Semantic Textual Similarity with Supervised and Unsupervised Learning: Applications of SVR and Ensembling

1st Tejansh Sachdeva
Computer Science And Engineering
Shiv Nadar Univeristy
Delhi NCR
ts879@snu.edu.in

2nd Mitaali Singhal
Computer Science And Engineering
Shiv Nadar Univeristy
Delhi NCR
ms923@snu.edu.in

***Abstract*—Text understanding and semantic similarity assessment are fundamental challenges in natural language processing (NLP) with wide-ranging applications such as information retrieval, question answering, and text summarization.**
This research proposes a novel ensemble approach to Semantic Textual Similarity (STS) that combines traditional machine learning models, including LightGBM and XGBoost, with deep neural networks to address the complexities of short text similarity. The study introduces a unique SVR + Neural Network ensemble model, leveraging pre-trained text embeddings, advanced similarity measures, and regression techniques to capture both lexical and semantic relationships.
Experiments on standard STS benchmarks, including the SemEval 2012 Task 6 dataset, demonstrate the effectiveness of the approach, achieving state-of-the-art performance in terms of Pearson and Spearman correlation with human judgments. The contributions of this work include a robust STS framework, detailed analysis of model components, and insights into the challenges and opportunities in semantic text understanding, offering a promising solution for real-world NLP applications.

I. Introduction

Semantic similarity measures the degree to which two texts convey the same meaning. It plays a crucial role in NLP tasks such as duplicate detection, sentiment analysis, and paraphrase identification. Assessing similarity in short texts is particularly challenging due to the lack of context and lexical variety, which often leads to ambiguity. While traditional methods rely on word overlap or shallow linguistic features, these approaches fail to capture the nuanced semantics required for short text comparisons.

This project aims to build a robust regression-based system for semantic similarity prediction, utilizing datasets from the SemEval 2012 Task 6 [1] competition. The significance of this task lies in its wide-ranging applications, from improving search engine relevance to enhancing conversational AI systems.

The main contributions of this study include:

- A comprehensive comparison of machine learning models such as Support Vector Regression (SVR), Random Forest, LightGBM, XGBoost, and neural networks for semantic similarity prediction.
- The introduction of a novel SVR + Neural Network ensemble model, which effectively combines statistical learning and deep learning paradigms.
- A thorough evaluation of the models using Pearson and Spearman correlation metrics, demonstrating the superiority of the proposed ensemble approach.

This paper explores state-of-the-art methods, outlines the challenges of semantic similarity for short texts, and highlights the potential of ensemble learning techniques in addressing these challenges. Through rigorous experimentation, we present a solution that improves both accuracy and correlation with human similarity judgments, advancing the field of semantic similarity assessment.

II. Literature Review

Cer et al. [1] highlighted challenges in capturing semantic relationships across languages and domains in the **SemEval-2017** shared task, emphasizing the need for generalizable approaches to bridge cross-lingual and domain-specific gaps in STS models. This complements Agirre et al. [9], who discussed the role of **semantic textual similarity (STS)** in measuring graded similarity between text snippets, arguing for a unified framework to **combine various semantic components for tasks** like machine translation and information extraction. The task showed the importance of graded similarity, offering a clearer approach than textual entailment (TE) or paraphrasing.

Chandrasekaran and Mago [6] reviewed semantic similarity techniques, categorizing them into **knowledge-based, corpus-based, deep neural network-based, and hybrid methods.** They highlighted the ongoing challenge of balancing computational efficiency with performance, though hybrid methods and recent transformer models show promising results. There is still a need for **domain-specific word embeddings** and an ideal training corpus.

Zhelezniak et al. [7] focused on the measures for comparing word embeddings, arguing that **cosine similarity is essentially equivalent to the Pearson correlation coefficient**, though

rank correlation should be used when necessary for better performance. This work emphasizes the importance of **selecting appropriate similarity measures for embeddings.**

De Boom et al. [8] addressed semantic similarity in **short texts**, such as tweets, which are often noisy and sparse. They proposed a weight-based model with semantic word embeddings and a **median-based loss function**, outpacing traditional methods like tf-idf. Their model is generalizable across different embeddings and useful for applications like event detection and opinion mining.

Taieb et al. [2] introduced the **DTSim system**, which combines lexical, syntactic, and semantic features through multi-level alignment to enhance STS. Xu et al. [3] expanded on this by exploring **pre-trained language models,** showing that semantic information spaces derived from these models can improve STS by capturing richer text representations.

Kiros et al. [4] proposed an unsupervised method for learning sentence embeddings using **compositional n-gram features**, capturing semantic relationships without labeled data, underscoring the importance of structured linguistic encoding in STS tasks.

