Contrastive Representation Learning in Language Model

1 - FaceNet: A Unified Embedding for Face Recognition and Clustering (Florian Schroff, Dmitry Kalenichenko, James Philbin)

2 - SimCSE: Simple Contrastive Learning of Sentence Embeddings (Tianyu Gao†∗ Xingcheng Yao‡∗ Danqi Chen†)

3 - An efficient framework for learning sentence representations (Lajanugen Logeswaran, Honglak Lee)

GIST AIGS/EECS INFONET 연구실

통합과정 김수민

Contrastive Representation Learning

Contrastive Learning

1. Definition

Anchor를 기준으로 특성이 가까운 이미지 또는 문장 또는 단어와 거리가 가까울 수 있게 학습시키는 것이다. 반대로, 특성이 다른 것은 멀어지게 학습 시키는 것이다.

The goal is to learn a representation of data such that similar instances are close together in the representation space, while dissimilar instances are far apart.

Problem setting: In person sighting, figure 1 is (Chick)병아리, figure 2 is (eagle)독수리, and figure 3 is a tiger. And they can be found in different representations. However, without special learning or methods, computers can not find different things. They just know the RGB values or vectors of every image.

Method

Contrastive Training Objectives: Loss

1. Triplet Loss

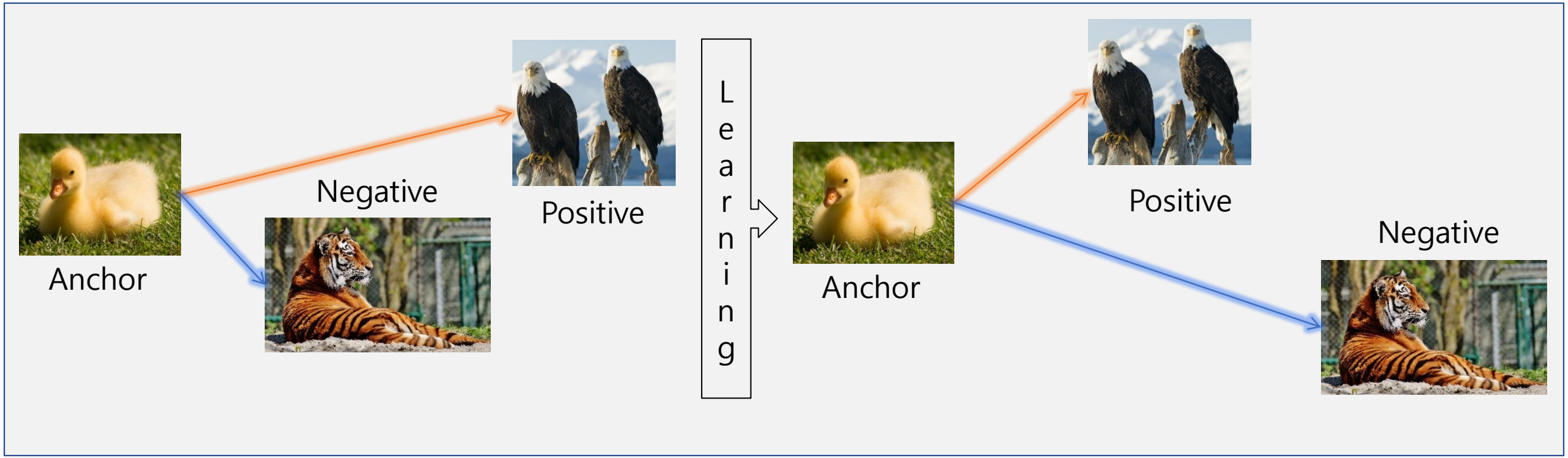
$$\mathcal{L}_{\text{triplet}}(\mathbf{x}, \mathbf{x}^+, \mathbf{x}^-) = \sum_{\mathbf{x} \in \mathcal{X}} \max\left(0, \| f(\mathbf{x}) - f(\mathbf{x}^+) \|_2^2 - \| f(\mathbf{x}) - f(\mathbf{x}^-) \|_2^2 + \epsilon\right)$$

where the margin parameter ϵ is configured as the minimum offset between distances of similar vs dissimilar pairs.

It is crucial to select challenging x^- to truly improve the model.

Contrastive Learning

1. Definition



The Contrastive learning method (Triplet Loss) minimizes the distance between an anchor and a positive, both of which have the same identity, and maximizes the distance between the anchor and a negative of a different identity.

