SimCSE: Simple Contrastive Learning of Sentence Embeddings

Mohamad Arvin Fadriansyah
Faculty of Computer Science
Universitas Indonesia
Depok, Indonesia
Email: mohamad.arvin@ui.ac.id

***Abstract*—This paper presents SimCSE, a simple contrastive learning framework that greatly advances the state-of-the-art sentence embeddings. We first describe an unsupervised approach, which takes an input sentence and predicts *itself* in a contrastive objective, with only standard dropout used as noise. This simple method works surprisingly well, performing on par with previous supervised counterparts. We find that dropout acts as minimal data augmentation and removing it leads to a representation collapse. Then, we propose a supervised approach, which incorporates annotated pairs from natural language inference datasets into our contrastive learning framework, by using "entailment" pairs as positives and "contradiction" pairs as hard negatives. We evaluate SimCSE on standard semantic textual similarity (STS) tasks, and our unsupervised and supervised models using $\text{BERT}_{\text{base}}$ achieve an average of 76.3% and 81.6% Spearman's correlation respectively, a 4.2% and 2.2% improvement compared to previous best results. We also show—both theoretically and empirically—that contrastive learning objective regularizes pre-trained embeddings' anisotropic space to be more uniform, and it better aligns positive pairs when supervised signals are available.**

I. Proses Menjalankan Kode

A. Inisialiasi Pytroch dan Parameter training

```
(myenv) C:\Users\moham\OneDrive\Desktop\NLP\
    SimCSE>python train.py --
    model_name_or_path bert-base-uncased --
    train_file data/wiki1m_for_simcse.txt --
    output_dir result/my-unsup-simcse-bert-
    base-uncased --num_train_epochs 1 --
    per_device_train_batch_size 64 --
    learning_rate 3e-5 --max_seq_length 32 --
    evaluation_strategy steps --
    metric_for_best_model stsb_spearman --
    load_best_model_at_end --eval_steps 125 --
    pooler_type cls --mlp_only_train --
    overwrite_output_dir --temp 0.05 --
    do_train --do_eval --fp16
```

- `--model_name_or_path bert-base-uncased`: Bagian ini menentukan model dasar yang digunakan untuk latihan, yaitu `bert-base-uncased`.
- `--train_file data/wiki1m_for_simcse.txt`: Dataset yang digunakan untuk training model, berupa file teks `wiki1m_for_simcse.txt`.
- `--output_dir result/my-unsup-simcse-bert-base-uncased`: Direktori untuk menyimpan model hasil latihan, yaitu `result/my-unsup-simcse-bert-base-uncased`.
- `--num_train_epochs 1`: Jumlah epoch (siklus pelatihan) yang diatur menjadi 1.
- `--per_device_train_batch_size 64`: Ukuran batch per perangkat (misalnya per GPU) yang diatur menjadi 64.
- `--learning_rate 3e-5`: Laju pembelajaran yang ditetapkan menjadi 3×10^{-5} atau 0.00003.
- `--max_seq_length 32`: Panjang maksimum setiap input teks adalah 32 token.
- `--evaluation_strategy steps`: Frekuensi evaluasi dilakukan berdasarkan jumlah langkah (steps) pelatihan.
- `--metric_for_best_model stsb_spearman`: Metrik yang digunakan untuk memilih model terbaik adalah `stsb_spearman` (korelasi Spearman pada STS-Benchmark).
- `--load_best_model_at_end`: Memuat model terbaik di akhir pelatihan berdasarkan metrik yang dipilih.
- `--eval_steps 125`: Evaluasi dilakukan setiap 125 langkah pelatihan.
- `--pooler_type cls`: Menggunakan representasi dari token `[CLS]` (class) sebagai representasi untuk setiap input teks.
- `--mlp_only_train`: Mengaktifkan lapisan MLP (Multi-Layer Perceptron) hanya saat pelatihan, untuk meningkatkan efisiensi.
- `--overwrite_output_dir`: Mengizinkan direktori keluaran (`output_dir`) ditimpa.
- `--temp 0.05`: Menetapkan suhu (temperature) menjadi 0.05 untuk loss function, suhu 0.05 termasuk rendah sehingga model dapat memberikan bobot yang tinggi terhadap pasangan kalimat yang mirip dan bobot rendah pada pasangan kalimat yang kurang mirip.
- `--do_train`: Menandakan bahwa perintah ini akan menjalankan proses pelatihan.
- `--do_eval`: Menandakan bahwa perintah ini akan menjalankan proses evaluasi setelah pelatihan selesai.
- `--fp16`: merujuk pada penggunaan presisi floating point 16-bit (dikenal juga sebagai half precision) selama pelatihan. Ini adalah teknik yang digunakan dalam deep learning untuk mempercepat pelatihan dan mengurangi

penggunaan memori tanpa mengorbankan akurasi model secara signifikan.

B. Device Setup

```
11/07/2024 20:35:29 - INFO - __main__ -
    PyTorch: setting up devices
11/07/2024 20:35:30 - WARNING - __main__ -
    Process rank: -1, device: cuda:0, n_gpu: 1
     distributed training: False, 16-bits
    training: True
11/07/2024 20:35:30 - INFO - __main__ -
    Training/evaluation parameters
    OurTrainingArguments(output_dir='result/my
    -unsup-simcse-bert-base-uncased',
    overwrite_output_dir=True, do_train=True,
    do_eval=True, do_predict=False,
    evaluation_strategy=<EvaluationStrategy.
    STEPS: 'steps'>, prediction_loss_only=
    False, per_device_train_batch_size=64,
    per_device_eval_batch_size=8,
    per_gpu_train_batch_size=None,
    per_gpu_eval_batch_size=None,
    gradient_accumulation_steps=1,
    eval_accumulation_steps=None,
    learning_rate=3e-05, weight_decay=0.0,
    adam_beta1=0.9, adam_beta2=0.999,
    adam_epsilon=1e-08, max_grad_norm=1.0,
    num_train_epochs=1.0, max_steps=-1,
    lr_scheduler_type=<SchedulerType.LINEAR: '
    linear'>, warmup_steps=0, logging_dir='
    runs\\Nov07_20-35-29_LAPTOP-EKA500FC',
    logging_first_step=False, logging_steps
    =500, save_steps=500, save_total_limit=
    None, no_cuda=False, seed=42, fp16=True,
    fp16_opt_level='O1', fp16_backend='auto',
    local_rank=-1, tpu_num_cores=None,
    tpu_metrics_debug=False, debug=False,
    dataloader_drop_last=False, eval_steps
    =125, dataloader_num_workers=0, past_index
    =-1, run_name='result/my-unsup-simcse-bert
    -base-uncased', disable_tqdm=False,
    remove_unused_columns=True, label_names=
    None, load_best_model_at_end=True,
    metric_for_best_model='stsb_spearman',
    greater_is_better=True, ignore_data_skip=
    False, sharded_ddp=False, deepspeed=None,
    label_smoothing_factor=0.0, adafactor=
    False, eval_transfer=False)
```

- `cuda:0` menunjukkan bahwa model akan dilatih pada perangkat GPU dengan `device 0`.
- `n_gpu: 1` menunjukkan bahwa terdapat 1 GPU yang tersedia untuk proses pelatihan.
- `distributed training: False` menunjukkan bahwa pelatihan ini tidak menggunakan pelatihan terdistribusi atau multi-GPU.
- `16-bits training: True` artinya pelatihan dilakukan dengan presisi 16-bit, atau *mixed precision training*, yang dapat mempercepat pelatihan sekaligus mengurangi penggunaan memori.