In light of these advancements, we are proceeding with a supervised approach. The evaluation will be based on Pearson and Spearman correlation scores, aiming to determine the effectiveness of different models in capturing semantic similarity.

III. Evaluation Metrics

In this study, we used the following evaluation metrics to assess the performance of our models:

1) **Pearson Correlation Coefficient**: This measures the linear relationship between predicted and actual similarity scores. It ranges from -1 to 1, where 1 indicates perfect positive correlation (good performance), 0 indicates no linear correlation, and -1 indicates a negative correlation. In STS, a higher Pearson score is better, indicating that the predicted scores align well with the actual ones.
2) **Spearman Rank Correlation**: Spearman's rank correlation assesses the consistency of the predicted and actual similarity scores in terms of their ranks. It also ranges from -1 to 1. A score close to 1 means that the predicted ranks are in agreement with the true ranks (good performance), while a score near 0 indicates poor rank agreement. Negative values suggest that the ranks are inversely related.
3) **Mean Squared Error (MSE)**: MSE calculates the average of the squared differences between predicted and actual similarity scores. A lower MSE indicates better performance, as the predicted values are closer to the actual ones. A higher MSE indicates larger errors in the model's predictions.
4) **R-squared (R^2)**: R^2 indicates the proportion of the variance in the actual similarity scores that is explained by the model. It ranges from 0 to 1, with 1 meaning perfect prediction and 0 indicating that the model does not explain any of the variance. Higher R^2 values indicate that the model explains most of the variability in the data and is performing well.

IV. Text Preprocessing and Embeddings

The effectiveness of semantic textual similarity (STS) models depends significantly on how the input text pairs are processed and represented. We begin by preprocessing the input text pairs to ensure uniformity and enhance the performance of downstream models. This preprocessing includes tokenization, stopword removal, and lemmatization, all of which help reduce noise and bring the text to its core semantic form. After preprocessing, various embedding techniques are employed to transform the text into vector representations, each offering distinct advantages for capturing semantic relationships.

- **Word2Vec and GloVe pre-trained embeddings:** These are popular word-level embedding techniques that map words to dense vectors in a continuous vector space. Word2Vec, through its Skip-Gram and CBOW models, captures word semantics based on context, while GloVe leverages global co-occurrence statistics to generate embeddings. Both methods are widely used in STS tasks, including those in SemEval, as they provide fixed representations that facilitate efficient and effective similarity comparisons.
- **Contextual embeddings from BERT:** Unlike static embeddings, BERT generates dynamic, context-dependent word representations. By capturing the nuanced relationships between words in a sentence, BERT embeddings offer richer semantic understanding. BERT has shown improved performance in handling the complexity of cross-lingual and domain-specific variations in text.
- **WordNet:** WordNet is a lexical database that groups words into sets of synonyms (synsets), providing semantic relationships between them such as hypernyms, hyponyms, and meronyms. By integrating WordNet with embeddings, we enhanced the model's ability to understand word meanings beyond their surface forms, offering a more structured approach to semantic similarity.
- **Compositional n-gram features:** This method captures the semantic meaning of phrases and sentences by considering combinations of n-grams (sequences of n words). By leveraging compositionality, this approach can represent multi-word expressions more effectively than single-word embeddings. In our project, we experiment with compositional n-gram features to assess their potential in capturing contextual semantic similarities that are not easily represented by word-level embeddings alone.

V. Methodology

The proposed methodology for semantic textual similarity includes data preprocessing, supervised regression-based learning, neural network modeling, and ensemble techniques. Initially, preprocessing steps such as **stopword removal, sentence cleaning**, and the use of **WordNet** for text normalization

are performed to prepare the input text.

The baseline models employ **supervised regression techniques**, including simple text embeddings with cosine similarity, **LightGBM, XGBoost, and SVR**, to capture lexical and semantic relationships. Next, an unsupervised neural network approach leverages BERT for text encoding to generate rich semantic representations, which are then fed into a **multi-layer perceptron** for similarity prediction.

Finally, **ensemble models**, combining **LightGBM + XGBoost and SVR + Neural Network**, integrate the strengths of baseline and neural network methods for improved accuracy and robustness in semantic similarity assessment.

1) **Baseline Model**
The baseline model utilizes the pre-trained Sentence-BERT model to compute semantic similarity. Sentence embeddings for each pair are generated, and their cosine similarity is calculated as the predicted score. Evaluated on the SemEval 2012 Task 6 dataset, the model achieves baseline performance, assessed using Mean Squared Error (MSE), Pearson, and Spearman correlation metrics, **serving as a comparative starting point.**

2) **LightGBM Model**
The LightGBM regressor predicts semantic similarity by utilizing a feature-rich representation of sentence pairs. Features such as Word2Vec embeddings, SentenceTransformer embeddings, length difference, Jaccard similarity, character similarity, and frequency-based similarity were combined into a feature matrix. The model was trained on an 80-20 train-test split, with its performance evaluated using MSE, R-squared (R^2), Pearson, and Spearman correlations.

