Journal of Artificial Intelligence Research 82 (2025) 1243-1266 Submitted 04/2024; published 03/2025

Semantic Alignment of Malicious Question Based on Contrastive Semantic Networks and Data Augmentation

Xinyan Wang WANGXINYAN@WHU.EDU.CN
Jinshuo Liu LIUJINSHUO@WHU.EDU.CN
Juan Deng DENGJUAN@WHU.EDU.CN
Meng Wang WANG_MENG@WHU.EDU.CN
Qian Deng DQIAN0525@163.COM
Youcheng Yan YANYOUCHENG@WHU.EDU.CN
Lina Wang LNWANG@WHU.EDU.CN
School of Cyber Science and Engineering, Wuhan University, Wuhan, Hubei Province 430072 China

Yunsong Ma YUMA3828@UNI.SYDNEY.EDU.AU
School of Computer Science, University of Sydney Sydney, Australia

Jeff Z. Pan J.Z.PAN@ED.AC.UK
The University of Edinburgh, Edinburgh, EH89YL United Kingdom

Abstract

The identification and filtration of malicious texts in social media environments represent a significant technical challenge aimed at protecting users from online violence and disinformation. This complexity stems from the diversity and innovativeness of social media texts, which include unique expressions and special sentence structures. Particularly, malicious texts in interrogative forms pose alignment challenges with traditional corpora due to existing methods' failure to exploit the text's deep global semantic representations. This issue is compounded by the scant research on Chinese texts, leading to inefficiencies in recognition accuracy. To mitigate these challenges, we introduce an innovative framework based on a **G**lobal **C**ontrastive **S**emantic **N**etwork (**GCSN**), designed to enhance malicious text recognition efficiency and accuracy by deeply learning global semantic knowledge. It comprises an encoder for global semantic information modelling and a graph-matching network for semantic similarity evaluation between question pairs, enabling the accurate identification and filtering of malicious texts with complex structures. Furthermore, we introduce a semantic consistency-based data augmentation method (**COMBINE**), using real-world data to generate balanced positive and negative samples, enriching the dataset and enhancing the model's ability to distinguish semantic consistency through contrastive learning. Experimental validation on two Chinese datasets demonstrates our model's exceptional performance, affirming its applicationa value in social media malicious text recognition. Our code is available at https://github.com/Wxy13131313131/GCSN-COMBINE

1. Introduction

Question Semantic Alignment (QSA) falls under the broader category of Semantic Text Similarity (STS). It aims to match user-posed questions with questions or answers stored in relevant documents or knowledge bases to provide accurate and pertinent information

promptly. In the real of social media malicious text recognition, ordinary users are frequently safeguarded against undesirable online content by detecting and obstructing pertinent malicious information. While this approach effectively curtails the dissemination of harmful content across open networks, it falls short in effectively identifying harassing content within private message conversations. Consequently, users continue to endure harassment from malicious actors in the form of questions shown in Table 1. This challenge primarily stems from the diverse natural language expressions and short question structures in private messages. As a result, malicious texts in question format often elude accurate alignment with existing malicious corpora.

Blackmail Message	**Wire Fraudulent Message**	**Nuisance Message**
信息时代，你以为你在什么地方我们查不出来吗？ (In the information age, do you really believe there's anywhere you can hide from us?)	亲，有购物经验吗？刷单考虑？ (Hey there, are you experienced with online shopping? Ever thought about boosting order numbers artificially?)	今晚**宾馆约吗? (How about meeting at ** Hotel tonight?)
只是8000不到，截至5点会发生什么你知道吧？ (It's just under 8,000. You know what will happen by 5 o'clock, right?)	之前的银行账户被冻结了，可以再打6000吗？ (Your previous bank account was frozen, right? Can you send another 6,000 to resolve this?)	美女，出来玩玩？次付 (Hey, care to come out and have some fun? Payment per meet.)
账户一直不操作，这边是什么意思？后果你承担的起吗？ (Your account remains inactive; what exactly are you implying? Can you bear the consequences?)	稳定项目，还要考虑？抓紧时间打款 (A stable project, and you're still hesitating? Hurry up and make the payment.)	蕾丝裙！有更漏的试试吗？ (Lace dress! Interested in trying something more revealing?)

Table 1: **Three categories of malicious "question"** The data originates from real-world information collected in collaboration with China Mobile. To ensure privacy and confidentiality, sensitive information has been anonymized or excluded from the dataset. (The English text below is our translation, striving to preserve the original meaning as closely as possible.)

Traditional semantic alignment methods typically fall into two main categories: sentence encoding methods and sentence interaction methods (Lan & Xu, 2018). These approaches are commonly utilized to compute the similarity between lengthy sentences within high-dimensional spaces (Wang et al., 2021). However, due to the sparse content inherent in short texts, the efficacy of conventional string-based methods becomes less applicable. Several studies have illustrated that incorporating context information from short texts aids models in comprehending text meaning more accurately and achieving semantic matching with greater precision than traditional string-based methods (Han et al., 2021). Nevertheless, when compared to typical short texts, private messages lack relevant contextual information, thus exacerbating lexical and global structural variability. To address this challenge, recent research (Othman et al., 2022; Li et al., 2024; Hu et al., 2021; Chen et al., 2022) has proposed various deep learning models based on lexical representations, such as (Othman et al., 2022) proposed a deep learning approach based on the Siamese architecture of Long Short-Term Memory (LSTM) networks to resolve semantic ambiguities through in-depth

analysis of words. However, these methods solely consider ambiguity at the word level and overlooks the ambiguity arising from deep global semantic information.

