Sequential Transfer Learning for Multi-Granular English Semantic Textual Similarity and Search De-duplication

Jin Zhuoyuan　　Diao Daokui　　Zhu yongchao

***Abstract*—Semantic Textual Similarity (STS) is a key capability for retrieval and search de-duplication: systems must not only recognize broad semantic alignment, but also distinguish whether two texts are equivalent in intent or logic. However, SBERT-based sentence embedding models are often insufficiently sensitive to fine-grained semantic conflicts, which can yield falsely high similarity scores even when the underlying intents are logically opposite. To address this issue, we propose a sequential transfer learning framework: we first train on Quora Question Pairs (QQP) to learn a robust binary equivalence boundary, then fine-tune on the STS Benchmark (STS-B) to calibrate continuous similarity scoring, and finally compare against traditional baselines. Experiments show that our pipeline maintains strong regression performance on STS-B (Pearson $= 0.89$) while achieving reliable de-duplication performance on QQP (on a hard, high-error-rate subset) with Accuracy $= 0.913$ and F1 $= 0.89$. Overall, our approach improves robustness to subtle semantic differences and mitigates the weakness of sentence-vector models in capturing fine-grained semantic details.**

***Index Terms*—NLP, transfer learning, search deduplication,**

I. Introduction

Semantic Textual Similarity (STS) aims to measure how close two texts are in meaning [1]. In real-world scenarios, information redundancy is pervasive: users often express the same intent using different wording. If a system fails to recognize such equivalence, it leads to fragmented results and duplicated content, reducing retrieval efficiency and degrading user experience.

Although Transformer-based representations have substantially improved semantic modeling, a key practical gap remains: *semantic relatedness* is not the same as *intent-level or logical equivalence* [2]. Traditional lexical methods (e.g., TF-IDF) are brittle under paraphrasing and vocabulary mismatch [3]. Meanwhile, SBERT-style sentence embedding models may assign high similarity to sentence pairs that are lexically similar but conflict on crucial semantic details, thereby merging non-equivalent sentences and directly harming de-duplication quality.

Thus, we adopt a multi-granular view of text-pair modeling requirements: (i) **binary equivalence** for de-duplication, and (ii) **graded similarity regression** for fine-grained ranking and scoring. We propose a granularity-aligned sequential transfer learning strategy: we first leverage large-scale binary supervision from QQP to learn a clear equivalence boundary, and then fine-tune on STS-B to calibrate similarity scores on the 0–5 scale.

The core goal of this design is to preserve fine-grained scoring ability while increasing sensitivity to subtle semantic conflicts, thereby reducing false positives in de-duplication.

II. Motivation

A. *The Challenge of Information Redundancy:*

In platforms like Quora, users often pose similar questions using diverse phrasing [4]. A simple keyword-based matching system results in fragmented information and a poor user experience. There is an urgent need for systems that understand ”intent” rather than just ”words.”

B. *The Bottleneck of Current Embeddings:*

While Sentence-BERT (SBERT) has revolutionized efficiency by producing reusable sentence embeddings, it faces a significant ”fine-grained sensitivity” problem [4]. For instance, sentences like ”You should exercise” and ”You should NOT exercise” might have a cosine similarity above 0.9 in vector space despite being logical opposites. This is unacceptable for high-stakes de-duplication.

III. Related Work

A. *Pre-Transformer STS Methods*

Before Transformers, Semantic Textual Similarity was commonly addressed with (i) knowledge-based methods using lexical resources or knowledge bases, (ii) corpus-based distributional approaches, and (iii) hybrid systems that combined heterogeneous signals such as lexical overlap and semantic relations. Neural word embeddings later enabled dense representations that reduced dependence on surface overlap and improved robustness to paraphrasing and vocabulary mismatch [5].

B. *Transformer-Based Similarity and Fine-Grained Failure Modes*

Transformer pretrained language models, notably BERT and its variants, became the dominant backbone for STS by providing contextualized embeddings that better capture context-dependent meaning. However, practical evidence suggests a persistent gap between *semantic relatedness* and *intent-level/logical equivalence*: models may assign high similarity to sentence pairs that are lexically close but differ in crucial semantics , leading to false positives in retrieval and de-duplication. [6].

TABLE I
DATASET STATISTICS.

Dataset	Train	Dev	Test
QQP (binary)	363,846	40,430	390,965
STS-B (0–5)	5,749	1,500	1,379

IV. Data

We use two English sentence-pair datasets corresponding to two granularities: (i) **binary semantic equivalence** for search de-duplication, and (ii) **graded semantic similarity** for fine-grained scoring.

