ALTeGraD 2023 Data Challenge

Molecule Retrieval with Natural Language Queries

RAMMAL Ahmad, EL MALLAH Rim, BENYAMINA Elyas

10 octobre 2024

1 Introduction

2 Model
- Text encoder
- Graph encoder
- Loss Functions
- Training and computational resources

3 Results
- Best Model
- Results obtained
- Comments

4 Conclusion

ALTeGraD 2023 Data Challenge

AR,REM,EB

Introduction

Model

Text encoder

Graph encoder

Loss Functions

Training and computational resources

Results

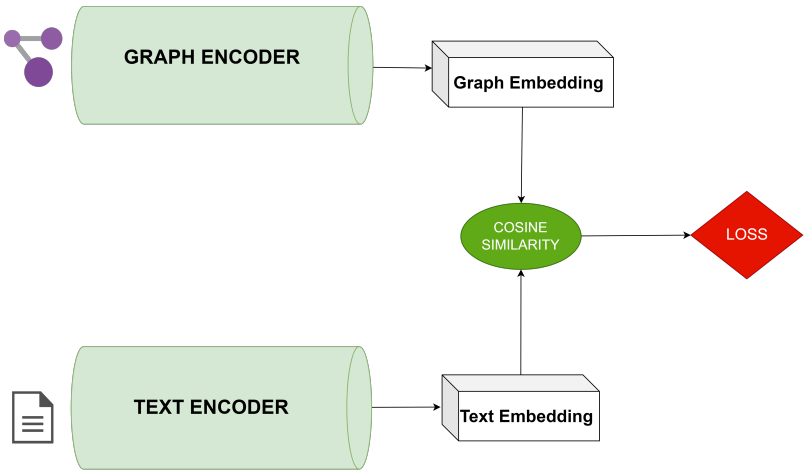
Best Model

Results obtained

Comments

Conclusion

- Objective : Retrieve molecules from text queries.
- Challenge : Texts and molecules have different representations.
- Solution : Co-training framework.
  - Simultaneous training of text and molecule encoders.
  - Uses contrastive learning.
- Goal : Map similar text-molecule pairs closely, push dissimilar pairs apart.
- Outcome : Enhanced molecule retrieval from text queries.



ALTeGraD 2023 Data Challenge

AR,REM,EB

Introduction

Model

Text encoder

Graph encoder

Loss Functions

Training and computational resources

Results

Best Model

Results obtained

Comments

Conclusion

DistilBERT

- A distilled version of BERT, designed to be smaller and faster.
- Retains 97% of BERT's language understanding capabilities with 40% fewer parameters.
- Optimized for speed and efficiency, making it suitable for resource-constrained environments.

SciBERT [SciBERT : Pretrained Language Model for Scientific Text (Iz Beltagy et al., EMNLP 2019)]

- A variant of BERT trained on a large corpus of scientific text.
- Tailored for natural language processing tasks in the scientific domain.
- Improves performance on scientific datasets by better capturing domain-specific jargon and concepts.

- **Key Difference :** DistilBERT is a general-purpose, lightweight model, while SciBERT specializes in understanding scientific text.

ALTeGraD 2023 Data Challenge

AR,REM,EB

Introduction

Model

Text encoder

Graph encoder

Loss Functions

Training and computational resources

Results

Best Model

Results obtained

Comments

Conclusion

Text tokenizer :

- DistilBERT, SciBERT and SBERT were tested as text tokenizers
- In practice, SciBERT worked better in out experiments

Final text embdedding :

- The text tokenizer gives 768-dimensional embeddings
- Additional linear layer to reduce dimension to 350 (improve efficiency, and high enough to keep sufficient information)
- Normalization layer, so that embeddings obtained are more likely to match the one obtained by the graph encoder

ALTeGraD 2023 Data Challenge

AR,REM,EB

Introduction

Model

Text encoder

Graph encoder

Loss Functions

Training and computational resources

Results

Best Model

Results obtained

Comments

Conclusion

Different layers

1. **GCN (Graph Convolutional Networks)** : Integrates features from a node's neighbors, simulating convolutional operations on graphs.
2. **GAT (Graph Attention Networks)** : Applies attention over neighbors, prioritizing information flow from more relevant nodes.
3. **GIN (Graph Isomorphism Networks)** : Enhances sensitivity to graph's structural nuances, aiming at distinguishing non-isomorphic graphs.