C. Argumen pada Training

Berikut adalah beberapa parameter penting dalam proses pelatihan model:

- `output_dir='result/my-unsup-simcse-bert-base-uncased'`
  Direktori tempat menyimpan hasil pelatihan, model, dan output lainnya.
- `overwrite_output_dir=True`
  Jika diaktifkan, direktori output akan ditimpa dengan hasil baru jika sudah ada.
- `do_train=True`
  Menunjukkan bahwa proses pelatihan (*training*) akan dilakukan.
- `do_eval=True`
  Menunjukkan bahwa evaluasi model (*evaluation*) akan dilakukan setelah pelatihan.
- `do_predict=False`
  Menunjukkan bahwa proses prediksi pada data uji tidak akan dilakukan.
- `evaluation_strategy=steps`
  Evaluasi model akan dilakukan pada setiap beberapa *steps*, bukan di setiap epoch.
- `per_device_train_batch_size=64`
  Batch size per perangkat (GPU) selama pelatihan adalah 64.
- `per_device_eval_batch_size=8`
  Batch size per perangkat selama evaluasi adalah 8.
- `gradient_accumulation_steps=1`
  Mengindikasikan jumlah langkah *gradient accumulation* yang dilakukan sebelum melakukan update parameter. Dalam hal ini, model akan melakukan update parameter setiap 1 langkah.
- `eval_accumulation_steps=None`
  Parameter ini akan mengatur jumlah langkah akumulasi pada evaluasi, yang tidak digunakan (`None`).
- `learning_rate=3e-05`
  Laju pembelajaran ditetapkan pada 3×10^{-5} atau 0.00003.
- `weight_decay=0.0`
  Tidak ada *weight decay* yang diterapkan pada model (nilai 0.0).
- `adam_beta1=0.9` dan `adam_beta2=0.999`
  Parameter β_1 dan β_2 untuk algoritma optimisasi Adam ditetapkan masing-masing 0.9 dan 0.999.
- `adam_epsilon=1e-08`
  Parameter epsilon untuk Adam optimizer adalah 1×10^{-8}.
- `max_grad_norm=1.0`
  Nilai maksimum untuk norma gradien, yang digunakan untuk menghindari *gradient exploding*, adalah 1.0.
- `num_train_epochs=1.0`
  Jumlah *epochs* pelatihan yang ditentukan adalah 1.
- `max_steps=-1`
  Parameter ini menunjukkan jumlah langkah pelatihan yang harus dilakukan. Jika negatif, maka pelatihan dilakukan berdasarkan jumlah epoch.
- `lr_scheduler_type='linear'`
  Scheduler untuk laju pembelajaran adalah tipe linear, yang secara linier menurunkan laju pembelajaran selama pelatihan.
- `warmup_steps=0`

Jumlah langkah *warmup* untuk scheduler laju pembelajaran adalah 0.

- `logging_dir='runs/Nov07_20-35-29_LAPTOP-EKA500FC'`
  Direktori tempat hasil log dari pelatihan disimpan.
- `logging_steps=500`
  Menentukan bahwa logging akan dilakukan setiap 500 langkah.
- `save_steps=500`
  Model akan disimpan setiap 500 langkah.
- `save_total_limit=None`
  Tidak ada batasan pada jumlah model yang disimpan.
- `no_cuda=False`
  Mengindikasikan bahwa *CUDA* diaktifkan sehingga pelatihan menggunakan GPU.
- `seed=42`
  Nilai acak (*random seed*) untuk reproduktibilitas hasil adalah 42.
- `fp16=True`
  Mengindikasikan pelatihan menggunakan presisi 16-bit (*mixed precision*).
- `fp16_opt_level='O1'`
  Menentukan tingkat pengoptimalan presisi 16-bit, yang dalam hal ini menggunakan tingkat optimasi O1.
- `fp16_backend='auto'`
  Backend untuk 16-bit presisi akan dipilih secara otomatis.
- `local_rank=-1`
  Indeks untuk pelatihan terdistribusi. Nilai -1 berarti pelatihan tunggal, bukan terdistribusi.
- `tpu_num_cores=None`
  Tidak ada TPU yang digunakan (nilai `None`).
- `debug=False`
  Menunjukkan bahwa mode debug tidak diaktifkan.
- `dataloader_drop_last=False`
  Tidak menjatuhkan batch terakhir yang lebih kecil dari ukuran batch saat pelatihan atau evaluasi.
- `eval_steps=125`
  Evaluasi akan dilakukan setiap 125 langkah.
- `dataloader_num_workers=0`
  Jumlah pekerja untuk proses data pada setiap batch adalah 0 (tidak ada pekerja paralel untuk data loading).
- `past_index=-1`
  Indeks parameter ”past” untuk fungsi cache model.
- `run_name='result/my-unsup-simcse-bert-base-uncased'`
  Nama run ini untuk mencatat hasil pelatihan.
- `disable_tqdm=False`
  Menunjukkan bahwa progress bar (tqdm) diaktifkan.
- `remove_unused_columns=True`
  Kolom yang tidak terpakai pada input akan dihapus untuk efisiensi.
- `label_names=None`
  Nama kolom label dalam dataset, `None` menunjukkan tidak ada label khusus yang digunakan.
- `load_best_model_at_end=True`
  Mengindikasikan bahwa model terbaik akan dimuat setelah pelatihan selesai, berdasarkan metrik evaluasi.
- `metric_for_best_model='stsb_spearman'`
  Metrik korelasi Spearman pada dataset STS-Benchmark yang digunakan untuk memilih model terbaik.
- `greater_is_better=True`
  Nilai yang lebih tinggi pada metrik evaluasi menunjukkan model yang lebih baik.
- `ignore_data_skip=False`
  Data yang dilewati tidak akan diabaikan selama pelatihan.
- `sharded_ddp=False`
  Menunjukkan bahwa *Sharded Data Parallelism* tidak diaktifkan.
- `deepspeed=None`
  Konfigurasi Deepspeed tidak diaktifkan.
- `label_smoothing_factor=0.0`
  Tidak ada *label smoothing* yang diterapkan.
- `adafactor=False`
  Algoritma Adafactor tidak digunakan.
- `eval_transfer=False`
  Tidak dilakukan evaluasi pada transfer learning.

D. Model Configuration:

```
[INFO|configuration_utils.py:445] 2024-11-07
    20:35:31,583 >> loading configuration file
     https://huggingface.co/bert-base-uncased/
    resolve/main/config.json from cache at C:\
    Users\moham/.cache\huggingface\
    transformers\3
    c61d016573b14f7f008c02c4e51a366c67ab274726fe2910691e
    .37395
    cee442ab11005bcd270f3c34464dc1704b715b5d7d52b1a461ab

[INFO|configuration_utils.py:481] 2024-11-07
    20:35:31,584 >> Model config BertConfig {
  "architectures": [
    "BertForMaskedLM"
  ],
  "attention_probs_dropout_prob": 0.1,
  "gradient_checkpointing": false,
  "hidden_act": "gelu",
  "hidden_dropout_prob": 0.1,
  "hidden_size": 768,
  "initializer_range": 0.02,
  "intermediate_size": 3072,
  "layer_norm_eps": 1e-12,
  "max_position_embeddings": 512,
  "model_type": "bert",
  "num_attention_heads": 12,
  "num_hidden_layers": 12,
  "pad_token_id": 0,
  "position_embedding_type": "absolute",
  "transformers_version": "4.2.1",
  "type_vocab_size": 2,
  "use_cache": true,
  "vocab_size": 30522
}
```

konfigurasi dari model `bert-base-uncased` adalah sebagai berikut,

- `hidden_size: 768` adalah ukuran tiap hidden layer pada model.