Contrastive Learning in Language model

1. What is the Difference?
   - **Contrastive Learning**: Focuses on distinguishing between similar and dissimilar data points by bringing similar pairs closer and pushing dissimilar pairs apart in the embedding space.
   - **Traditional Learning**: Typically focuses on minimizing a loss function based on direct prediction errors (e.g., classification error).
2. Why Sentences are Weak for Augmentation
   - Sentences are complex and have a structured meaning that can be difficult to preserve when creating augmented versions. Simple augmentation techniques like rephrasing might change the underlying meaning.
   - The meaning of sentences is heavily dependent on context, making it challenging to generate valid augmentations without introducing noise.
3. Embedding Necessity (It usually needs embedding in the model)
   - **Representational Learning:** Embeddings are needed to transform sentences into a continuous vector space where semantic similarities and differences can be measured effectively. These embeddings capture the syntactic and semantic properties of the text.
   - **Augmentation Compatibility:** Contrastive learning often requires embeddings to compare and contrast different examples within a shared vector space. This facilitates the learning process by providing a consistent framework for measuring similarity and dissimilarity.

1. SimCSE: Simple Contrastive Learning of Sentence Embeddings

1. Proposed Methods
2. Note that **z is just the standard dropout mask** in Transformers and we do not add any additional dropout.

$$\ell_i = -\log \frac{e^{\text{sim}(\mathbf{h}_i, \mathbf{h}_i^+)/\tau}}{\sum_{j=1}^{N} e^{\text{sim}(\mathbf{h}_i, \mathbf{h}_j^+)/\tau}}$$

Contrastive loss (originally)

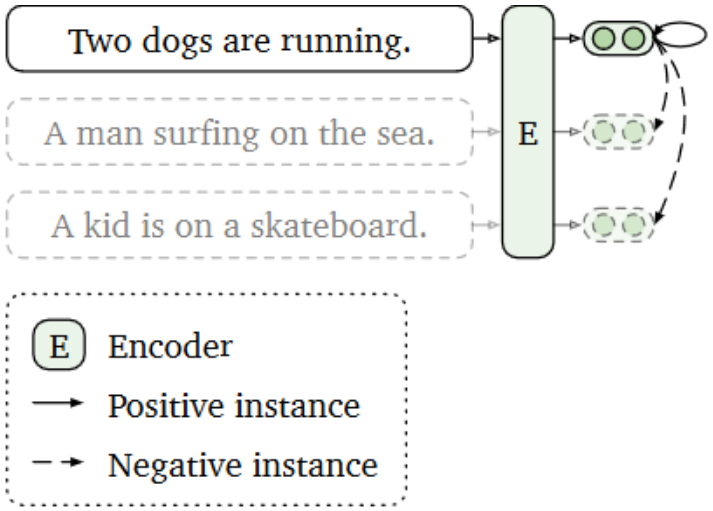
$$\ell_i = -\log \frac{e^{\text{sim}(\mathbf{h}_i^{z_i}, \mathbf{h}_i^{z_{i'}})/\tau}}{\sum_{j=1}^{N} e^{\text{sim}(\mathbf{h}_i^{z_i}, \mathbf{h}_j^{z_{j'}})/\tau}}$$

The objective loss function of SimCSE

Method

1. SimCSE: Simple Contrastive Learning of Sentence Embeddings

Proposed Methods: **Unsupervised SimCSE**



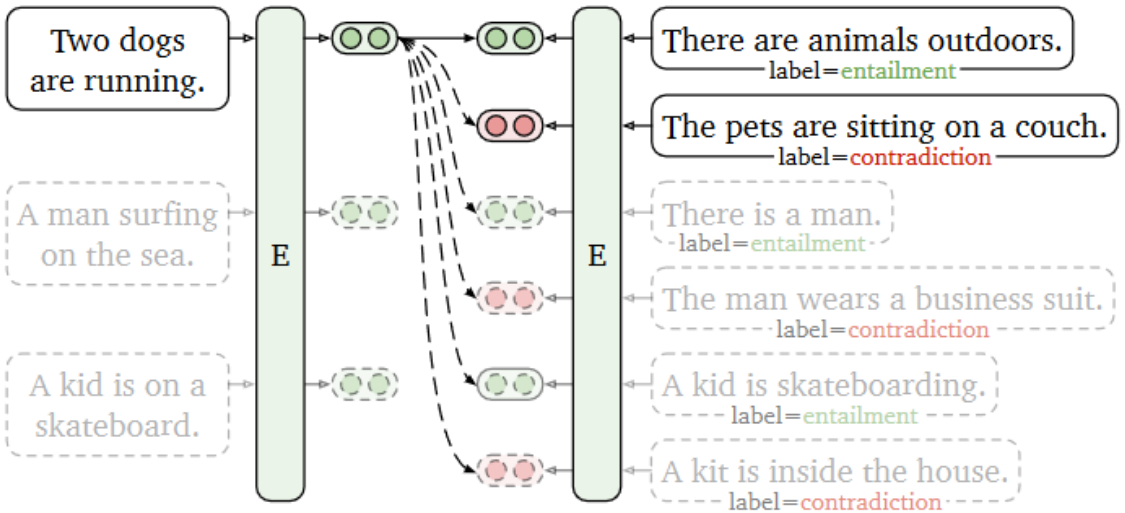
- **Encoder (E)**: The encoder is responsible for transforming input sentences into their corresponding embeddings. This transformation is the core of the SimCS E method.
- **Positive Instance (solid arrow)**: This indicates a sentence that is similar or identical to the anchor sentence (e.g., a slightly different form of "Two dogs are running").
- **Negative Instance (dashed arrow)**: This indicates a sentence that is dissimilar to the an chor sentence (e.g., "A man surfing on the sea").