Use of multiple layers :

1. 3 GCN layers (baseline model)
2. 3 GAT layers (best performance), 5 GAT layers (leads to overfitting)
3. Use of GCN, GAT and GIN independently and combine results (using mean)

We add a normalisation layer at the end.

ALTeGraD 2023 Data Challenge

AR,REM,EB

Introduction

Model

Text encoder

Graph encoder

Loss Functions

Training and computational resources

Results

Best Model

Results obtained

Comments

Conclusion

Contrastive loss (used in baseline model)

With logits $= v_1 \times v_2^T$, n the number of rows of logits,

labels $= \text{diag}(\text{range}(0, \text{logits.shape}[0])) = \begin{bmatrix} 0 & & \\ & \ddots & \\ & & n-1 \end{bmatrix}$ and CE the cross-entropy loss, we define the contrastive loss as follows :

$$\text{CL}(v_1, v_2) = \text{CE}(\text{logits}, \text{labels}) + \text{CE}(\text{logits}^T, \text{labels})$$

We also define the contrastive loss with temperature parameter :

$$\text{CL}_\tau(v_1, v_2) = \text{CE}\left(\frac{\text{logits}}{\tau}, \text{labels}\right) + \text{CE}\left(\frac{\text{logits}^T}{\tau}, \text{labels}\right)$$

ALTeGraD 2023 Data Challenge

AR,REM,EB

Introduction

Model

Text encoder

Graph encoder

Loss Functions

Training and computational resources

Results

Best Model

Results obtained

Comments

Conclusion

Negative contrastive loss [Text2Mol : Cross-Modal Molecule Retrieval with Natural Language Queries (Edwards et al., EMNLP 2021)]

With logits $= v_1 \times v_2^T$, eye $=$ diag_embed(labels) and BCEL the the Binary Cross-Entropy with Logits Loss, we define the negative sampling contrastive loss as follows :

$$\text{NegativeSamplingCL}(v_1, v_2, \text{labels}) = \text{BCEL}\left(\text{logits}, \text{eye}\right) + \text{BCEL}\left(\text{logits}^T, \text{eye}\right)$$

We also define the negative sampling contrastive with a temperature scaling parameter :

$$\text{NegativeSamplingCL}_\tau(v_1, v_2, \text{labels}) = \text{BCEL}(\frac{\text{logits}}{\tau}, \text{eye}) + \text{BCEL}(\frac{\text{logits}^T}{\tau}, \text{eye})$$

- We modified the dataloader to give triplets (text, graph, label) where $label = 1$ if the text and the graph are related to the same molecule, and $label = 0$ otherwise. In our training, we chose to have the same proportion (50%) of triplets with $label = 1$ and $label = 0$.
- With this loss, the model learns to differentiate matching pairs and non-matching pairs.
- Encourages the model to consider information from both text and graph.
- We used a trainable temperature scaling parameter.

ALTeGraD 2023 Data Challenge

AR,REM,EB

Introduction

Model

Text encoder

Graph encoder

Loss Functions

Training and computational resources

Results

Best Model

Results obtained

Comments

Conclusion

Cosine Embedding Loss

The Cosine Embedding Loss is defined as follows :

$$\text{CEL}(v_1, v_2, \text{label}, \text{margin}) = \begin{cases} 1 - \cos(v_1, v_2) & \text{if } \textit{label} = 1 \\ \max(0, \cos(v_1, v_2) - \text{margin}) & \text{if } \textit{label} = -1 \end{cases}$$

- We used the same dataloader as before

ALTeGraD 2023 Data Challenge

AR,REM,EB

Introduction

Model

Text encoder

Graph encoder

Loss Functions

Training and computational resources

Results

Best Model

Results obtained

Comments

Conclusion

Triplet Loss

The triplet loss is defined as follows :

$$\text{TripletL}(v, v_p, v_n, \text{margin}) = max(d(v, v_p) - d(v, v_n) + \text{margin}, 0)$$

In this loss we consider three embeddings :

- Embedding v associated to the graph of a certain molecule.
- Embedding v_p (called positive) linked to a text describing the previous molecule.
- Embedding v_n (called negative) linked to a text describing another molecule.