- num_hidden_layers: 12 menunjukkan total hidden layer pada model.
- "attention_probs_dropout_prob": 0.1
  Nilai dropout untuk probabilitas dalam *attention layer*, yaitu 0.1 atau 10%. Dropout digunakan untuk mencegah *overfitting*.
- "gradient_checkpointing": false
  Menyatakan bahwa *gradient checkpointing* tidak diaktifkan. *Gradient checkpointing* dapat mengurangi penggunaan memori dengan mengorbankan sedikit performa pelatihan.
- "hidden_act": "gelu"
  Fungsi aktivasi pada lapisan tersembunyi adalah *Gaussian Error Linear Unit* (GELU), fungsi non-linear yang umum pada BERT.
- "hidden_dropout_prob": 0.1
  Nilai dropout pada lapisan tersembunyi (hidden layer) sebesar 0.1 atau 10%.
- "initializer_range": 0.02
  Rentang awal untuk inisialisasi bobot, yaitu 0.02.
- "intermediate_size": 3072
  Ukuran lapisan *intermediate* dalam *feed-forward network* pada setiap blok *transformer*, yaitu 3072.
- "layer_norm_eps": 1e-12
  Parameter epsilon pada layer normalisasi, digunakan untuk stabilitas numerik saat menghitung layer normalisasi.
- "max_position_embeddings": 512
  Jumlah maksimum posisi yang dapat diwakili oleh *position embeddings*, yaitu 512, yang berarti model dapat menangani urutan hingga 512 token.
- "model_type": "bert"
  Jenis model adalah BERT, atau *Bidirectional Encoder Representations from Transformers*.
- "num_attention_heads": 12
  Jumlah *attention heads* dalam setiap lapisan *transformer*, yaitu 12.
- "num_hidden_layers": 12
  Jumlah lapisan tersembunyi dalam model, yaitu 12, yang menunjukkan kedalaman model.
- "pad_token_id": 0
  ID token yang digunakan untuk *padding*, yaitu 0. Token ini digunakan untuk melengkapi input agar memiliki panjang yang sama.
- "position_embedding_type": "absolute"
  Tipe *position embedding* yang digunakan adalah absolut, di mana setiap posisi dihitung secara absolut.
- "transformers_version": "4.2.1"
  Versi pustaka *Transformers* yang digunakan, yaitu 4.2.1.
- "type_vocab_size": 2
  Ukuran kosakata tipe (type vocabulary), yaitu 2, digunakan untuk menangani dua jenis segmen teks (contohnya dalam tugas klasifikasi dua kalimat).
- "use_cache": true
  Parameter ini menentukan apakah cache aktivasi akan digunakan selama inferensi, untuk mempercepat proses.
- "vocab_size": 30522
  Ukuran kosakata yang digunakan model, yaitu 30522, yang menunjukkan jumlah token unik yang dapat ditangani oleh model.

E. Inisialiasi dan Tipe Model

Arsitektur model dan inialisasi dideskripsikan sebagaai berikut,

- Arsitektur model menggunakan BertForMaskedLM, yang mana di desain pada tugas masked language modelling.
- Versi dari library transformses yang digunakan adalah versi 4.2.1.

F. Proses pelatihan

```
[INFO|tokenization_utils_base.py:1766]
    2024-11-07 20:35:33,472 >> loading file
    https://huggingface.co/bert-base-uncased/
    resolve/main/vocab.txt from cache at C:\
    Users\moham/.cache\huggingface\
    transformers\45
    c3f7a79a80e1cf0a489e5c62b43f173c15db47864303a55d623b
    .
    d789d64ebfe299b0e416afc4a169632f903f693095b4629a7ea2

[INFO|tokenization_utils_base.py:1766]
    2024-11-07 20:35:33,472 >> loading file
    https://huggingface.co/bert-base-uncased/
    resolve/main/tokenizer.json from cache at
    C:\Users\moham/.cache\huggingface\
    transformers\534479488
    c54aeaf9c3406f647aa2ec13648c06771ffe269edabebd4c412d
    .7
    f2721073f19841be16f41b0a70b600ca6b880c8f3df6f3535cbc

[INFO|modeling_utils.py:1027] 2024-11-07
    20:35:33,834 >> loading weights file https
    ://huggingface.co/bert-base-uncased/
    resolve/main/pytorch_model.bin from cache
    at C:\Users\moham/.cache\huggingface\
    transformers\
    a8041bf617d7f94ea26d15e218abd04afc2004805632abc0ed20
    .
    faf6ea826ae9c5867d12b22257f9877e6b8367890837bd60f7c5

[WARNING|modeling_utils.py:1134] 2024-11-07
    20:35:37,889 >> Some weights of the model
    checkpoint at bert-base-uncased were not
    used when initializing BertForCL: ['cls.
    predictions.bias', 'cls.predictions.
    transform.dense.weight', 'cls.predictions.
    transform.dense.bias', 'cls.predictions.
    decoder.weight', 'cls.seq_relationship.
    weight', 'cls.seq_relationship.bias', 'cls
    .predictions.transform.LayerNorm.weight',
    'cls.predictions.transform.LayerNorm.bias
    ', 'bert.pooler.dense.weight', 'bert.
    pooler.dense.bias']
- This IS expected if you are initializing
    BertForCL from the checkpoint of a model
    trained on another task or with another
    architecture (e.g. initializing a
    BertForSequenceClassification model from a
     BertForPreTraining model).
```

```
- This IS NOT expected if you are initializing
     BertForCL from the checkpoint of a model
    that you expect to be exactly identical (
    initializing a
    BertForSequenceClassification model from a
     BertForSequenceClassification model).
[WARNING|modeling_utils.py:1145] 2024-11-07
    20:35:37,890 >> Some weights of BertForCL
    were not initialized from the model
    checkpoint at bert-base-uncased and are
    newly initialized: ['mlp.dense.weight', '
    mlp.dense.bias']
You should probably TRAIN this model on a down
    -stream task to be able to use it for
    predictions and inference.
[INFO|trainer.py:441] 2024-11-07 20:35:56,548
    >> The following columns in the training
    set don't have a corresponding argument in
     `BertForCL.forward` and have been ignored
    : .
[INFO|trainer.py:358] 2024-11-07 20:35:56,549
    >> Using amp fp16 backend
11/07/2024 20:35:56 - INFO - simcse.trainers -
     ***** Running training *****
11/07/2024 20:35:56 - INFO - simcse.trainers -
     Num examples = 1000000
11/07/2024 20:35:56 - INFO - simcse.trainers -
     Num Epochs = 1
11/07/2024 20:35:56 - INFO - simcse.trainers -
     Instantaneous batch size per device = 64
11/07/2024 20:35:56 - INFO - simcse.trainers -
     Total train batch size (w. parallel,
    distributed & accumulation) = 64
11/07/2024 20:35:56 - INFO - simcse.trainers -
     Gradient Accumulation steps = 1
11/07/2024 20:35:56 - INFO - simcse.trainers -
     Total optimization steps = 15625
```

- **Tokenizer and Model Loading**
  - `loading file` log entries menunjukkan bawa vocabulary dan JSON configuration berasal dari cached files pada sistem.
  - weights dari model diambil dari cache untuk inisialiasi pada model BertForCL (suatu model kustom pada contrastive learning).
- **Warning on Unused Weights**
  - Beberapa weights (`'cls.predictions.bias'`, etc.) tidak digunakan pada `BertForCL`. Hal ini dikarenakan `BertForCL` hanya menggunakan lapisan yang relevan untuk contrastive learning dan mengabaikan lapisan yang digunakan untuk melakukan prediksi pada model pre-training.
  - `'mlp.dense.weight'` dan `'mlp.dense.bias'` weights diinisialiasi tetapi tidak ada di pre-trained `bert-base-uncased` model. proses training berguna untuk melatih parameter ini terhadap target task.
- **Trainer Setup and Training Initialization**
  - `Using amp fp16 backend` mengindikasikan bahwa Automatic Mixed Precision (AMP) training diaktifkan untuk presisi 16-bit yang bertujuan untuk meningkatkan efisiendi komputasi.
  - `Num examples = 1000000`: Terdapat 1,000,000 contoh training.
  - `Num Epochs = 1`: Training terjadi selama 1 epoch.
  - `Instantaneous batch size per device = 64`: tiap proses device (GPU) merupakan suatu batch yang terdiri dari 64 sampel.
  - `Total train batch size` terdapat 64 batch size dalam single training.
  - `Gradient Accumulation steps = 1`: tidak terdapat akumulasi gradient; gradien selalu di update pada tiap batch.
  - `Total optimization steps = 15625`: total optimization steps didapat dari rumus sebagai berikut,

$$\text{Total Optimization Steps} = \frac{\text{Jumlah Sampel} \times \text{Epochs}}{\text{Batch Size}}$$

Pada kasus ini terdapat 1,000,000 sampel, dengan batch size sebanyak 64 dan 1 epoch:

$$\text{Total Optimization Steps} = \frac{1,000,000 \times 1}{64} = 15,625$$

Tiap tiap stepnya, terdapat perhitungan gradien dan proses mengupdate parameter dari model, yang mana hal ini dapat meningkatkan peforma dari model.