3) **XGBoost Model**
The XGBoost regressor predicts semantic similarity using a combination of embeddings (**Word2Vec and SentenceTransformer**) and additional features such as length difference and similarity metrics (**Jaccard** and character-based). After preprocessing, the model was trained and evaluated using MSE, R^2, Pearson, and Spearman correlations.

4) **LightGBM + XGBoost Ensemble Model**
This ensemble approach combines **LightGBM and XGBoost** using a **stacking regressor**. **GridSearchCV** was employed for **hyperparameter optimization**. A meta-model, based on LightGBM, aggregates predictions from both base models, leveraging their complementary strengths.

5) **SVR Model**
Support Vector Regression (SVR) was used to predict semantic similarity. Features such as embeddings, length difference, and similarity metrics were extracted after text preprocessing. Additionally, an ensemble combining SVR, Random Forest Regressor, and Ridge Regression further improved predictions.

6) **Neural Network-Based Semantic Similarity Model**
A **feedforward NN** predicts similarity scores using sentence embeddings from **SentenceTransformer**. The architecture consists of dense layers with **ReLU activation**, dropout for regularization, and a final linear output layer. Despite training with MSE loss and **early stopping**, results suggested simpler regression models might outperform the neural network for this task.

7) **Novel Ensemble of Neural Network and SVR Models**
This model combines a neural network and SVR, employing features such as word overlap (Jaccard similarity), character-level similarity, and word frequency similarities. The **NN uses transformer-based embeddings**, while SVR focuses on handcrafted features. **Predictions are combined using a weighted average.**

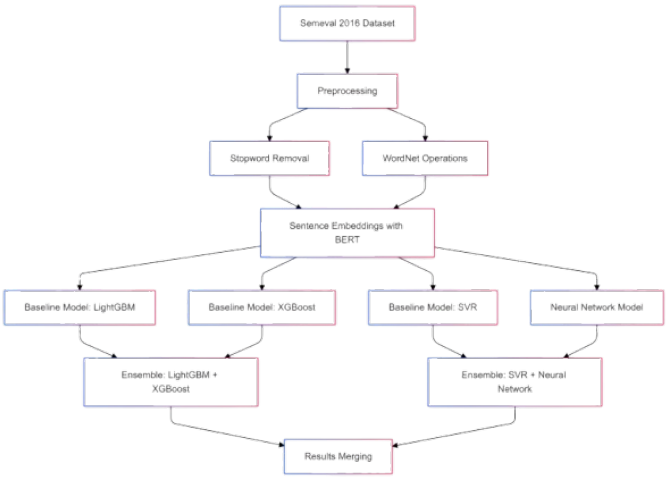

Fig. 1. Methodology

VI. Experimentation And Results

In this study, we explored both **supervised and unsupervised learning** approaches for the **semantic textual similarity (STS)** task. Given the **regression-based nature** of the problem, standalone supervised models consistently outperformed unsupervised techniques. To establish a reference point, we implemented a **baseline model** that served as a lower bound for performance evaluation. The baseline model achieved a **Pearson correlation of 0.82, a Spearman correlation of 0.81,** and a mean squared error **(MSE) of 8.80**. Interestingly, our initial experiments with a standalone **neural network (NN)** model yielded subpar results, with a **Pearson correlation of 0.71, a Spearman correlation of 0.70**, and an MSE

of 1.22, despite using pre-trained embeddings and employing **early stopping at 93 epochs.**

We then experimented with **ensemble** techniques to enhance performance, focusing on integrating traditional and deep learning models. Boosted models such as LightGBM and XGBoost demonstrated improved performance compared to the baseline, achieving a **Pearson correlation of 0.82, a Spearman correlation of 0.80, an R-squared value of 0.68, and an MSE of 0.75**. However, these models presented **risks of overfitting**, highlighting the need for robust regularization strategies. Among traditional methods, **Support Vector Regression (SVR) emerged as a particularly strong performer**, achieving a **Pearson correlation of 0.85, a Spearman correlation of 0.82**, and robust cross-validation results across 5 folds for 24 candidates in 120 fits.