A custom Dataloader was created for this loss.

ALTeGraD 2023 Data Challenge

AR,REM,EB

Introduction

Model

Text encoder

Graph encoder

Loss Functions

Training and computational resources

Results

Best Model

Results obtained

Comments

Conclusion

Few words on computational resources :

- We were working with Google Colab and Kaggle.
- We used a batch size of 32 (couldn't increase it due to GPU limitations).
- Training time was between 6 and 30 hours (depending on architecture and number of epochs).

Training methods :

- Better results when we don't freeze any layers in training (we tried to freeze layers of the pre-trained text encoder).
- Best models obtained were trained on approximately 100 epochs.
- Learning rate was set to $2 \cdot 10^{-5}$ for the first epochs and gradually reduced to $1 \cdot 10^{-7}$.

ALTeGraD 2023 Data Challenge

AR,REM,EB

Introduction

Model

Text encoder

Graph encoder

Loss Functions

Training and computational resources

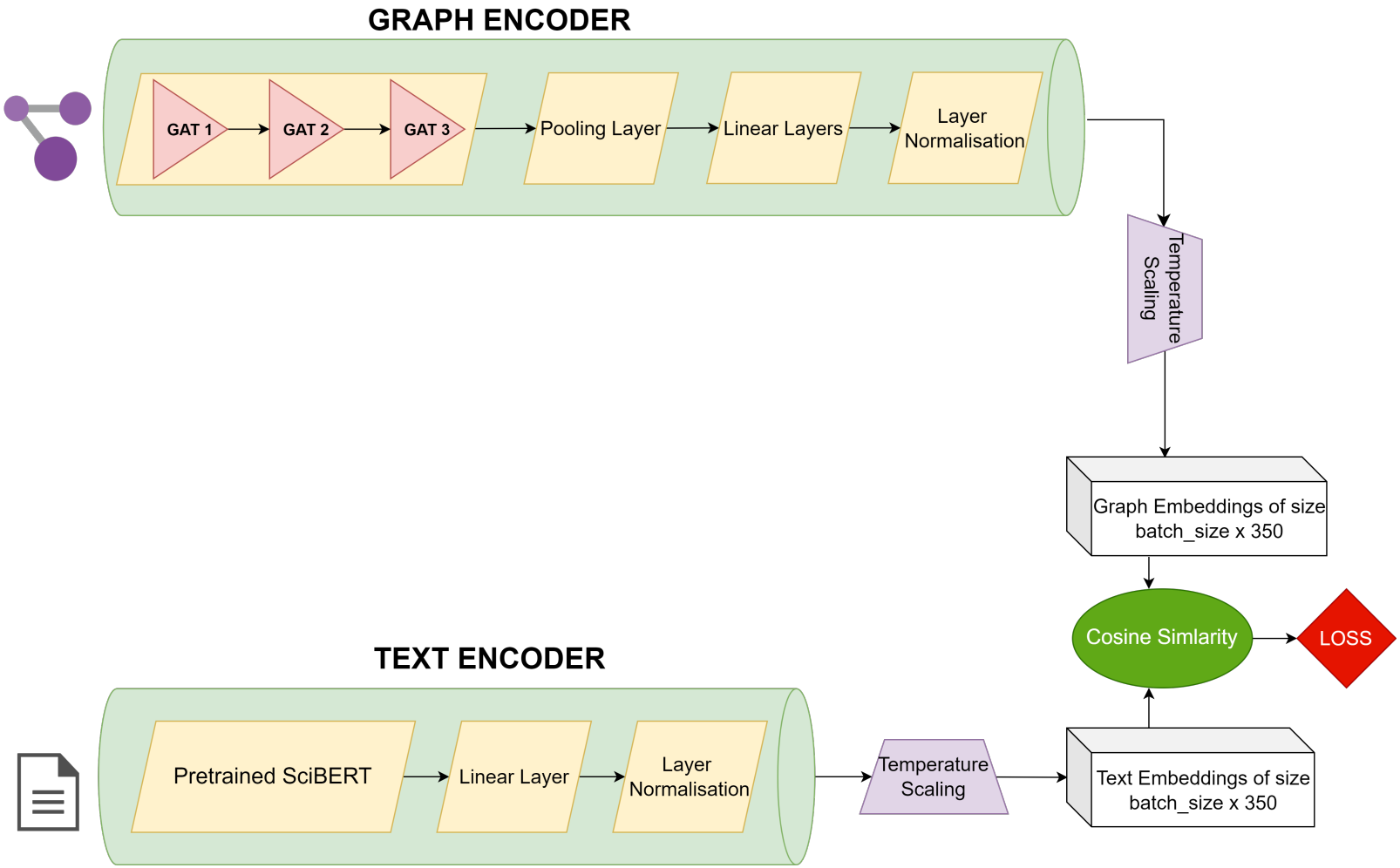
Results

Best Model

Results obtained

Comments

Conclusion



ALTeGraD 2023 Data Challenge

AR,REM,EB

Introduction

Model

Text encoder

Graph encoder

Loss Functions

Training and computational resources

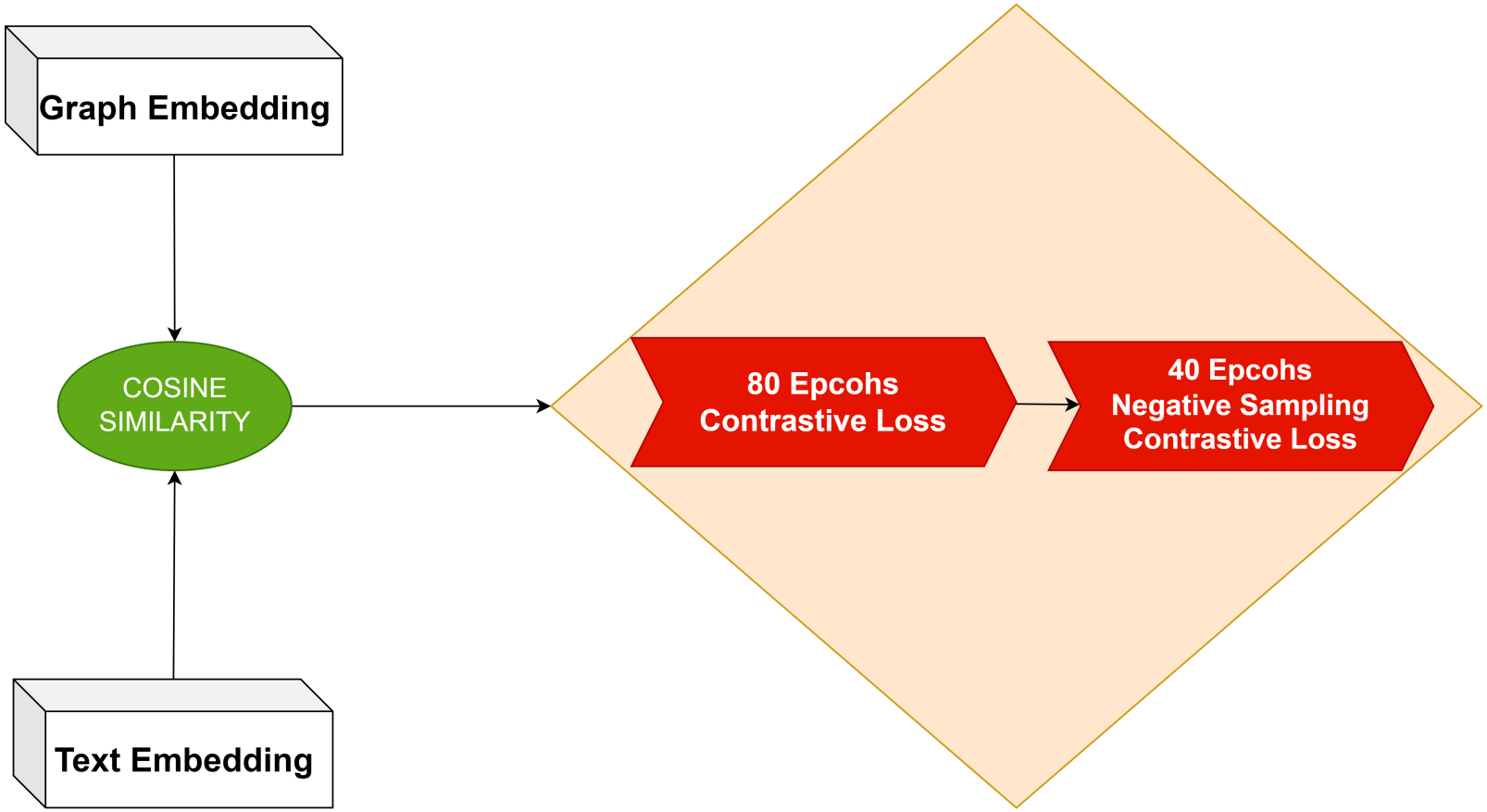