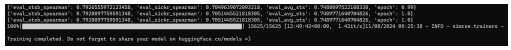

Fig. 1. Pelatihan telah selesai

```
{'loss': 0.0001, 'learning_rate':
    2.4000000000000003e-07, 'epoch': 0.99}
{'eval_stsb_spearman': 0.7926555972123458,
     'eval_sickr_spearman':
    0.7049639072093218, 'eval_avg_sts':
    0.7488097522108338, 'epoch': 0.99}
{'eval_stsb_spearman': 0.7928097759591348,
     'eval_sickr_spearman':
    0.7051445521818305, 'eval_avg_sts':
    0.7489771640704826, 'epoch': 1.0}
{'eval_stsb_spearman': 0.7928097759591348,
     'eval_sickr_spearman':
    0.7051445521818305, 'eval_avg_sts':
    0.7489771640704826, 'epoch': 1.0}
100%|| 15625/15625 [12:49:42<00:00, 1.42it
    /s]11/08/2024 09:25:38 - INFO - simcse.
    trainers -
Training completed. Do not forget to share
     your model on huggingface.co/models =)
```

G. Hasil pelatihan

- **eval_sickr_spearman**:
  - Metrik ini mengukur nilai korelasi Spearman antara prediksi model dan nilai label pada dataset SICK-R (Sentences Involving Compositional Knowledge - Relatedness).

  - SICK-R adalah dataset lain yang berfokus pada pengukuran kesamaan atau keterkaitan antar kalimat dengan anotasi semantik.
  - Nilai yang lebih tinggi menunjukkan bahwa model semakin mampu menangkap makna yang setara atau mirip antar pasangan kalimat, seperti yang disepakati oleh anotasi manusia dalam dataset SICK-R.

$$\text{eval\_sickr\_spearman} = 0.7051$$

- **eval_avg_sts**:
  - Ini adalah rata-rata dari `eval_stsb_spearman` dan `eval_sickr_spearman`.
  - Rata-rata ini memberikan gambaran umum dari performa model dalam menangkap kesamaan semantik pada dua dataset yang berbeda, STS-B dan SICK-R.
  - Nilai yang lebih tinggi pada metrik ini menunjukkan bahwa model konsisten dalam kinerjanya pada kedua dataset.

$$\text{eval\_avg\_sts} = 0.7490$$

- **loss**:
  - Nilai loss menunjukkan seberapa besar kesalahan prediksi model selama proses pelatihan. Semakin rendah nilainya, semakin baik model dalam menyesuaikan prediksinya dengan nilai sebenarnya.
  - Nilai ini dihitung berdasarkan fungsi loss tertentu yang dirancang untuk task kesamaan semantik, seperti kontrastif atau cross-entropy loss.
  - Pada akhir pelatihan, loss mencapai nilai sangat rendah, yaitu:

$$\text{loss} = 0.0001$$

- **learning_rate**:
  - Ini adalah nilai laju pembelajaran, yang mengontrol seberapa besar pembaruan parameter model pada setiap iterasi pelatihan.
  - Nilai ini biasanya menurun secara bertahap selama pelatihan untuk menghindari perubahan besar pada parameter ketika model mendekati konvergensi.
  - Pada akhir pelatihan, nilai learning rate sangat kecil:

$$\text{learning\_rate} = 2.4 \times 10^{-7}$$

- **epoch**:
  - Epoch menunjukkan jumlah siklus pelatihan yang telah diselesaikan pada seluruh dataset. Dalam hasil ini, model dilatih hingga mencapai epoch 1.0, yang berarti dataset telah dilalui satu kali penuh dalam pelatihan.
- **Model Loading**

```
[INFO|modeling_utils.py] loading weights
    file result/my-unsup-simcse-bert-base-
    uncased\pytorch_model.bin
[INFO|modeling_utils.py] All model
    checkpoint weights were used when
    initializing BertForCL.
[INFO|modeling_utils.py] All the weights
    of BertForCL were initialized from the
    model checkpoint at result/my-unsup-
    simcse-bert-base-uncased.
```

  - Bagian ini menunjukkan bahwa bobot model BERT berhasil dimuat dari checkpoint yang disimpan sebelumnya. Model (BertForCL) kini siap digunakan untuk prediksi.
- **Model Saving**

```
[INFO|trainer.py] Saving model checkpoint
    to result/my-unsup-simcse-bert-base-
    uncased
[INFO|configuration_utils.py]
    Configuration saved in result/my-unsup-
    simcse-bert-base-uncased\config.json
[INFO|modeling_utils.py] Model weights
    saved in result/my-unsup-simcse-bert-
    base-uncased\pytorch_model.bin
```

  - setelah training model akan disimpan disini
- **Training Metrics**

```
{'train_runtime': 46186.2844,
 'train_samples_per_second': 0.338,
 'epoch': 1.0}
```

  - Total runtime sekitar 46,186 detik. Tiap detiknya terdapat 0.338 training. dengan total epoch sebanyak 1