The encoder takes an input sentence and generates an embedding. The positive in stance is used to ensure that similar sentences have closer embeddings. The negat ive instance helps in ensuring that dissimilar sentences have embeddings that are further apart.

Method

1. SimCSE: Simple Contrastive Learning of Sentence Embeddings

Proposed Methods: **Supervised SimCSE**

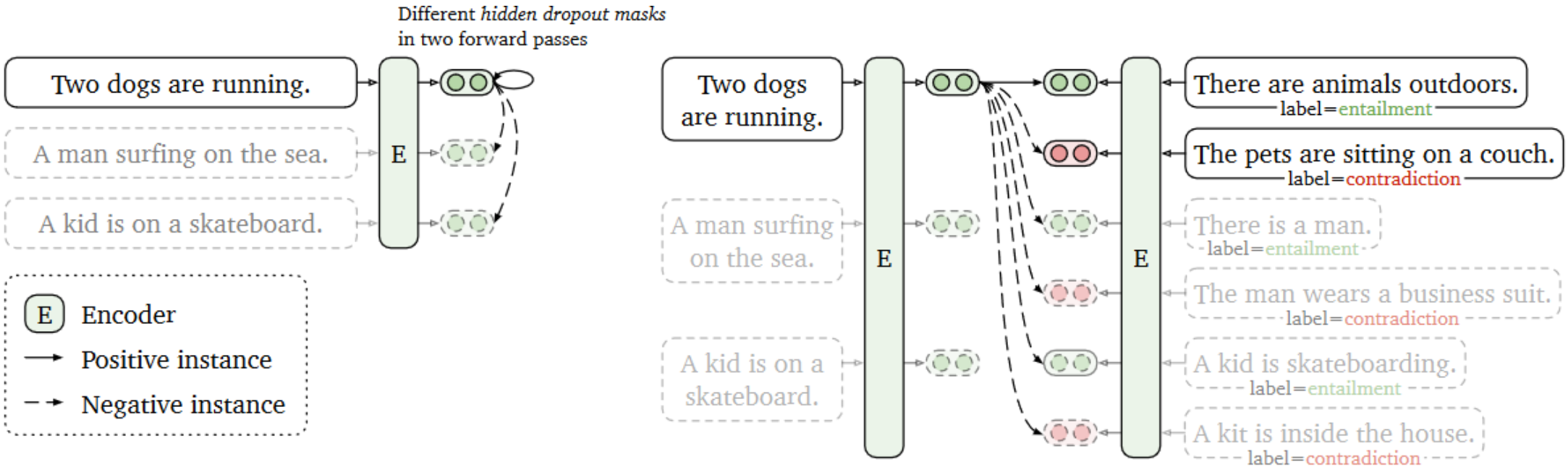


1. **Positive Instance**: Sentences with an entailment relationship (e.g., "Two dogs are running" and "There are animals outdoors").
2. **Negative Instance**: Sentences with a contradiction relationship (e.g., "Two dogs are running" and "The pets are sitting on a couch").
3. The encoder generates embeddings for each input sentence. The embeddings are then compared to determine if they represent entailment (positive instances) or contradiction (negative instances). To connect sentences with entailment labels and dashed lines for contradiction labels.