Results

Best Model

Results obtained

Comments

Conclusion



ALTeGraD 2023 Data Challenge

AR,REM,EB

Introduction

Model

Text encoder

Graph encoder

Loss Functions

Training and computational resources

Results

Best Model

Results obtained

Comments

Conclusion

Model Architecture				Loss				Train Setting		Result
Graph Enc.	Text Enc.	Out. Dim.	Temp. Scaling	Ctr. Loss	Neg. Sam.	Triplet Loss	Cos. Emb.	Epochs	Freeze Layers	Score
3 GAT	SciBERT	350	✓	✓	✓	X	X	120	X	0.87
3 GCN	SciBERT	350	✓	✓	✓	X	X	120	X	0.82
3 GAT	SciBERT	768	✓	X	✓	X	X	100	X	0.81
3 GAT	SciBERT	768	X	X	✓	X	X	60	X	0.79
5 GAT	SciBERT	768	X	X	✓	X	X	60	X	0.76
GCN+GAT+GIN	SciBERT	768	X	X	✓	X	X	60	X	0.74
3 GAT + 2 GCN	SciBERT	768	X	X	✓	X	X	50	X	0.71
3 GCN	SciBERT	768	X	✓	X	X	X	10	X	0.52
3 GAT	SciBERT	768	X	X	X	✓	X	50	X	0.46
3 GAT	SciBERT	768	X	X	X	X	✓	60	X	0.45
3 GCN	DistilBERT	768	X	✓	X	X	X	10	X	0.43
3 GCN	DistilBERT	768	X	✓	X	X	X	10	✓	0.4
3 GCN	SBERT	768	X	✓	X	X	X	20	X	0.38
3 GCN	SBERT	768	X	✓	X	X	X	20	✓	0.35

ALTeGraD 2023 Data Challenge

AR,REM,EB

Introduction

Model

Text encoder

Graph encoder

Loss Functions

Training and computational resources

Results

Best Model

Results obtained

Comments

Conclusion

What worked ?

1. Reducing the output dimension
2. Using contrastive loss then negative contrastive sampling loss
3. Use a temperature scaling parameter in the loss
4. Training on a large number of epochs
5. Using SciBERT as a text encoder
6. Using three GAT layers as a graph encoder
7. Changing the dataloader (generate in the same proportion positive and negative pairs)

What did not work ?

1. Using Triplet Loss (may require fine-tuning or selecting pairs)
2. Using too much (> 3) layers for graph encoder (didn't enhance performance and increase training time)
3. Using other graph encoders (GCN and GIN)

ALTeGraD 2023 Data Challenge

AR,REM,EB

Introduction

Model

Text encoder

Graph encoder

Loss Functions

Training and computational resources

Results

Best Model

Results obtained

Comments

Conclusion

- Co-training framework : Integrating SciBERT with a graph attention network for molecule retrieval from textual queries.
- Key observation : Choice of loss function crucial for results (Achieved 0.878 score).

Future avenues :

- Ensemble methods for model predictions.
- Experimentation with various loss functions.
- Parameter finetuning (e.g., scheduler, Triplet Loss margin).
- Initial use of Negative Sampling Contrastive Loss followed by Triplet Loss, or other combinaison of losses.