H. Evaluasi Model Training

```
{11/08/2024 09:25:44 - INFO - __main__ -
    *** Evaluate ***
11/08/2024 09:28:21 - INFO - root -
    Generating sentence embeddings
11/08/2024 09:30:07 - INFO - root -
    Generated sentence embeddings
11/08/2024 09:30:07 - INFO - root -
    Training pytorch-MLP-nhid0-rmsprop-
    bs128 with (inner) 5-fold cross-
    validation
11/08/2024 09:30:30 - INFO - root - Best
    param found at split 1: l2reg = 0.001
    with score 79.51
11/08/2024 09:30:53 - INFO - root - Best
    param found at split 2: l2reg = 0.001
    with score 79.93
11/08/2024 09:31:18 - INFO - root - Best
    param found at split 3: l2reg = 0.01
    with score 79.61
11/08/2024 09:31:40 - INFO - root - Best
    param found at split 4: l2reg = 0.01
    with score 79.51
11/08/2024 09:32:01 - INFO - root - Best
    param found at split 5: l2reg = 0.001
    with score 79.53
11/08/2024 09:32:02 - INFO - root -
    Generating sentence embeddings
11/08/2024 09:32:36 - INFO - root -
    Generated sentence embeddings
11/08/2024 09:32:36 - INFO - root -
    Training pytorch-MLP-nhid0-rmsprop-
    bs128 with (inner) 5-fold cross-
    validation
```

```
11/08/2024 09:32:45 - INFO - root - Best
    param found at split 1: l2reg = 0.0001
    with score 86.06
11/08/2024 09:32:54 - INFO - root - Best
    param found at split 2: l2reg = 0.01
    with score 85.93
11/08/2024 09:33:01 - INFO - root - Best
    param found at split 3: l2reg = 0.01
    with score 86.03
11/08/2024 09:33:10 - INFO - root - Best
    param found at split 4: l2reg = 0.001
    with score 85.53
11/08/2024 09:33:18 - INFO - root - Best
    param found at split 5: l2reg = 0.01
    with score 85.13
11/08/2024 09:33:18 - INFO - root -
    Generating sentence embeddings
11/08/2024 09:35:51 - INFO - root -
    Generated sentence embeddings
11/08/2024 09:35:51 - INFO - root -
    Training pytorch-MLP-nhid0-rmsprop-
    bs128 with (inner) 5-fold cross-
    validation
11/08/2024 09:36:15 - INFO - root - Best
    param found at split 1: l2reg = 0.01
    with score 94.29
11/08/2024 09:36:42 - INFO - root - Best
    param found at split 2: l2reg = 0.0001
    with score 94.52
11/08/2024 09:37:01 - INFO - root - Best
    param found at split 3: l2reg = 0.001
    with score 94.29
11/08/2024 09:37:24 - INFO - root - Best
    param found at split 4: l2reg = 0.001
    with score 94.56
11/08/2024 09:37:46 - INFO - root - Best
    param found at split 5: l2reg = 0.001
    with score 94.18
11/08/2024 09:37:47 - INFO - root -
    Generating sentence embeddings
11/08/2024 09:38:22 - INFO - root -
    Generated sentence embeddings
11/08/2024 09:38:22 - INFO - root -
    Training pytorch-MLP-nhid0-rmsprop-
    bs128 with (inner) 5-fold cross-
    validation
11/08/2024 09:38:46 - INFO - root - Best
    param found at split 1: l2reg = 0.0001
    with score 87.11
11/08/2024 09:39:07 - INFO - root - Best
    param found at split 2: l2reg = 0.0001
    with score 86.25
11/08/2024 09:39:30 - INFO - root - Best
    param found at split 3: l2reg = 0.001
    with score 87.67
11/08/2024 09:39:53 - INFO - root - Best
    param found at split 4: l2reg = 1e-05
    with score 87.12
11/08/2024 09:40:17 - INFO - root - Best
    param found at split 5: l2reg = 0.0001
    with score 86.42
11/08/2024 09:40:18 - INFO - root -
    Computing embedding for train
11/08/2024 09:48:12 - INFO - root -
    Computed train embeddings
11/08/2024 09:48:12 - INFO - root -
    Computing embedding for dev
11/08/2024 09:48:24 - INFO - root -
    Computed dev embeddings
11/08/2024 09:48:24 - INFO - root -
    Computing embedding for test
11/08/2024 09:48:49 - INFO - root -
    Computed test embeddings
11/08/2024 09:48:49 - INFO - root -
    Training pytorch-MLP-nhid0-rmsprop-
    bs128 with standard validation..
11/08/2024 09:49:51 - INFO - root - [('reg
    :1e-05', 85.09), ('reg:0.0001', 84.98),
     ('reg:0.001', 85.21), ('reg:0.01',
    85.21)]
11/08/2024 09:49:51 - INFO - root -
    Validation : best param found is reg =
    0.001 with score 85.21
11/08/2024 09:49:51 - INFO - root -
    Evaluating...
11/08/2024 09:50:02 - INFO - root - *****
    Transfer task : TREC *****


11/08/2024 09:51:00 - INFO - root -
    Computed train embeddings
11/08/2024 09:51:04 - INFO - root -
    Computed test embeddings
11/08/2024 09:51:04 - INFO - root -
    Training pytorch-MLP-nhid0-rmsprop-
    bs128 with 5-fold cross-validation
11/08/2024 09:51:18 - INFO - root - [('reg
    :1e-05', 81.51), ('reg:0.0001', 81.46),
     ('reg:0.001', 80.85), ('reg:0.01',
    77.16)]
11/08/2024 09:51:18 - INFO - root - Cross-
    validation : best param found is reg =
    1e-05 with score 81.51
11/08/2024 09:51:18 - INFO - root -
    Evaluating...
11/08/2024 09:51:19 - INFO - root - *****
    Transfer task : MRPC *****


11/08/2024 09:51:19 - INFO - root -
    Computing embedding for train
11/08/2024 09:54:05 - INFO - root -
    Computed train embeddings
11/08/2024 09:54:05 - INFO - root -
    Computing embedding for test
11/08/2024 09:55:15 - INFO - root -
    Computed test embeddings
11/08/2024 09:55:15 - INFO - root -
    Training pytorch-MLP-nhid0-rmsprop-
    bs128 with 5-fold cross-validation
11/08/2024 09:55:27 - INFO - root - [('reg
    :1e-05', 74.9), ('reg:0.0001', 74.75),
    ('reg:0.001', 74.41), ('reg:0.01',
    73.63)]
11/08/2024 09:55:27 - INFO - root - Cross-
    validation : best param found is reg =
    1e-05 with score 74.9
11/08/2024 09:55:27 - INFO - root -
    Evaluating...
11/08/2024 09:55:28 - INFO - __main__ -
    ***** Eval results *****
11/08/2024 09:55:28 - INFO - __main__ -
    epoch = 1.0
11/08/2024 09:55:28 - INFO - __main__ -
```

```
    eval_CR = 85.74
11/08/2024 09:55:28 - INFO - __main__ -
    eval_MPQA = 86.91
11/08/2024 09:55:28 - INFO - __main__ -
    eval_MR = 79.62
11/08/2024 09:55:28 - INFO - __main__ -
    eval_MRPC = 74.9
11/08/2024 09:55:28 - INFO - __main__ -
    eval_SST2 = 85.21
11/08/2024 09:55:28 - INFO - __main__ -
    eval_SUBJ = 94.37
11/08/2024 09:55:28 - INFO - __main__ -
    eval_TREC = 81.51
11/08/2024 09:55:28 - INFO - __main__ -
    eval_avg_sts = 0.7778725782061404
11/08/2024 09:55:28 - INFO - __main__ -
    eval_avg_transfer = 84.03714285714285
11/08/2024 09:55:28 - INFO - __main__ -
    eval_sickr_spearman =
    0.7441769787344114
11/08/2024 09:55:28 - INFO - __main__ -
    eval_stsb_spearman =
    0.8115681776778695}
```

- Model melakukan evaluasi di berbagai titik waktu dengan log menunjukkan bagian "Evaluating..." dan "Generating sentence embeddings". Embeddings adalah representasi vektor dari kalimat yang kemudian digunakan dalam transfer task untuk melihat seberapa baik model memahami semantik teks
- Pada setiap titik evaluasi, embeddings dihasilkan untuk set data pelatihan, validasi, atau tes.
- Setelah cross-validation, model melakukan pelatihan tambahan dengan validasi standar (non-cross-validation) menggunakan pengaturan regularisasi terbaik yang ditemukan.
- Setelah cross-validation, model melakukan pelatihan tambahan dengan validasi standar (non-cross-validation) menggunakan pengaturan regularisasi terbaik yang ditemukan.
- Log menampilkan hasil terbaik untuk tiap regularisasi, seperti `[('reg:1e-05', 85.09), ('reg:0.0001', 84.98), ('reg:0.001', 85.21), ('reg:0.01', 85.21)]` dan reg = 0.001 ditemukan sebagai parameter terbaik dengan skor 85.21.
- Model dievaluasi pada berbagai transfer task, yang menguji kinerja model dalam memahami semantik kalimat. Tugas-tugas ini meliputi, CR (Customer Reviews), MPQA (Opinion Polarity), MR (Movie Reviews), MRPC (Microsoft Paraphrase Corpus), SST2 (Sentiment Treebank), SUBJ (Subjectivity Classification), TREC (Question Classification
- Setiap tugas memiliki skor evaluasi yang menunjukkan kinerja model pada tugas tersebut. dengan skor sebagai berikut,

$$\begin{aligned}
\text{eval_CR} &= 85.74 \\
\text{eval_MPQA} &= 86.91 \\
\text{eval_MR} &= 79.62 \\
\text{eval_MRPC} &= 74.9 \\
\text{eval_SST2} &= 85.21 \\
\text{eval_SUBJ} &= 94.37 \\
\text{eval_TREC} &= 81.51 \\
\text{eval_avg_sts} &= 0.7778725782061404 \\
\text{eval_avg_transfer} &= 84.03714285714285 \\
\text{eval_sickr_spearman} &= 0.7441769787344114 \\
\text{eval_stsb_spearman} &= 0.8115681776778695
\end{aligned}$$

- eval_avg_transfer menunjukkan rata-rata performa pada semua transfer task
- Model juga dievaluasi menggunakan korelasi Spearman pada tugas semantic textual similarity (STS).
- eval_sickr_spearman dan eval_stsb_spearman menunjukkan nilai korelasi Spearman pada set data SICK-R dan STS-B, yaitu 0.744 dan 0.812, yang mencerminkan seberapa baik model memprediksi kesamaan semantik.