A. *Quora Question Pairs (QQP)*

The Quora Question Pairs (QQP) dataset targets **duplicate question detection**, i.e., whether two user questions express the same intent. Each example consists of a pair of questions (`question1`, `question2`) and a binary label: `duplicate` or `not_duplicate`. This task is commonly used as a proxy for industrial **search de-duplication** and paraphrase identification [7].

We use the official QQP version from the GLUE benchmark. The labeled release contains 404,290 question pairs and shows a moderate class imbalance: approximately 63.08% non-duplicates and 36.92% duplicates. Therefore, we report F1 in addition to Accuracy on the validation set to provide a more informative evaluation under imbalance.

B. *Semantic Textual Similarity Benchmark (STS-B)*

The Semantic Textual Similarity Benchmark (STS-B) provides **graded similarity annotations**. Each sentence pair is assigned a real-valued score on a **0–5** scale, where 0 indicates nearly no semantic overlap and 5 indicates semantic equivalence. It is commonly modeled as a regression task and evaluated with correlation metrics such as Pearson correlation to measure the agreement between predicted scores and human judgments [8].

STS-B contains 8,628 sentence pairs in total with train/dev/test splits shown in Table I. It covers multiple genres, including news (4,299 pairs), captions (3,250 pairs), and forums (1,079 pairs). This cross-domain diversity helps assess robustness across writing styles and contexts.

C. *Custom Fine-Grained Stress-Test Set (Ours)*

To explicitly evaluate robustness against fine-grained semantic conflicts that may lead to false positives in search de-duplication, we construct a custom stress-test dataset. The objective is to examine whether models can distinguish sentence pairs that are lexically similar but differ in crucial semantic aspects such as negation, reversals, and subtle intent shifts.

The dataset consists of 32 English sentence pairs with binary labels, where 1 denotes semantic equivalence and 0 indicates non-equivalence introduced through controlled semantic modifications. The dataset covers eight perturbation types, with four pairs per type. The overall label distribution includes 15 positive pairs and 17 negative pairs.

We apply a stratified split by label with a fixed random seed to create a development set and a test set. The development set contains 12 pairs and is used to tune the decision threshold by maximizing F1-score, while the test set contains 20 pairs and is used for reporting final robustness results. The dataset is strictly used for evaluation and is not involved in model training.

The perturbation types are defined as follows:

- **Negation:** Inserting or removing “not” or “no”.
- **Directional Change:** Replacing “increase” with “decrease”.
- **Comparative Flip:** Reversing “better than” to “worse than”.
- **Role Swap:** Exchanging subject and object (e.g., “A hit B” vs. “B hit A”).
- **Numeric Change:** Modifying quantities (e.g., “50%” vs. “10%”).
- **Quantifier Shift:** Changing “all” to “some” or “none”.
- **Modal Shift:** Adjusting certainty (e.g., “must” vs. “might”).
- **Direction Swap:** Reversing spatial relations (e.g., “North” vs. “South”).

V. Methodology

In this section, we describe the three hierarchical approaches implemented to evaluate Semantic Textual Similarity (STS) and search de-duplication. We transition from traditional lexical matching to static dense representations, and finally to our proposed Sequential Transfer Learning framework based on BERT Cross-Encoders [9].

A. *Lexical Baseline: TF-IDF*

As a primary baseline, we utilize the Term Frequency-Inverse Document Frequency (TF-IDF) method. This approach represents sentences as sparse vectors in a high-dimensional vocabulary space. For a given sentence pair (S_1, S_2), the similarity is computed using the cosine distance between their respective TF-IDF vectors v_1 and v_2:

$$\mathrm{sim}(S_1, S_2) = \frac{v_1 \cdot v_2}{\|v_1\|\|v_2\|} \tag{1}$$

While efficient, this method relies purely on lexical overlap and fails to capture semantic synonyms or context-dependent meanings [10].

B. *Static Embedding Baseline: Word2Vec with Mean Pooling*

To introduce semantic information, we employ pre-trained static word embeddings based on Word2Vec. Given a sentence $S = \{w_1, w_2, \ldots, w_n\}$, each token w_i is mapped to a dense vector representation $\mathrm{Emb}(w_i)$ learned from large-scale corpora. The sentence representation is computed using mean pooling over its constituent word vectors:

$$u = \frac{1}{n}\sum_{i=1}^{n} \mathrm{Emb}(w_i) \tag{2}$$

Similarity between two sentences is then measured using cosine similarity between their corresponding dense vectors [11]. This approach captures distributional semantic information encoded in Word2Vec embeddings. However, it represents the sentence as a "bag-of-words," ignoring word order, syntactic structure, and complex compositional semantics [**?**].

