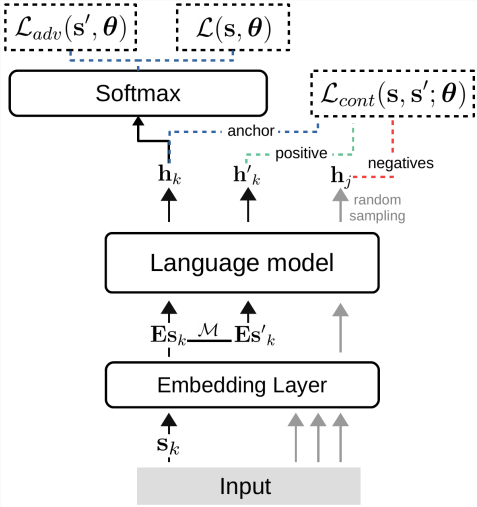

Figura 2: ACTL Framework. Image obtained from the original paper (Rim u. a., 2021).

Consideraciones sobre ATCL

- Con el ejemplo adversario alejamos la representación de la frase original, y forzamos la cercanía mediante contrastive learning.
- Para la función de pérdida contrastiva, $\mathcal{L}_{\text{cont}}$, los ejemplos negativos se samplean del conjunto $\mathbf{H}(\mathbf{S}) - \{h_{kj}\}$, para evitar tomar a la misma h_{kj} como ejemplo negativo.

Definición (Función de pérdida de ATCL)

La función de pérdida que se utiliza para entrenar el framework ATCL es:

$$\mathcal{J}(\theta)_{ATCL} = \sum_{\mathbf{s},\mathbf{s}'} (\mathcal{L} + \alpha \mathcal{L}_{adv} + \beta \mathcal{L}_{cont})$$

Experimentos y resultados

Modelo de Lenguaje

Datasets:

Traducción automática

Conclusiones

- omega Yea

Gracias por su atención

Referencias

[Gao u. a. 2018] Gao, Ji ; Lanchantin, Jack ; Soffa, Mary L. ; Qi, Yanjun: Black-box Generation of Adversarial Text Sequences to Evade Deep Learning Classifiers. (2018). – URL http://arxiv.org/abs/1801.04354

[Jin u. a. 2019] Jin, Di ; Jin, Zhijing ; Zhou, Joey T. ; Szolovits, Peter: Is BERT Really Robust? Natural Language Attack on Text Classification and Entailment. (2019). – URL http://arxiv.org/abs/1907.11932

[Morris u. a. 2020] Morris, John X. ; Lifland, Eli ; Yoo, Jin Y. ; Qi, Yanjun: TextAttack: A Framework for Adversarial Attacks in Natural Language Processing. (2020). – URL https://arxiv.org/abs/2005.05909

[Rim u. a. 2021] Rim, Daniela N. ; Heo, DongNyeong ; Choi, Heeyoul: Adversarial Training with Contrastive Learning in NLP. (2021). – URL https://arxiv.org/abs/2109.09075