I. Evaluasi Model dengan Data Testing (lokal)

```
python evaluation.py --model_name_or_path
    result/my-unsup-simcse-bert-base-
    uncased --pooler cls --task_set sts --
    mode test
```

- `--model_name_or_path result/my-unsup-simcse-bert-base-uncased`
  Argumen ini menunjukkan path model SimCSE yang telah dilatih sebelumnya, yang disimpan di folder `result/my-unsup-simcse-bert-base-uncased`. Model ini adalah varian BERT (`bert-base-uncased`) yang dilatih menggunakan metode unsupervised SimCSE untuk menghasilkan embedding yang memiliki kemiripan semantik yang lebih baik.
- `--pooler cls`
  Argumen ini menentukan metode pooling untuk mendapatkan embedding dari model. `cls` berarti kita menggunakan token `[CLS]` (token pertama dalam setiap input) sebagai representasi embedding dari keseluruhan input teks.
- `--task_set sts`
  Argumen ini menunjukkan kumpulan tugas yang dievaluasi. `sts` berarti bahwa tugas evaluasi yang digunakan adalah **Semantic Textual Similarity (STS)**, yaitu tugas yang mengukur kemampuan model dalam menentukan kemiripan antara dua kalimat.

- `--mode test`
  Menjalankan mode `test`, yang berarti model akan diuji pada data uji STS tanpa pelatihan tambahan.

```
2024-11-13 12:04:22,361 : MSRpar : pearson
    = 0.6015, spearman = 0.6148
2024-11-13 12:04:23,753 : MSRvid : pearson
    = 0.8558, spearman = 0.8548
2024-11-13 12:04:24,955 : SMTeuroparl :
    pearson = 0.5017, spearman = 0.6115
2024-11-13 12:04:27,290 : surprise.OnWN :
    pearson = 0.7511, spearman = 0.7036
2024-11-13 12:04:28,496 : surprise.SMTnews
    : pearson = 0.6714, spearman = 0.5829
2024-11-13 12:04:28,498 : ALL : Pearson =
    0.7458, Spearman = 0.6724
2024-11-13 12:04:28,498 : ALL (weighted
    average) : Pearson = 0.6932, Spearman =
     0.6895
2024-11-13 12:04:28,498 : ALL (average) :
    Pearson = 0.6763, Spearman = 0.6735

2024-11-13 12:04:28,500 : ***** Transfer
    task : STS13 (-SMT) *****


2024-11-13 12:04:29,781 : FNWN : pearson =
     0.6056, spearman = 0.6158
2024-11-13 12:04:31,488 : headlines :
    pearson = 0.7813, spearman = 0.7744
2024-11-13 12:04:32,756 : OnWN : pearson =
     0.8464, spearman = 0.8339
2024-11-13 12:04:32,757 : ALL : Pearson =
    0.8033, Spearman = 0.8088
2024-11-13 12:04:32,757 : ALL (weighted
    average) : Pearson = 0.7835, Spearman =
     0.7767
2024-11-13 12:04:32,757 : ALL (average) :
    Pearson = 0.7444, Spearman = 0.7414

2024-11-13 12:04:32,759 : ***** Transfer
    task : STS14 *****


2024-11-13 12:04:34,076 : deft-forum :
    pearson = 0.5744, spearman = 0.5615
2024-11-13 12:04:35,513 : deft-news :
    pearson = 0.7995, spearman = 0.7744
2024-11-13 12:04:37,694 : headlines :
    pearson = 0.7646, spearman = 0.7440
2024-11-13 12:04:39,505 : images : pearson
     = 0.8306, spearman = 0.8012
2024-11-13 12:04:41,398 : OnWN : pearson =
     0.8589, spearman = 0.8435
2024-11-13 12:04:43,939 : tweet-news :
    pearson = 0.7755, spearman = 0.7110
2024-11-13 12:04:43,941 : ALL : Pearson =
    0.7559, Spearman = 0.7259
2024-11-13 12:04:43,941 : ALL (weighted
    average) : Pearson = 0.7788, Spearman =
     0.7493
2024-11-13 12:04:43,941 : ALL (average) :
    Pearson = 0.7673, Spearman = 0.7393

2024-11-13 12:04:43,944 : ***** Transfer
    task : STS15 *****


2024-11-13 12:04:45,762 : answers-forums :
     pearson = 0.7552, spearman = 0.7601
2024-11-13 12:04:47,982 : answers-students
    : pearson = 0.7420, spearman = 0.7447
2024-11-13 12:04:49,861 : belief : pearson
    = 0.8132, spearman = 0.8297
2024-11-13 12:04:51,876 : headlines :
    pearson = 0.7998, spearman = 0.8045
2024-11-13 12:04:53,813 : images : pearson
     = 0.8551, spearman = 0.8714
2024-11-13 12:04:53,815 : ALL : Pearson =
    0.7931, Spearman = 0.8017
2024-11-13 12:04:53,816 : ALL (weighted
    average) : Pearson = 0.7953, Spearman =
     0.8039
2024-11-13 12:04:53,816 : ALL (average) :
    Pearson = 0.7930, Spearman = 0.8021

2024-11-13 12:04:53,818 : ***** Transfer
    task : STS16 *****


2024-11-13 12:04:54,665 : answer-answer :
    pearson = 0.6881, spearman = 0.6850
2024-11-13 12:04:55,268 : headlines :
    pearson = 0.7726, spearman = 0.7888
2024-11-13 12:04:56,058 : plagiarism :
    pearson = 0.8057, spearman = 0.8140
2024-11-13 12:04:57,438 : postediting :
    pearson = 0.8727, spearman = 0.8883
2024-11-13 12:04:57,989 : question-
    question : pearson = 0.6932, spearman =
     0.6877
2024-11-13 12:04:57,990 : ALL : Pearson =
    0.7498, Spearman = 0.7583
2024-11-13 12:04:57,991 : ALL (weighted
    average) : Pearson = 0.7675, Spearman =
     0.7741
2024-11-13 12:04:57,991 : ALL (average) :
    Pearson = 0.7664, Spearman = 0.7727

2024-11-13 12:04:57,991 :

***** Transfer task : STSBenchmark*****


2024-11-13 12:05:19,555 : train : pearson
    = 0.7861, spearman = 0.7641
2024-11-13 12:05:25,916 : dev : pearson =
    0.8049, spearman = 0.8090
2024-11-13 12:05:31,315 : test : pearson =
     0.7673, spearman = 0.7561
2024-11-13 12:05:31,318 : ALL : Pearson =
    0.7866, Spearman = 0.7747
2024-11-13 12:05:31,319 : ALL (weighted
    average) : Pearson = 0.7863, Spearman =
     0.7706
2024-11-13 12:05:31,319 : ALL (average) :
    Pearson = 0.7861, Spearman = 0.7764

2024-11-13 12:05:31,324 :

***** Transfer task : SICKRelatedness*****


2024-11-13 12:05:44,960 : train : pearson
```

```
    = 0.7931, spearman = 0.7189
2024-11-13 12:05:46,721 : dev : pearson =
    0.7820, spearman = 0.7414
2024-11-13 12:06:07,136 : test : pearson =
    0.7873, spearman = 0.7143
2024-11-13 12:06:07,140 : ALL : Pearson =
    0.7897, Spearman = 0.7176
2024-11-13 12:06:07,140 : ALL (weighted
    average) : Pearson = 0.7896, Spearman =
    0.7177
2024-11-13 12:06:07,142 : ALL (average) :
    Pearson = 0.7874, Spearman = 0.7249
```

```
------ test ------
+--------+--------+--------+--------+--------+-------------+--------------------+--------+
| STS12  | STS13  | STS14  | STS15  | STS16  | STSBenchmark | SICKRelatedness |  Avg.  |
+--------+--------+--------+--------+--------+-------------+--------------------+--------+
| 67.24  | 80.88  | 72.59  | 80.17  | 75.83  |    75.61    |       71.43        | 74.82  |
+--------+--------+--------+--------+--------+-------------+--------------------+--------+
+------+------+------+------+------+------+------+------+
|  MR  |  CR  | SUBJ | MPQA | SST2 | TREC | MRPC | Avg. |
+------+------+------+------+------+------+------+------+
| 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
+------+------+------+------+------+------+------+------+
```