C. *Proposed Method: Sequential Transfer Learning BERT Cross-Encoder*

The core of our research addresses the limitations of Bi-Encoder models (e.g., SBERT). Bi-Encoders compute independent embeddings for each sentence, which often leads to "false positive" errors in scenarios involving fine-grained semantic conflicts, such as negations or subtle logical shifts (e.g., "A is B" vs. "A is NOT B").

To mitigate these issues, we propose a **Sequential Transfer Learning (STL)** framework using a **BERT Cross-Encoder** architecture [4].

*1) **Cross-Encoder Architecture**:* Unlike Bi-Encoders, our Cross-Encoder processes the sentence pair simultaneously. The input is formatted as:

$$\text{Input} = [\text{CLS}] + S_1 + [\text{SEP}] + S_2 + [\text{SEP}] \tag{3}$$

This allows the Transformer's self-attention mechanism to perform full-word-level interaction across both sentences at every layer, enabling the model to detect subtle discrepancies that independent pooling would overlook.

*2) **Sequential Training Strategy**:* To achieve multi-granular sensitivity, we implement a two-stage training pipeline:

1) **Phase 1: Binary Classification on QQP.** We first fine-tune the BERT model on the Quora Question Pairs (QQP) dataset ($N \approx 400,000$). The model learns a robust binary boundary for logical equivalence using a classification head over the $[CLS]$ token. The objective is to minimize the Cross-Entropy Loss:

$$\mathcal{L}_{CE} = -\sum y \log(p) + (1-y)\log(1-p) \tag{4}$$

This stage provides the model with the ability to distinguish between "similar-looking" but "logically distinct" queries [7].

2) **Phase 2: Continuous Regression on STS-B.** Using the weights from Phase 1, we adapt the model to the STS Benchmark (STS-B). We replace the classification head with a regression head to output a continuous similarity score $\hat{y} \in [0,1]$. The model is calibrated using Mean Squared Error (MSE):

$$\mathcal{L}_{MSE} = \frac{1}{M}\sum_{j=1}^{M}(y_j - \hat{y}_j)^2 \tag{5}$$

By bridging the gap between hard logical de-duplication and soft semantic scoring, the STL-BERT framework effectively enhances the model's robustness to nuanced semantic variances [8].

VI. Simulation

This section details the simulation framework designed to evaluate semantic matching models. Our methodology follows a five-stage pipeline: baseline establishment, model exploration, sequential training, unified evaluation, and robustness stress testing. The primary objective is to analyze model sensitivity to fine-grained semantic variances and stability under adversarial perturbations.

A. *Lexical and Static Baselines*

To establish a lower bound for performance, two categories of representative traditional baselines are implemented:

1) **TF-IDF Lexical Baseline:** Utilizing the classic Term Frequency-Inverse Document Frequency (TF-IDF) statistical method. Sentence pairs are encoded as sparse vectors based on word overlap. Similarity is determined by the cosine distance between vectors, representing the performance floor of literal word matching.
2) **Word2Vec Static Embedding Baseline:** We employ the pre-trained **Word2Vec (Skip-gram)** model from Google as the second control group. This model provides 300-dimensional dense vector representations. For sentence processing, a **Mean Pooling** strategy is applied, where the sentence vector $u = \frac{1}{n}\sum_{i=1}^{n} \mathrm{v}_i$ is the arithmetic average of all constituent word vectors. Semantic proximity is then evaluated via cosine similarity. This baseline tests the limits of non-contextual static embeddings in capturing semantics.

B. *Contextual Model Exploration*

Under a unified evaluation protocol, we conduct rapid validation across several mainstream Transformer-based pre-trained models, including **BERT**, **ALBERT**, **RoBERTa**, and **SBERT**:

- **Bi-Encoders (SBERT):** Mapping sentences into an independent embedding space and utilizing cosine similarity to optimize retrieval efficiency.
- **Cross-Encoders (BERT/RoBERTa):** Leveraging deep self-attention mechanisms for word-level interaction between sentence pairs, providing a high-precision benchmark for classification and regression tasks.

C. *Sequential Transfer Learning Strategy (STL-BERT)*

For BERT-based models, we implement a two-stage Sequential Transfer Learning (STL) strategy [12]:

- **Phase A (Paraphrase Pre-training):** The model is first trained on the Quora Question Pairs (QQP) dataset ($N \approx 400,000$) to learn robust binary decision boundaries for logical equivalence.
- **Phase B (Similarity Fine-tuning):** The weights are subsequently transferred to the Semantic Textual Similarity Benchmark (STS-B) for regression fine-tuning, calibrating outputs for fine-grained semantic intensity.