Since automatic parsers (Banarescu et al., 2013)have demonstrated superior capabilities in representing sentences with diverse structural variations as unified Abstract Meaning Representation (AMR) structures, they are considered more suitable for generating global semantic representations. In this study, we introduce **G**lobal **C**ontrastive **S**emantic **N**etwork (**GCSN**) for aligning global semantic representations of questions. GCSN employs the AMR framework to encapsulate global semantic structural information. It consists of a text encoder responsible for capturing the semantic nuances of questions, a graph encoder designed to model the abstract semantic structures of these sentences, and a graph matching network tasked with evaluating the similarity between the utterances. Initially, we employ a pre-trained language model to encode the textual semantic features of questions while preserving the initial semantic labels associated with each word. Subsequently, we generate abstract semantic graphs for questions to extract structural features via a custom AMR parser. This process yields a global semantic feature representation of the interrogative text based on the retained semantic labels. Furthermore, we introduce a two-layer Graph Neural Network (GNN) as the graph encoder, which inputs global semantic features in graph format into the model, ensurinng consistency in global semantic information across interrogative texts.Throughout this procedure, to enhance the model's suitability for the Chinese language environment, we introduce an extensive data augmentation technique specifically tailored for Chinese (**COMBINE**). This method seeks to cultivate a broader spectrum of positive and negative example samples derived from real-world data. Additionally, we employ contrastive learning to promote structural coherence among samples sharing identical semantic information, encouraging closer proximity within the sample space.

The primary contributions of this article are outlined as follows:

- We introduce an innovative framework based on a Global Contrastive Semantic Graph Network (GCSN), designed to enhance malicious text recognition efficiency and accuracy by deeply learning global semantic knowledge. It comprises an encoder for global semantic information modeling and a graph matching network for global semantic similarity evaluation between question pairs, enabling the accurate identification and filtering of malicious texts with complex structures.

- By analyzing real-world data, we categorize positive and negative example samples into five distinct categories. For these categories, we devise the data enhancement technique (COMBINE) encompassing algorithms such as entity substitution, synonym substitution, random insertion, word-level noise enhancement, and reverse translation. These methods are engaged to generate a more comprehensive set of the positive and negative example samples from real-world data.

- We validate the effectiveness of our proposed framework through experiments conducted on both the BQ and LCQMC datasets. Our results demonstrate the superior performance of the GCSN model in semantic alignment tasks.

2. Related Work

2.1 Question Semantic Alignment

QSA is a crucial task that has garnered significant attention for Natural Language Processing (NLP). Traditional methods for QSA primarily rely on representation-based text representation models. Reimers and Gurevych (2019) employed a siamese framework to independently obtain the semantic representation vectors of two texts and subsequently used the cosine similarity function to infer the relationship between them. Yu et al. (2020) utilized the Tree topology of TreeLSTM to perceive vector information across multiple dimensions. Furthermore, Qi et al. (2022) introduced a complete message transfer network (FITN), which incorporated a novel memory-based attention mechanism to transfer both representational and interactive information via a global interaction matrix. In parallel, Othman et al. (2022) proposed a deep learning approach based on a siamese architecture of LSTM networks to enhance the model's capacity to deeply analyse the lexical meaning of words and questions.

Despite the success achieved by representation-based methods in various short text matching tasks, they still exhibit certain limitations. These methods typically operate at the surface level of text and struggle to handle complex semantic environments. To address this, Zou et al. (2022) encoded word vectors using bidirectional LSTM networks and focused on crucial contextual information by aligning related entities across different texts using cross-attention coefficients. It also employed context-attention mechanisms to focus on significant information within the context. Additionally, some researchers opt to enrich the semantics of short sentences by introducing complementary statements or leveraging knowledge bases. Liu et al. (2023) proposed a model that combined contrastive learning and external knowledge, guiding the model to encode the original text semantically. Chen et al. (2022) used a self-attention mechanism to enhance the representation of brief texts contextualized by external information. Yang et al. (2021) introduced a novel computation method utilized both semantic and syntactic information, leveraging extensive corpora to convey word meanings and address polysemy issues. While previous studies have often enriched feature information by introducing complementary statements or knowledge bases, it is significant to note that additional knowledge may introduce noise and impact matching accuracy.

Moreover, there is a lack of research focusing on the Chinese. Chen et al. (2020) showed superior performance by incorporating word sequences and proposed a sentence matching framework leveraging multi-granularity input information with paired word lattices formed by multiple participle hypotheses. Ma and Guo (2023) devised a hybrid model for semantic matching in Chinese language based on multilevel external knowledge (HME). However, many of these studies primarily considered semantic impacts at the word level, neglecting the significant influence of Chinese grammatical structure on global semantic information.

2.2 AMR In Other Domains

In evaluating the polyglot applicability of AMR, Wein and Schneider (2024) explored its effectiveness in capturing semantic nuances in multilingual contexts, highlighting its capacity to discern subtle differences between sentences. Considerable scholarly attention has been directed towards AMR exploration, with researchers exploring its utility across diverse

linguistic landscapes, including Persian Takhshid et al. (2022), Turkish Oral et al. (2024), and Celtic Heinecke and Shimorina (2022), thus assuming pivotal roles in multilingual computational linguistics. This prior scholarly endeavor lays the foundational groundwork for extending the AMR framework to the Chinese language domain. Concurrently, AMR has found applications in various NLP domains. For instance, Kapanipathi et al. (2021) applied AMR to Question Answering, employing AMR parsing to facilitate question comprehension independent of task constraints. Additionally, Wein and Schneider (2023) integrated AMR into text translation endeavors. Given the thematic resonance between these studies and our own investigation, it implicitly underscores the viability of applying AMR to QSA tasks.

3. Global Contrastive Semantic Network

We introduce a global contrasive semantic network (GCSN) based on contrastive semantic graphs, to enhance the extraction and implementation of global semantic information from questions. The architectural overview is presented in Figure 1. Before delving into the intricacies of our model, we also provide a theoretical overview of question semantic alignment and data augmentation. Additionally, our data augmentation algorithm (COMBINE) will be introduced comprehensively.