Fig. 2. Tabel hasi evaluasi dengan data testing

- `STS12 hingga STS16`: Secara umum, nilai korelasi Spearman yang dihasilkan berada di kisaran 67% hingga 80%. Ini adalah hasil yang cukup baik, terutama pada dataset yang lebih baru seperti `STS13`, `STS15`, dan `STS16` yang menunjukkan performa lebih tinggi (nilai di atas 75%). Semakin tinggi nilai korelasi, semakin baik model dalam memahami kesamaan semantik antar kalimat, sehingga hasil ini bisa dianggap baik untuk sebagian besar dataset STS, terutama yang di atas 70%.
- `STSBenchmark`: Nilai Spearman sebesar 75.61 pada dataset ini menunjukkan bahwa model memiliki kemampuan yang cukup baik dalam mengukur kesamaan teks di data benchmark standar ini. Nilai di atas 70% umumnya menunjukkan pemahaman yang baik, sehingga model ini dapat diandalkan pada task kesamaan teks.
- `SICKRelatedness`: Pada dataset ini, model mencapai nilai Spearman 71.43. Hasil ini menunjukkan performa yang cukup memadai dalam menangani hubungan semantik antar kalimat, meskipun sedikit lebih rendah dibandingkan beberapa dataset STS lainnya. Namun, nilai di atas 70% masih dianggap baik dan bisa diterima untuk task SICK.
- `Rata-rata STS`: Rata-rata keseluruhan 74.82% untuk semua dataset STS menunjukkan performa model yang konsisten baik dalam menangani berbagai jenis data kesamaan semantik. Dengan rata-rata di atas 70%, ini menunjukkan bahwa model cukup solid dalam task STS secara umum.
- `Task Klasifikasi Tekstual (MR, CR, SUBJ, MPQA, SST2, TREC, MRPC)`: Semua nilai nol (0.00) pada task klasifikasi tekstual menandakan bahwa model tidak dievaluasi pada task-task ini atau tidak mendukung task klasifikasi tersebut. Ini bukan indikasi performa buruk, melainkan lebih pada ketidakhadiran evaluasi untuk task tersebut.