D. *Stress Test and Robustness Analysis*

We evaluate the robustness of SBERT-STS-B and our model BERT (QQP→STS-B) on the custom fine-grained stress-test dataset introduced above. The goal is to assess whether models can correctly distinguish sentence pairs that are lexically similar but differ in key semantic aspects, such as negation, comparative flips, role swaps, and subtle intent changes.

For each model, we report accuracy, F1-score, and correlation metrics on the test set. The development set is used to tune decision thresholds for fair comparison. Overall and category-level analysis reveals each model's strengths and weaknesses under different perturbation types, highlighting the advantage of cross-task transferred BERT in capturing subtle semantic distinctions.

E. *Evaluation Metrics and Presentation*

Accuracy and F1-score are reported for classification tasks, while Pearson (r), Spearman (ρ), and Mean Absolute Error (MAE) are reported for regression tasks. All thresholds are optimized on the validation set to ensure fair comparison across models.

VII. RESULT

This section evaluates the performance of the models on QQP (classification) and STS-B (regression) tasks, followed by a robustness stress test on the top-performing models.

A. *Overall Performance Comparison*

Table II summarizes the performance of lexical baselines, static embeddings, pre-trained contextual models, and our proposed Sequential Transfer Learning (STL) framework.

TABLE II
PERFORMANCE COMPARISON ACROSS DIFFERENT MODELS ON QQP AND STS-B TEST SETS.

Model	**QQP (Classification)**		**STS-B (Regression)**	
	Acc.	**F1**	**Pearson (r)**	**Spearman (ρ)**
Lexical (TF-IDF)	0.684	0.347	0.565	0.558
Static Emb (Word2Vec)	0.634	0.616	0.635	0.628
BERT (Pre-trained)	0.376	0.534	—	—
SBERT (Pre-trained)	0.736	0.713	0.866	0.839
BERT (STS-B)	0.614	0.634	0.844	0.804
SBERT (STS-B)	0.750	0.721	0.889	0.866
BERT (QQP→STS-B)	**0.922**	**0.891**	**0.900**	**0.875**

B. *Baseline and Pre-trained Models*

Traditional lexical matching (TF-IDF) and static embeddings (Word2Vec) define the performance floor, failing to capture semantic logic. Among pre-trained models, vanilla **BERT** performs poorly in zero-shot QQP classification (0.376), as its raw representations are not optimized for similarity tasks. In contrast, **SBERT** demonstrates superior zero-shot generalization on STS-B (0.866) due to its specialized Siamese network architecture.

C. *Single-Task Fine-tuned Models*

Specific fine-tuning on STS-B enhances the capture of semantic intensity, with **BERT (STS-B)** and **SBERT (STS-B)** reaching Pearson correlations of 0.844 and 0.889, respectively. However, their mediocre performance on QQP (Acc 0.61–0.75) suggests that regression-based training alone provides insufficient logical rigor for binary parity judgment.

D. *Two-Stage Sequential Transfer Learning*

The two-stage model, **BERT (QQP→STS-B)**, achieves the highest benchmarks with a 0.922 QQP accuracy and 0.900 STS-B Pearson correlation. This breakthrough stems from the synergy of ”logic first, perception later”: Phase A (QQP) establishes a robust logical constraint for equivalence, while Phase B (STS-B) calibrates these boundaries for fine-grained intensity. This strategy effectively bridges the gap between discrete logic and continuous similarity.

E. *Robustness and Stress Testing*

TABLE III
AGGREGATED ROBUSTNESS RESULTS ON STRESS TEST SUITE.

Model	**Acc.**	**F1**	**Pearson (r)**	**Spearman (ρ)**	**MAE**
BERT (QQP→STS-B)	**0.650**	**0.632**	**0.342**	**0.322**	**0.507**
SBERT-STS-B	0.550	0.571	0.200	0.235	0.528

1) Overall Robustness: As shown in Table III, BERT (QQP→STS-B) outperforms SBERT-STS-B across all metrics under semantic stress testing. BERT achieves an accuracy of 0.650 and an F1-score of 0.632, compared to 0.550 and 0.571 for SBERT. In terms of correlation, BERT obtains a Pearson coefficient of 0.342 and a Spearman coefficient of 0.322, exceeding SBERT's 0.200 and 0.235. BERT also yields a lower MAE (0.507 vs. 0.528). These results indicate that BERT maintains more stable similarity estimation and stronger ranking consistency under semantic perturbations.

TABLE IV
ACCURACY BY CATEGORY UNDER SEMANTIC PERTURBATIONS.