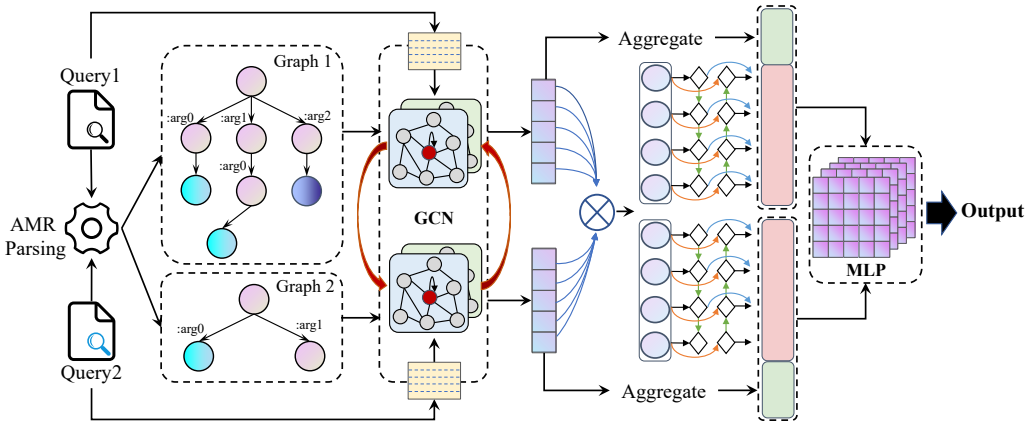


Figure 1: **Overview of the Structure of GCSN.** The figure illustrates a overview of GCSN. Initially, input question pairs undergo AMR parsing to construct their corresponding global semantic graphs. These graphs and original texts are subsequently encoded to represent both word and sentence level features. The model then facilitates features interactions at the word and sentence levels. Ultimately, the data are fed into a Multi-Layer Perceptron (MLP) to generate a score indicative of the semantic congruence between the two questions.

3.1 Background

Question Semantic Alignment The objective of the semantic matching task is to compute the semantic similarity relationship between two questions. S_a and S_b represent two distinct interrogative text sentences, where $S_a = \{v_1^a, v_2^a, \ldots, v_{n_a}^a\}$ and $S_b = \{v_1^b, v_2^b, \ldots, v_{n_b}^b\}$.

$\widetilde{y}$ denote the semantic similarity relationships between S_a and S_b, and determine whether two texts are semantically identical using the function $f(S_a, S_b)$. v_i^a and v_j^b represent the i-th and j-th tokenization in sentences S_a and S_b, respectively. n_a and n_b denote the lengths of the two texts.

Data Augmentation Before detailed introduction of the data augmentation framework we propose, we first provide a formal definition of the data augmentation task. For a given question text dataset $D = \{u_1, u_2, \ldots, u_n\}$, the purpose of data augmentation is to find a set $U_i = \{u^j, j \in (1, n)\}$ based on the semantic information s_i of the text u_i to expand the original dataset. Furthermore, the data augmentation methodology we utilize encompasses the subsequent algorithms.

3.2 Global Semantic Graph Construction

we extract the structural details of interrogative text using the AMR framework. Specifically, an automated parser is employed to convert the interrogative text into a Directed Acyclic Graph (DAG), wherein each node corresponds to a concept or entity, and edges encompass core roles (e.g. ARG0, ARG1), alongside non-core roles (e.g., time, location), modifiers (domains), and others. For example, the AMR graph is represented as $G = \langle V_G, E_G \rangle$, where V_G denotes the set of nodes and E_G signifies the set of edges. $E_G = \{(u, v, r) \mid (u, v) \in (V_G \times V_G), r \in R\}$, where r constitutes a set of predefined semantic relationship types and R is the set of all relations. Figure 2 illustrates the AMR structures of two examples, presented in both Chinese and English. In the graph on the left-hand side, the nodes "I" and "APP" denote two entity nodes. The predicates "develop-02" and "possible-01" are linked by ARG1. Furthermore, the "possible-01" node in this AMR structure denotes the answer sought for the interrogative. In the Chinese example depicted on the right-hand side, the nodes "红豆" (red bean) and "薏米" (barley) are connected to the same entity node "粥" (porridge), indicating a modifying relationship and associated with the predicate "减肥-01" (weight loss). The answer to the interrogative is represented by the "interrogative" node.

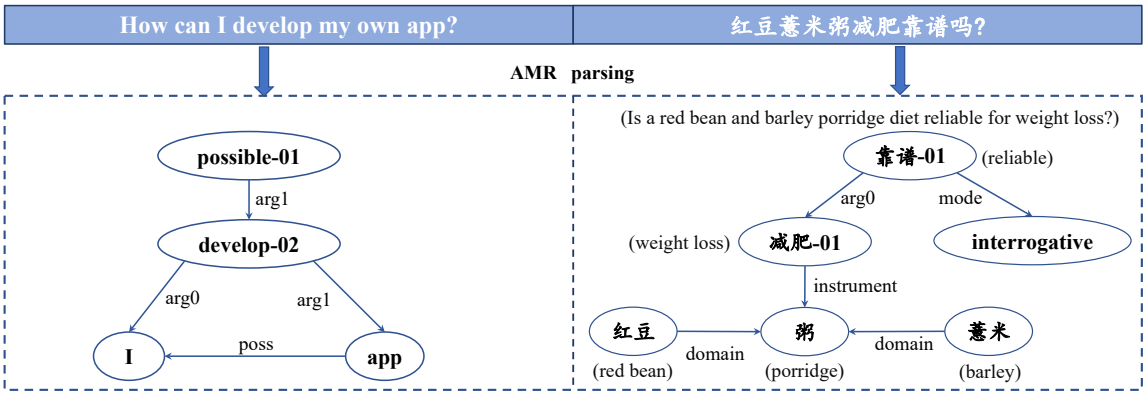


Figure 2: **Two examples of AMR structures in both Chinese and English languages.** We show two detailed examples of AMR structures in both Chinese and English.