II. Screenshot menjalankan program

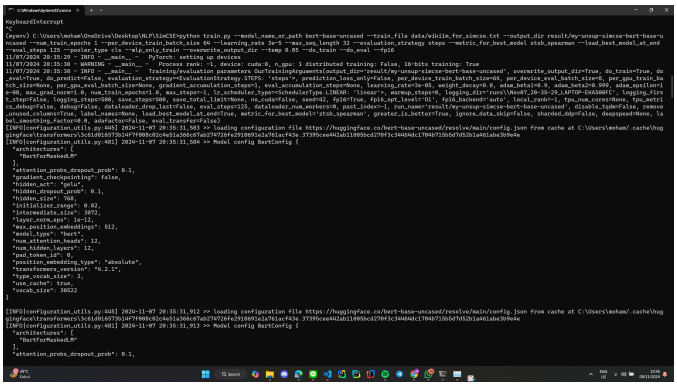

Fig. 3. Inisialiasi Pytroch dan Parameter training

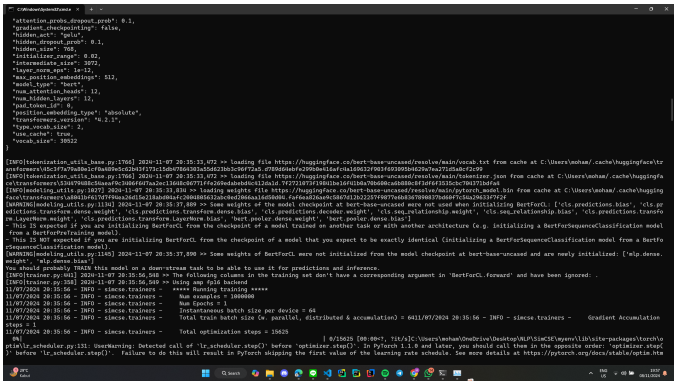

Fig. 4. Device Setup

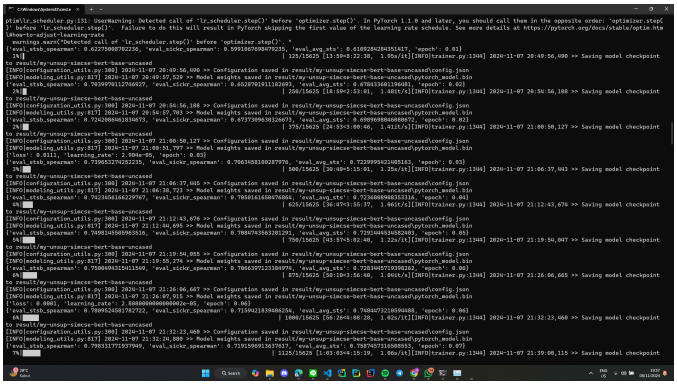

Fig. 5. Proses Pelatihan

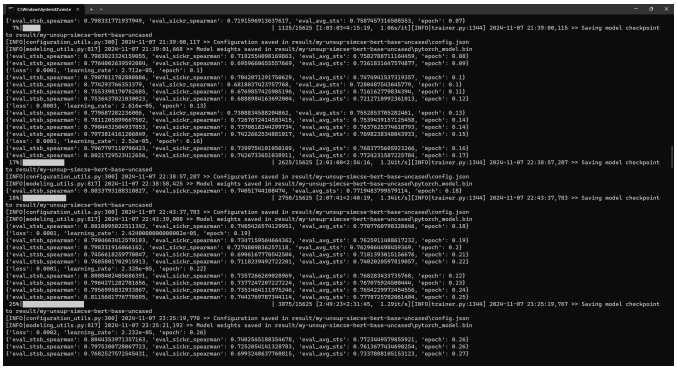

Fig. 6. Proses Pelatihan

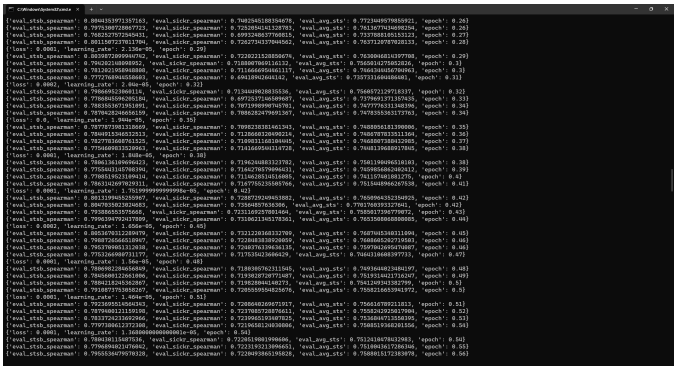

Fig. 7. Proses Pelatihan

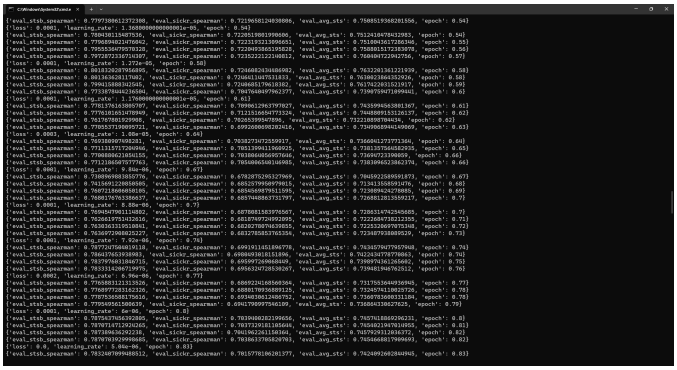

Fig. 8. Proses Pelatihan

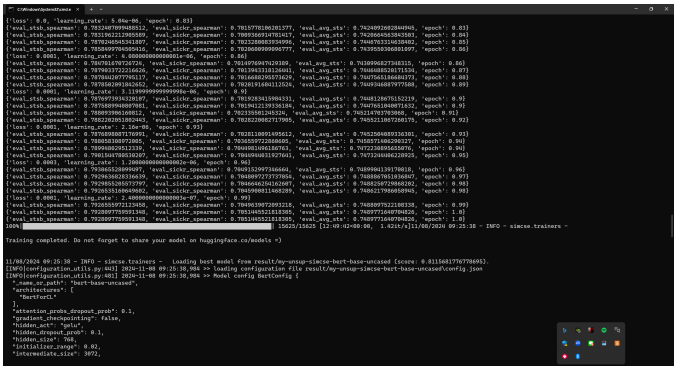

Fig. 9. Hasil Pelatihan

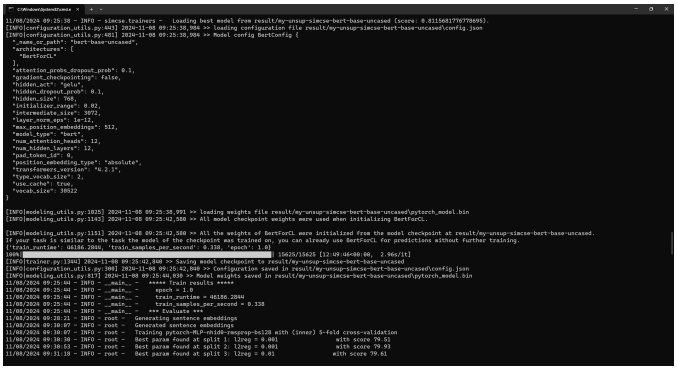

Fig. 10. Evaluasi Model Training

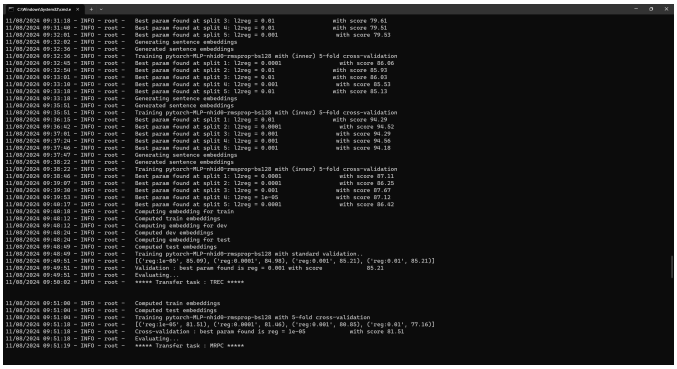

Fig. 11. Evaluasi Model Training

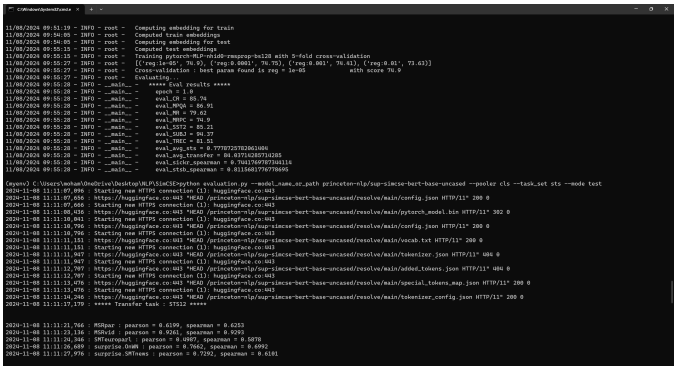

Fig. 12. Evaluasi Model Training serta inisialisasi evaluasi testing dengan model dari Huggingface (princeton-nlp/sup-simsce-bert-base-uncased

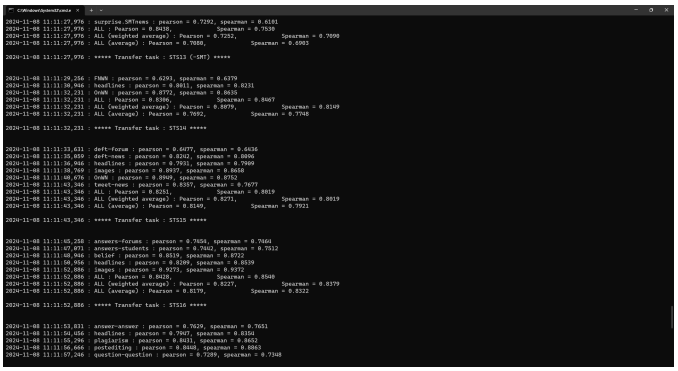

Fig. 13. Proses Testing pada model dari Huggingface (princeton-nlp/sup-simsce-bert-base-uncased

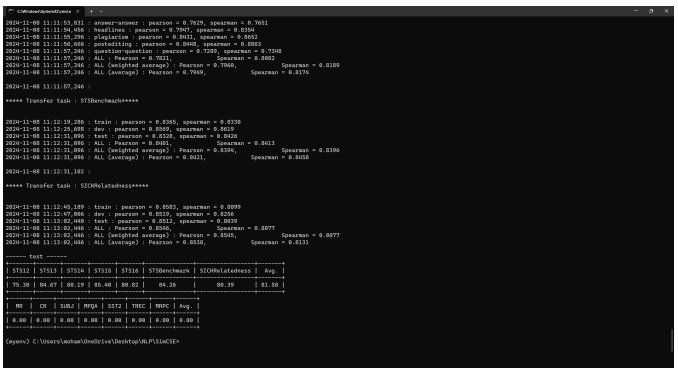

Fig. 14. Hasil Testing pada model dari Huggingface (princeton-nlp/sup-simsce-bert-base-uncased

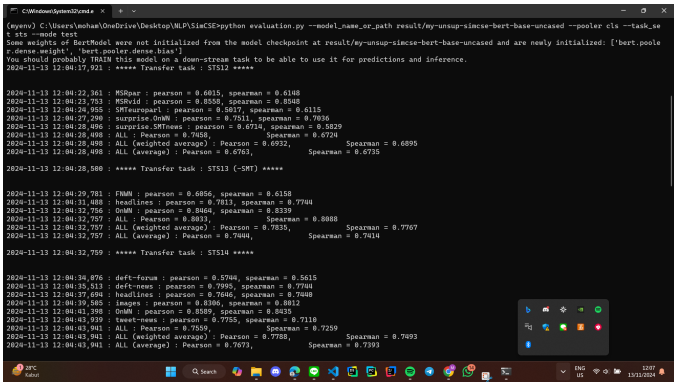

Fig. 15. Evaluasi Testing pada model dari hasil training lokal

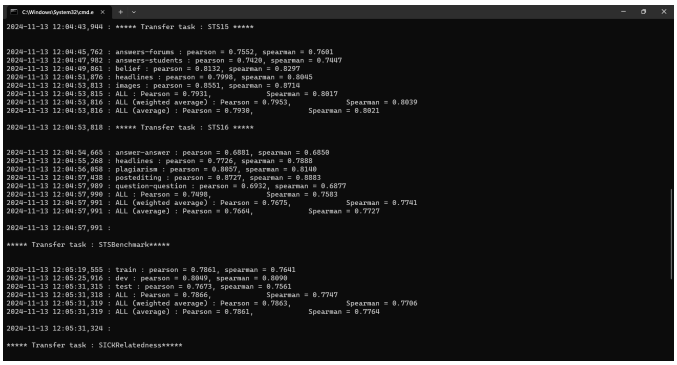

Fig. 16. Evaluasi Testing pada model dari hasil training lokal

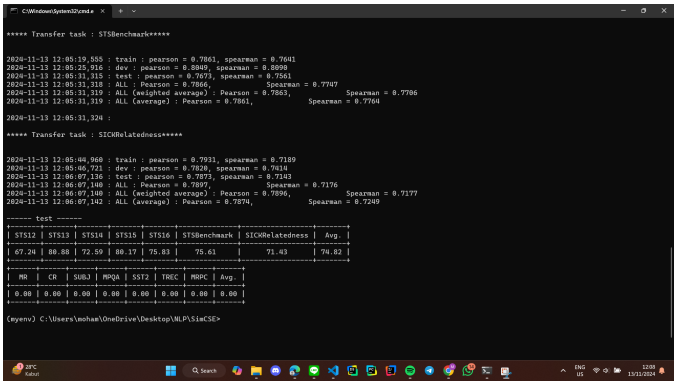

Fig. 17. Hasil Evaluasi testing pada model dari hasil training lokal