Category	**BERT (QQP→STS-B)**	**SBERT-STS-B**
Quantifier	1.00	1.00
Modal	**1.00**	0.50
Direction	**1.00**	0.67
Comparative	**0.67**	0.33
Numeric	0.67	0.67
Negation	0.50	0.50
Role Swap	0.33	0.33
Inc/Dec	0.33	**0.67**

2) Category-Level Analysis: Table IV shows that both models achieve perfect performance on Quantifier (1.00) and similar results on Numeric (0.67). However, performance drops for Negation (0.50 for both models) and Role Swap (0.33 for both), suggesting shared weaknesses in compositional semantic changes.

BERT demonstrates stronger robustness on logical perturbations, achieving 1.00 on Modal and Direction (compared to 0.50 and 0.67 for SBERT) and 0.67 on Comparative (vs. 0.33). In contrast, SBERT performs better on Inc/Dec (0.67 vs. 0.33).

Overall, BERT shows stronger robustness across most categories, while both models exhibit limitations under more complex semantic reversals.

F. contribution of each person

- **JIN Zhuoyuan:** Design and implementation of the STL-BERT model, design and construction of the custom dataset, overall performance comparison, robustness analysis of the model, and writing the Methodology, Simulation, and Result sections of the paper.
- **DIAO Daokui:** Training and fine-tuning of the STL-BERT model, development of baseline models and fine-tuned models, performance testing and comparison of the STL-BERT model, writing the Abstract, Introduction, and DATA sections of the paper, creating experimental charts, and organizing references.
- **ZHU Yongchao:** Implementation of baseline models , testing different models on the test sets, researching Related Work, summarizing failure cases of traditional STS methods and Transformer models in fine-grained semantics, and writing the Related Work and Motivation sections of the paper.

References

[1] D. Bär, C. Biemann, I. Gurevych, and T. Zesch, "Ukp: Computing semantic textual similarity by combining multiple content similarity measures," in ** SEM 2012: The First Joint Conference on Lexical and Computational Semantics–Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012)*, 2012, pp. 435–440.

[2] Y. Yang, S. Yuan, D. Cer, S. yi Kong, N. Constant, P. Pilar, H. Ge, Y.-H. Sung, B. Strope, and R. Kurzweil, "Learning semantic textual similarity from conversations," 2018. [Online]. Available: https://arxiv.org/abs/1804.07754

[3] P. Bafna, D. Pramod, and A. Vaidya, "Document clustering: Tf-idf approach," in *2016 International conference on electrical, electronics, and optimization techniques (ICEEOT)*, 2016, pp. 61–66.

[4] M. V. Koroteev, "Bert: a review of applications in natural language processing and understanding," *arXiv preprint arXiv:2103.11943*, 2021.

[5] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2013.

[6] Y. Zhang, J. Baldridge, and L. He, "PAWS: Paraphrase adversaries from word scrambling," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, 2019.

[7] Y. Reif and R. Schwartz, "Fighting bias with bias: Promoting model robustness by amplifying dataset biases," 2023. [Online]. Available: https://arxiv.org/abs/2305.18917

[8] D. Cer, M. Diab, E. Agirre, I. Lopez-Gazpio, and L. Specia, "SemEval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation," in *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, S. Bethard, M. Carpuat, M. Apidianaki, S. M. Mohammad, D. Cer, and D. Jurgens, Eds. Vancouver, Canada: Association for Computational Linguistics, Aug. 2017, pp. 1–14. [Online]. Available: https://aclanthology.org/S17-2001/

[9] A. Deshpande, C. Jimenez, H. Chen, V. Murahari, V. Graf, T. Rajpurohit, A. Kalyan, D. Chen, and K. Narasimhan, "C-STS: Conditional semantic textual similarity," in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, H. Bouamor, J. Pino, and K. Bali, Eds. Singapore: Association for Computational Linguistics, Dec. 2023, pp. 5669–5690. [Online]. Available: https://aclanthology.org/2023.emnlp-main.345/

[10] H. C. Wu, R. W. P. Luk, K. F. Wong, and K. L. Kwok, "Interpreting tf-idf term weights as making relevance decisions," *ACM Transactions on Information Systems (TOIS)*, vol. 26, no. 3, pp. 1–37, 2008.

[11] T. Gao, X. Yao, and D. Chen, "Simcse: Simple contrastive learning of sentence embeddings," 2022. [Online]. Available: https://arxiv.org/abs/2104.08821

[12] J. Phang, T. Févry, and S. R. Bowman, "Sentence encoders on stilts: Supplementary training on intermediate labeled-data tasks," *CoRR*, vol. abs/1811.01088, 2018. [Online]. Available: http://arxiv.org/abs/1811.01088