3.3 Encoder

To acquire the interrogative text's global semantic feature representation, we begin by inputting the interrogative text into the text encoder. This process yields the global semantic information of the text and generates the initial semantic labels. Subsequently, the AMR nodes are initialized based on the semantic labels obtained.

Text Encoder RoBERTa (Liu et al., 2019) functions as a semantic encoder for questions, adept at comprehensively grasping and characterizing diverse linguistic nuances. Employing it enables us to extract the intricate global semantic details of the interrogative text and maintain initialized semantic labels for each word, facilitating the initialization of AMR nodes obtained from the automatic parser.

Graph Encoder Since AMR graphs are directed acyclic graphs with multiple edges, our graph encoder employs a two-layer graph convolutional neural network (R-GCN) (Schlichtkrull et al., 2018). Unlike GCN, R-GCN efficiently manages heterogeneous graph data with diverse relations. Moreover, R-GCN facilitates message passing and aggregation through parameter sharing. The update rule for node v_i in the graph is as follows:

$$h_i^{(l+1)} = \sigma\left(\sum_{r\in R}\sum_{j\in N_i^r}\frac{1}{c_{i,r}}W_r^{(l)}h_j^{(l)} + W_0^{(l)}h_i^{(l)}\right) \tag{1}$$

$h_i^{(l)} \in \mathbb{R}^{d^{(l)}}$ represents the l-th hidden state of node v_i in the graph, where $d^{(l)}$ denotes the dimensionality of the representation at this layer. N_i^r denotes the set of neighboring nodes of node v_i under relation $r \in R$; $W_r^{(l)} \in \mathbb{R}^{d\times d}$ represents the weight matrix of the edge, while $W_0^{(l)} \in \mathbb{R}^{d\times d}$ denotes the weight of nodes; $c_{i,r}$ is a problem-specific normalization constant assigned as $|N_i^r|$. To prevent overfitting rare relations, a basis-decomposition approach is adopted for regularization, enabling effective parameter sharing among different relation types. Each $W_r^{(l)}$ is defined as follows:

$$W_r^{(l)} = \sum_{b=1}^{B} a_{rb}^{(l)} V_b^{(l)} \tag{2}$$

For different types of relationships r, the parameter matrix $W_r^{(l)}$ is a linear combination of basis transformation $V_b^{(l)}$ and coefficients $a_{rb}^{(l)}$, where $V_b^{(l)} \in \mathbb{R}^{d^{(l+1)}\times d^{(l)}}$.

3.4 Global Semantic Interaction Network

The Multilayer Graph Matching Network (MGMN) architecture (Ling et al., 2021) has demonstrated remarkable success in graph matching tasks. We adopt a similar multilayer graph neural network to achieve semantic alignment between two questions. Additionally, during the pre-training phase, we enhance the alignment of global structural and semantic information in the textual semantics of questions by integrating the comparative graph learning method.

3.4.1 Word-Level Feature Interaction

To establish a more intricate and information-rich interaction between the two questions for learning their word-sentence-level embedding vectors, we first compute the cross-sentence attention coefficients $\alpha_{(i,j)}$ between each $v_i \in V_1$ in G_1 and all $v_j \in V_2$ in G_2. Similarly, we compute the cross-sentence attention coefficients $\beta_{(j,i)}$ between each $v_j \in V_2$ in G_2 and all $v_i \in V_1$ in G_1. Specifically, these two cross-sentence attention coefficients can be independently calculated using a function f_s, where $\vec{h}_i^l (l = 1, 2)$ symbolizes the i th node's hidden layer embedding in G_l.

$$\begin{aligned} \alpha_{i,j} &= f_s\big(\vec{h}_i^1, \vec{h}_j^2\big) = \text{cosine}\big(\vec{h}_i^1, \vec{h}_j^2\big), \quad v_j \in \mathcal{V}_2 \\ \beta_{j,i} &= f_s\big(\vec{h}_j^2, \vec{h}_i^1\big) = \text{cosine}\big(\vec{h}_j^2, \vec{h}_i^1\big), \quad v_i \in \mathcal{V}_1 \end{aligned} \tag{3}$$

Then, we learn the word-sentence-level embedding vectors. Specifically, by averaging the embeddings of every v_j in G_2 and their respective attention coefficients, we compute the sentence-level embedding vector $\tilde{\vec{h}}_{G,avg}^{2,i}$ of G_2 from the viewpoint ($v_i \in V_1$ in G_1). Similarly, $\tilde{\vec{h}}_{G,avg}^{1,i}$ is derived from the perspective $v_j \in V_2$ in G_2 as the word-sentence-level embedding vector of G_1. The following is how we compute these two word-sentence embeddings:

$$\begin{aligned} \tilde{\vec{h}}_{G,avg}^{2,i} &= \sum_{j \in \mathcal{V}^2} \alpha_{i,j} \vec{h}_j^2, \quad v_i \in \mathcal{V}_1 \\ \tilde{\vec{h}}_{G,avg}^{1,j} &= \sum_{i \in \mathcal{V}^1} \beta_{j,i} \vec{h}_i^1, \quad v_j \in \mathcal{V}_2 \end{aligned} \tag{4}$$

Next, we update the node vectors through a multi-view matching function f_m, and consider the newly generated node interaction features as the fresh feature matrices of graphs G_1 and G_2:

$$\begin{aligned} \tilde{\vec{h}}_i^1 &= f_m\left(\vec{h}_i^1, \tilde{\vec{h}}_{G,avg}^{2,i}, W_m\right), \quad v_i \in \mathcal{V}_1 \\ \tilde{\vec{h}}_j^2 &= f_m\left(\vec{h}_j^2, \tilde{\vec{h}}_{G,avg}^{1,j}, W_m\right), \quad v_i \in \mathcal{V}_2 \\ f_m(\vec{x}_1, \vec{x}_2, \vec{w}_k) &= \text{cosine}(\vec{x}_1 \odot \vec{w}_k, \vec{x}_2 \odot \vec{w}_k), \quad k = 1, ..., \tilde{d} \end{aligned} \tag{5}$$

Where $\vec{w}_k$ is the k-th learnable weight vector from the k-th view, $\odot$ is the element-wise multiplication operator. Since a multi-view matching function f_m encompasses a total of $\widetilde{d}$ views, the trainable weight matrix is $W_m = \{\vec{w_k}\}_{(k=1)}^{\widetilde{d}} \in \mathbb{R}^{d' \times \widetilde{d}}$. Subsequently, we obtain a $\widetilde{d}$-dimensional similarity feature vector, denoted as $\widetilde{\vec{h}} \in \mathbb{R}^{\widetilde{d}}$.