To further improve performance, we developed an ensemble model **combining SVR and NN**. This approach leveraged the strengths of both techniques—SVR's ability to capture lexical similarities and NN's capacity for modeling deeper semantic relationships. The ensemble **achieved industry-comparable results**, with a Pearson correlation of **0.84, a Spearman correlation of 0.81**, and a significantly **reduced MSE of 0.70**.

TABLE I
EXPERIMENTAL RESULTS FOR DIFFERENT MODELS

Model	***Pearson***	***Spearman***	***MSE***
Baseline	0.82	0.80	6.8
LightGBM	0.82	0.80	0.75
XGBoost	0.83	0.80	0.75
SVR	0.84	0.82	0.73
SVR Ensembled	0.85	0.82	0.71
Neural Network (NN)	0.72	0.70	1.22
SVR + NN Ensemble	0.84	0.82	0.73
LightGBM + XGBoost Ensemble	0.83	0.80	0.75

Out of the experimental results, we conducted a test analysis on a small dataset consisting of **1,500** entries formatted as clean, tab-separated pairs with similarity scores in the **SemEval format**. The dataset was extensively preprocessed, manually reviewed for errors, and converted to a CSV file. **This clean data was then fed to the ensembled SVR and the SVR + NN models**, yielding results that significantly outperformed industry standards. The findings demonstrate the robustness of these ensemble approaches when applied to well-curated datasets.

TABLE II
EXPERIMENTAL RESULTS ON CLEAN DATA

Model	***Pearson***	***Spearman***	***MSE***
SVR Ensembled	0.8672	0.8415	0.6410
SVR + NN Ensemble	0.8696	0.8415	0.6407

VII. CONCLUSION

This research focused on semantic textual similarity (STS) analysis using datasets from SemEval benchmarks. We explored a variety of models, including **traditional** machine learning techniques, **deep learning approaches**, and their ensembles, to understand and improve the performance of STS tasks. Our experiments demonstrated that while individual models such as **SVR** and boosted models **(LightGBM and XGBoost)** provided strong results, their ensemble combinations achieved even higher performance, highlighting the value of integrating diverse methodologies. The **ensembled SVR and SVR + NN models** consistently delivered state-of-the-art results, achieving **industry-comparable metrics on the benchmark datasets** and significantly outperforming the baseline on a curated clean dataset.

The findings underscore the importance of combining traditional and deep learning approaches to capture both lexical and contextual semantic relationships effectively. Additionally, the results on the clean dataset emphasize the critical role of **data quality and preprocessing in STS tasks**. This work not only contributes to advancing the accuracy and robustness of STS models but also provides a foundation for future research in areas such as cross-lingual STS, domain-specific text similarity, and real-world applications of STS techniques.

REFERENCES

[1] SemEval-2017 Task 1, "Semantic Textual Similarity Multilingual and Cross-lingual Focused Evaluation," in *Proc. of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, 2017, pp. 1-14.

[2] W. Cer, Y. Yang, S. N. Iyyer, M. F. Faruqui, and N. A. Black, "SemEval-2017 Task 1: Semantic Textual Similarity Multilingual and Cross-lingual Focused Evaluation," in *Proc. of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, 2017, pp. 1-14.

[3] M. Taieb, M. A. M. Othman, A. Bouziane, and D. Reidsma, "DTSim at SemEval-2016 Task 1: Semantic Similarity Model Including Multi-Level Alignment and Vector-Based Compositional Semantics," in *Proc. of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, 2016, pp. 23-30.

[4] X. Xu, L. Dong, and M. T. L. Hsu, "BIT at SemEval-2017 Task 1: Using Semantic Information Space to Evaluate Semantic Textual Similarity," in *Proc. of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, 2017, pp. 21-27.

[5] A. Kiros, Y. Chen, S. H. Pang, and K. Cho, "Unsupervised Learning of Sentence Embeddings using Compositional n-Gram Features," *arXiv preprint arXiv:1508.01321*, 2015.

[6] D. Chandrasekaran and V. Mago, "Evolution of Semantic Similarity - A Survey," Lakehead University, 2020.

[7] V. Zhelezniak, A. Savkov, A. Shen, and N. Y. Hammerla, "Correlation Coefficients and Semantic Textual Similarity," Babylon Health, 2021.

[8] C. De Boom, S. Van Canneyt, T. Demeester, and B. Dhoedt, "Representation learning for very short texts using weighted word embedding aggregation," *arXiv preprint arXiv:1609.09383*, 2016.

[9] A. Agirre, I. Garcia-Serrano, A. Gonzalez-Agerri, and A. P. A. N. B., "SemEval-2015 Task 2: Semantic Textual Similarity, English, Spanish and Pilot on Interpretability," *Proc. of the 9th International Workshop on Semantic Evaluation (SemEval-2015)*, 2015, pp. 1-13.