Method

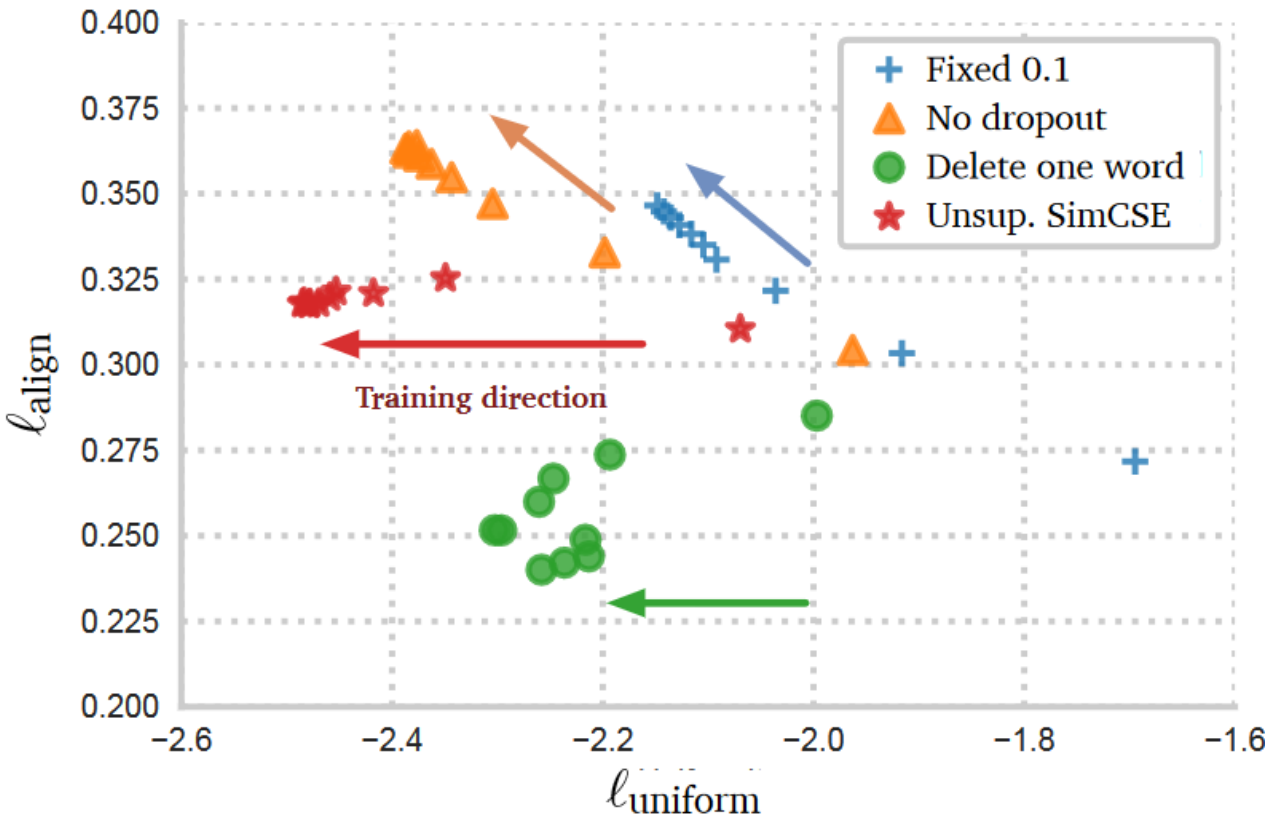
1. SimCSE: Simple Contrastive Learning of Sentence Embeddings

Compared Unsupervised SimCSE and Supervised SimCSE



Unsupervised SimCSE predicts the input sentence itself from in-batch negatives, with different hidden dropout ma sks applied. **Supervised SimCSE** leverages the NLI datasets and takes the entailment (premise hypothesis) pairs as positives, and contradiction pairs as well as other in-batch instances as negatives.

Result



They treat dropout as data augmentation for text sequences.

A sample is fed into the encoder twice with different dropout masks and these two versions are the positive pair whereas the other in-batch samples are considered negative pairs.

It feels quite similar to the *cutoff augmentation, but dropout is more flexible with less well-defined semantic meaning of what content can be masked off.

*cutoff augmentation: It involves randomly setting a subset of features or a portion of the input data (e.g., pixels in an image, words in a text) to zero. This process can be seen as a form of regularization, forcing the model to rely on the remaining, non-zeroed features to make predictions, thus reducing overfitting and improving performance on unseen data.

Key words and Insights

1. Domain adaptation
   - Adjusting a model trained in one domain to perform well in another.
   - **Role in Contrastive Learning:** Helps in transferring learned representations from a source domai n to a target domain, improving the model's robustness across different data distributions.
2. Continual Pretraining
   - Training a model on a large dataset to learn general features before fine-tuning on a specific task.
   - **Role in Contrastive Learning**: Provides a strong initialization by learning general semantic repre sentations, which can then be refined through contrastive learning to distinguish between similar a nd dissimilar examples.

Thank you for Listening