We employ bidirectional LSTM (BiLSTM) (Computation, 2016; Melamud et al., 2016) aggregation to acquire insights from the word sentence matching layer. We accept a randomized arrangement of word embeddings as input and BiLSTM indicates each sentence by jointing forward and backward final hidden layer outputs. $\tilde{\vec{h}}_G^1 \in \mathbb{R}^{2d}$ denotes the set of word-level embedding vectors of the sentence graph G_1 or G_2.

$$\tilde{\vec{h}}_G^l = BiLSTM\left(\left\{\tilde{\vec{h}}_i^1\right\}_{i=1}^{\{\text{N,M}\}}\right), \quad l = \{1, 2\} \tag{6}$$

Where N and M represent the number of nodes in graphs G_1 and G_2, respectively.

Finally, $[\tilde{\vec{h}}_G^1; \tilde{\vec{h}}_G^2]$ is a consists of two connected word-level embedding vectors. We subsequently feed these vectors into 3 fully connected layer and reduce their dimensions to one. We applied a sigmoid activation function to restrict the similarity score $\tilde{y}$ to the range (0,1).

$$\tilde{y_1} = s(G^1, G^2) = sigmod(MLP[\tilde{\vec{h}}_G^1, \tilde{\vec{h}}_G^2]) \tag{7}$$

3.4.2 Sentence-Level Feature Interaction

To better apprehend the sentence-level interactions, we employ the results of the direct aggregation computation as the original node embedding. Once we obtain the sentence-level embedding vectors $\vec{h}_G^1$ and $\vec{h}_G^2$, predicting their consistency becomes pivotal. Layer embeddings aptly capture the salient information of the graphs, rendering them invaluable for various tasks. To glean the inter-graph features and effectively assess graph similarity, we generate global embeddings for all nodes in G^1 and G^2.

Let $H^1 = \vec{h_i^1}(i=1)^N \in \mathbb{R}^{N \times d'}$ and $H^2 = \vec{h_j^2}(j=1)^M \in \mathbb{R}^{M \times d'}$ represent the context embeddings of all nodes in G^1 and G^2, respectively. Subsequently, leveraging the computed node embeddings H^1 and H^2, along with the node embedding vectors of G^1 and G^2, we connect them in the following manner and utilize them as the sentence-layer embedding vectors.

$$\vec{h}_G^l = BiLSTM\left(\left\{\vec{h}_i^l\right\}_{i=1}^{\{N,M\}}\right), \quad l = \{1, 2\} \tag{8}$$

Here we use the same BiLSTM aggregator function and prediction function as introduced in 3.4.1. Subsequently, we utilize the aggregated sentence-level embedding vectors $\vec{h_G^1}$ and $\vec{h_G^2}$ to compute $\tilde{y_2}$ using the following formula.

$$\tilde{y_2} = s(G^1, G^2) = sigmod(MLP[\vec{h_G^1}, \vec{h_G^2}]) \tag{9}$$

3.5 Output Layer

Following the preceding interactions between nodes and graphs, as well as between graphs themselves, we can enhance the capture of features from both individual nodes and the overall graph structure. This allows us to account not only for semantic consistency at the word level but also for the global semantic structure consistency when assessing graph consistency between the two graphs. Essentially, our model incorporates global semantic information not only during encoding but also when making consistency predictions. To conduct a more thorough analysis of the obtained sentence-level feature vectors and the word-level feature vectors, we connect the two embedding vectors of each sentence to get the final feature representation vector $\vec{H_G^1}$ and $\vec{H_G^2}$:

$$\begin{aligned} \vec{H_G^1} &= MLP[\vec{h_G^1}, \tilde{\vec{h}}_G^1] \\ \vec{H_G^2} &= MLP[\vec{h_G^2}, \tilde{\vec{h}}_G^2] \end{aligned} \tag{10}$$

These concatenated vectors are subsequently inputted into the prediction layer to yield the final similarity score $\tilde{y}$, as depicted below:

$$\tilde{y} = s(G^1, G^2) = sigmod(MLP[\vec{H_G^1}, \vec{H_G^1}]) \tag{11}$$

3.6 Positive and Negative Example Generation

To enhance the model's performance, particularly within the Chinese context, we introduce a tailored data augmentation method (COMBINE). This method aims to create question-text pairs with both global semantic consistency and inconsistency using real-world data, thereby enhancing and balancing existing datasets more effectively. By constructing positive and negative examples, the model learns global semantic consistency more efficiently during training. In this chapter, we will provide a detailed explanation of the positive and negative example construction process.

Example1	Example2	Tag	Type
为什么不理我啊你? (Why are you ignoring me?)	你为什么不理我? (Why are you ignoring me?)	1	Dialogic Text
寻衅滋事一般会怎么处理? (What's the usual course of action for provocations?)	寻衅兹事一般会怎么处理? (What's the usual course of action for provocations?)	1	Spelling Errors
有关于牢房的电影吗? (Is there a film about the prison ?)	有关于监狱的电影吗? (Is there a film about the prison ?)	1	Lexical Understanding
R17充电接口什么样? (What does the R17 charging port look like?)	R9m充电接口什么样? (What does the R9m charging port look like?)	0	Lexical Understanding
哎呀你说什么 (What did you say?)	啥玩意这说的 (What the hell is that?)	0	Colloquial Text
嗨！放首稻香怎么样 (Hi! How about a "Daoxiang"?)	你放的稻香怎么样呢 (How about the inari you put in?)	0	Syntactic Structure

Table 2: **Classification of Questions** The English text below is our translation, striving to preserve the original meaning as closely as possible.

The LCQMC dataset underwent an initial analysis, resulting in its division into five distinct categories according to positive and negative samples: lexical comprehension, syntactic configuration, orthographic inaccuracies, informal text, and conversational text. We give several sample texts in Table 2. It is noteworthy that certain English expressions, although completely synonymous in meaning, may lack precise equivalents in Chinese, due to variations in their structure and vocabulary.

In addition, we selected 2,000 samples of malicious text data from real-world sources [1]. These examples encompass a wide range of malicious text patterns encountered in real-world scenarios, ensuring the dataset reflects practical use cases. The selection was based

1. These data were provided by China Mobile. Due to data privacy concerns, we will not disclose our dataset publicly.

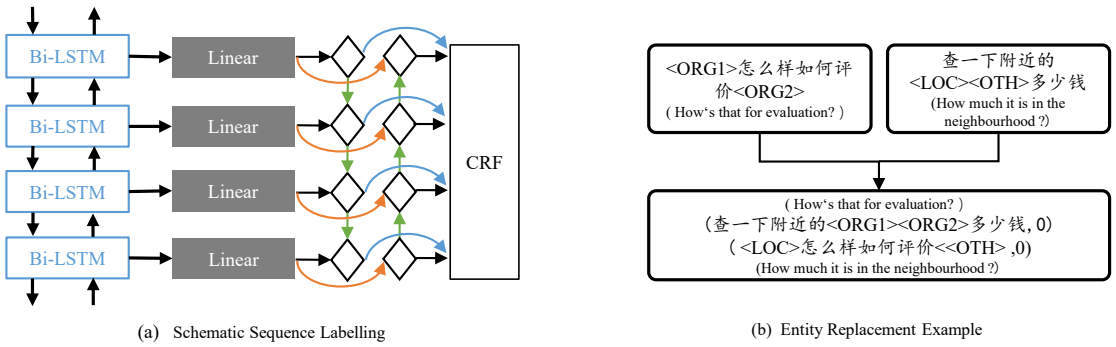
on factors such as text length, complexity, and ambiguity types to supplement the data distribution across five categories and balance the ratio of positive and negative examples.

Furthermore, we designed the COMBINE algorithm, which includes word-level entity replacement, synonym swapping, random insertion, lexical noise amplification, and sentence-level back translation. These algorithms are deployed to handle cross-lexical and sentence-level data from real-world sources, improving their practicality in model training and dataset enrichment by balancing the ratio of positive and negative examples.

3.6.1 Entity Replacement Algorithm

we employ named entity recognition techniques to discern and annotate the components of language and specific nouns present within the text. As depicted in Figure 3-(a), the input text $u = \{w_0, w_1, \ldots, w_n\}$ use BERT to obtain embeddings for word pairs, where w_i denotes the word in the input text u. Following this, BiLSTM computes character probabilities as prospective labels, succeeded by the application of the CRF layer to derive the ultimate sequence labeling outcomes.

The model's annotation utilizes the BIO annotation method, covering a range of entity types including personal names (PER), locations (LOC), works (WOR), organizations (ORG), and other proper nouns (OTH). Furthermore, synonym replacement and random insertion algorithms, which are based on masked language models, utilize parts of speech. This results in the annotation of text with corresponding parts of speech, including adjectives (ADJ), nouns (N), verbs (V), and other grammatical categories(O). Figure 3-(b) illustrates our example using Chinese.



(a) Schematic Sequence Labelling

(b) Entity Replacement Example

Figure 3: **Entity Replacement Algorithm.** As shown in the Figure3-(a), following the embedding of question pairs, we employ a BiLSTM to compute the emission probabilities for characters as candidate labels, with a CRF layer subsequently determining the final sequence labeling outcomes. Upon obtaining the named entity recognition results, we substitute the named entities with matching tag question pairs of the same type, thereby augmenting the original dataset, shown in Fingure 3-(b).

3.6.2 Synonym Replacement And Random Insertion Algorithms

Traditional synonym replacement techniques involve the utilization of synonym dictionaries or word embedding models like word2vec and FastText. However, these approaches solely assess similarity at the lexical level, disregarding global semantics and resulting in contextually inappropriate and unnatural sentences. Conversely, pre-trained models trained on masked language prediction tasks have the capability to mask segments of words and predict the masked portions based on contextual cues. Thus, we introduce a synonym replacement and random insertion algorithm leveraging a masked language model.

Specifically, given the original dataset $D = \{(x_1^1, x_1^2, y_1), (x_n^1, x_n^2 \cdot y_n)\}$ and a well-trained model M, where the input of the model (x_i^1, x_i^2) satisfies the model output $M(x_i^1, x_i^2) = y_i$ for $1 \leq i \leq n$. Assuming the current original sample is (x_1, x_2, y), the goal of generating new samples is to obtain x_{new} through synonym replacement or random insertion, ensuring similarity$(x_{\text{new}}, x) \geq \alpha$, where α is a similarity threshold, and similarity denotes the similarity function. The generated new samples and the original samples yield augmented samples $(x^1, x^2, x_{\text{new}}^1, x_{\text{new}}^2, y)$, where x_{new}^1 and x_{new}^2 represent the new samples generated based on x_1 and x_2, respectively.

Algorithm 1: Data Augmentation Algorithm Based on Masked Language Model

Input: Sentence Pair $SP = (x_1, x_2)$, label y, text similarity model M, mask language model MLM
Output: New example x_{new}

Text Segmentation $x = [w_1, w_2, ..., w_n]$; `// Segment the text`
Compute token importance $I_i \forall w_i \in x$; `// Calculate word importance`
for x *in* $[x_1, x_2]$ **do**
  for i *in descending order of* I_i **do**
    $S_{\text{mask}} \leftarrow S_{[1:i-1]}[t]S_{(i+1:na)}$; `// Create masked sentence`
    Predict top-K token T for mask M S_{mask} with MLM $L = \{\}$; **for** t *in* T **do**
      if *POS_Filter*(t, i) **then**
        $L[t] = S_{[1:i-1]}[t]S_{[i+1:na]}$;
        if $\exists t \in T$ *and* $M(L[t],) \geq \alpha$ **then**
          return $S_{\text{new}} \leftarrow L[t']$ where $L[t']$ has maximum similarity with x ;
          `// Choose text output satisfying semantic similarity`
        end
      end
    end
  end
end
return $S_{\text{new}} \leftarrow$ None;

The synonym replacement and random insertion algorithm based on masked prediction Algorithm 1, is divided into three steps in this paper: 1) Sentence preprocessing; 2) Ranking of word importance; 3) Random insertion and synonym replacement. Before initiating word replacement within a sentence, we prioritize words based on their semantic significance. This ranking assists us in selecting highly important words for replacement, thereby minimizing

interference with the text and preserving the original semantics to the fullest extent possible. The formula used to compute the importance of words is delineated as follows:

$$I_{w_i} = \begin{cases} G_y(x^1, x^2) - G_y(x^1_{/w_i}, x^2), \\ \text{if} G_y(x^1, x^2) = G_y(x^1_{/w_i}, x^2) \\ (G_y(x^1, x^2) - G_y(x^1_{/w_i}, x^2)) + \\ (G_y(x^1, x^2) - G_y(x^1_{/w_i}, x^2)), \\ \text{if} G_y(x^1, x^2) \neq G_y(x^1_{/w_i}, x^2) \end{cases} \tag{12}$$

where I_{w_i} denotes the importance of w_i in x^1, $x^1_{/w_i}$ denotes the text after removing w_i in x^1, and $G_y(x^1, x^2)$ is the confidence score of the model when the input is (x^1, x^2).

3.6.3 Word Granularity Woise Enhancement Algorithm

The algorithm delineating word-level text enhancement is depicted in Algorithm 2. The input comprises the original dataset D, alongside lists of homophones, morphemes, inflections, and a transition probability $\delta, \delta \in (0, 1]$.

Traversing through all sentence pairs in the dataset, we commence by executing word-level (Span-level) noise enhancement. This involves acquiring a random number $r \in (0, 1]$ through a random method. If $r \geq \delta$, the noise enhancement operation ensues, encompassing:

- The redundant word replacement operation extends the word by inserting a random word.
- The word omission replacement operation deletes the word to mimic a missing text error.
- The word order error replacement operation involves randomly swapping the positions of words to simulate word order errors.
- The word selection error replacement operation utilizes morphological and phonetic word lists for random replacements, thus simulating word selection errors.

Additionally, considering potential grammatical errors due to regional dialects present in the question text, such as variations in nasal sounds, rolling and flattening of tongues, etc., the text is initially recognized in pinyin format. Consequently, its consonants or rhymes are randomly substituted. Thus, the text is perceived in pinyin format, with its vowels or rhymes randomly replaced to mimic dialect pronunciation. For instance, "knowledge" may be recognized as "zhi-shi" in pinyin, and "zh" may be randomly substituted with "z" to produce "zi-shi". This process randomly replaces "zh" with "z", thereby converting pinyin to "zi-shi" and selecting words pronounced similarly, such as "姿势" and "滋事".

Upon completing word-level text enhancement, we proceed with token-level enhancement. Similarly, the replacement operation is conducted with a probability of $1 - \delta$, which includes:

- The redundant word replacement operation entails expanding the text through random word insertion. Primarily, this involves inserting random tone words from a

predefined list into the text, with insertion positions biased towards the beginning and end of the sentence.

- In the missing word replacement operation, words are deleted to simulate text omission errors.
- The word order error replacement operation involves randomly selecting words and altering their positions to simulate word order errors.
- The word selection error replacement operation employs homophonic word lists and other random replacements to simulate word selection errors.

Algorithm 2: Word-Span Granularity Noise Enhancement Algorithm

Input: Dataset D, Sentence Pair (x_1, x_2) in D, label=y, Homophone Word List, Homomorphous Span List, Modal Word, Transformation Probability=δ;
Output: New Dataset D_{noise};

$D_{\text{noise}} = \{\}$;
$ax = \text{wordList}_x = [w_1, w_2, ..., w_n]$; `// Text Segmentation`
for x *in* $[x_1, x_2]$ **do**
  changeDict $= \{\}$; `// Collect dictionary of replacement operations`
  for w *in* $wordList_x$ **do**
    $r = \text{random}()$;
    if $r \geq \delta$ **then**
      changeDict$[w] = w'$ change with span-level augmentation method;
      `// Word-level noise enhancement operation`
    end
  end
  for c *in* x **do**
    $r = \text{random}()$;
    if $r \geq \delta$ **then**
      changeDict$[c] = c'$ change c with token-level augmentation method;
      `// Character-level noise enhancement operation`
    end
  end
  x_{noise} = replace word or span of x with changeDict; `// Replace text`
  Put $(x_{\text{noise}}, x_1, y), (x_{\text{noise}}, x_2, y)$ into D_{noise};
end
Return: D_{noise};

Similarly, we randomly replace consonants or syllables to simulate dialect pronunciation. For example, "知" is recognized in pinyin as "zhi", and we randomly replace "zh" with "zi", selecting a single character under that pronunciation to replace. Likewise, we convert "z" to "zi" in pinyin and select a single character under that pronunciation to replace, such as "字 ("zi")" or "自"("zi"), and so forth.

When applying the aforementioned enhancement methods, we limit the editing distance between the generated text and the original text to no more than 3 to prevent excessive noise addition and preserve the original semantics. Algorithm 2 is a pseudo-code representation of this enhancement approach.

3.6.4 Back-Translation Algorithm

Back-translation is a frequently employed technique for data augmentation in machine translation, where BLEU serves as an evaluation metric (Sennrich et al., 2016; Shleifer, 2019). This method involves translating the original data into different languages using neural network models and subsequently translating the translated text back into the original language using the same model, thereby generating new text with identical semantics, the process is illustrated in Figure 4.

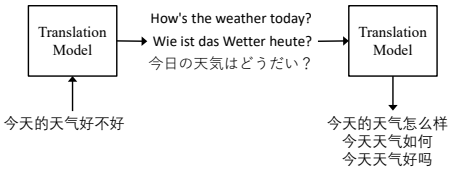


Figure 4: **Back-Translation Algorithm.**

Compared to developing standalone machine translation models, utilizing open translation platform APIs offers several advantages. These platforms are typically trained on extensive datasets, leading to more precise and robust back-translation outcomes. Among the commonly used translation platforms are Baidu Translate, Google Translate, and Youdao Translate. Given the abundance of proper nouns in the dataset, we took measures to ensure translation accuracy by selecting representative texts for testing. The results, presented in the Table 3 revealed that Google Translate effectively retained the noun information from the original text. Conversely, both Baidu Translate and Youdao Translate exhibited inaccuracies in translating proper nouns. Consequently, we opted to employ Google Translate[2] for the back-translation implementation.

Type	Example1	Example2
Original Text	打开酷我音乐播放一首音乐 (Open Kuwo Music and play a piece of music)	孙尚香是王者荣耀女英雄 (Sun Shangxiang is a heroine of King of Glory)
Baidu	打开KuWo音乐并播放音乐 (Open KuWo Music and play music)	孙尚香是国王的女主人公 (Sun Shangxiang is the king's heroine)
Google	打开酷我音乐播放一段音乐 (Open Kuwo Music and play a piece of music)	王者荣耀女主孙尚香 (King of Glory female lead Sun Shangxiang)
Youdao	打开cool Me Music，播放音乐 (Open cool Me Music and play music)	荣耀之王女英雄 (King of Glory Heroine)

Table 3: **A Comparison of Three Translation Models.** In this Table, we give two sample texts and their enhanced texts obtained after employing back translation method using each of the three translation models. (The English text below is our translation, striving to preserve the original meaning as closely as possible.)

2. https://translate.google.com/

4. Experiments

In this study, we conducted extensive experiments on two Chinese datasets, using accuracy and F1 score as the evaluation criterion, and compared our approach with leading question alignment techniques. To validate the effectiveness of our methodology, we partnered with China Mobile to gather user data, ensuring the anonymization of sensitive information to protect user privacy. Due to the sensitive and confidential nature of the data, it was not feasible to disclose it publicly. By manipulating these real-world datasets, we enhanced the LCQMC (Liu et al., 2018) dataset and prove the efficacy and accuracy of our method.

Parameter Settings we use an advanced and enhanced Chinese AMR parser[3] to parse the AMR structures of questions. The learning rate of the model is 5e-5, with a batch size of 64. The maximum input length for LCQMC and BQ is 128. We set seed as 42. All the aforementioned experiments are conducted on a 24 GB RTX4090.

4.1 Dataset and Baselines

Dataset We conduct experiments on two large Chinese language datasets, BQ (Chen et al., 2018) and LCQMC. BQ Corpus comprises question matching data within the banking and finance domain, consisting of 120,000 pairs of questions extracted from online banking system logs. The LCQMC dataset, derived from questions posed by users in the Baidu Knows community, adheres to the original paper's data segmentation. The LCQMC dataset has a positive-to-negative sample ratio of 1.38:1, while the BQ dataset maintains a balanced ratio of 1:1. In both datasets, the questions tend to have varying sentence lengths, so we adjusted the sentence lengths by either truncating or padding as necessary. Accuracy serves as an evaluation metric for assessing the model's performance across three datasets, the distribution of the datasets is detailed in Table 4.

Baselines The following models are our baseline models. Specifically, **BiMPM** (Mueller & Thyagarajan, 2016) and **ESIM** (Chen et al., 2017) serve as traditional interaction-based models, the interaction-based BiMPM employs BiLSTM to encode each sentence and compares two sentences from different perspectives. ESIM utilizes two BiLSTMs: one for encoding sentences and the other for integrating word alignment information between two distinct sentences. **ERNIE3.0** (Sun et al., 2021) and **ZEN2.0** (Song et al., 2021) are representative of pre-trained models. ERNIE is a multi-task pre-training framework for question semantic alignment tasks. ZEN2.0 is an improved n-gram enhanced pre-trained encoder for Chinese and Arabic. **COIN** (Hu et al., 2021) and **HME** (Ma & Guo, 2023) are both models that incorporate external knowledge. **LET-BERT** (Lyu et al., 2021) and **GMN-BERT** (Chen et al., 2020) employ semantic graph structures for training.

4.2 Main Result

This endeavor is particularly challenging due to the unique lexicon and sentence structures that pervade social media communication. Interrogative malicious texts, for instance,elude traditional alignment techniques primarily because these methods do not fully capture the deep global semantic structures inherent to such content. This challenge is aggravated by limited research on Chinese texts, resulting in suboptimal recognition accuracy.

3. https://hanlp.hankcs.com/

	LCQMC			BQ		
	Train	**Test**	**Val**	**Train**	**Test**	**Val**
TrainSample size (K)	238	8	12	100	10	10
Average length	12.1	12.3	11.5	11.4	11.7	11.1
Pos:Neg		1.38:1			1:1	

Table 4: **A Description of Two Datasets**

By addressing this limitation, our research introduces the GCSN framework, a novel construct based on a contrastive semantic graphs. This framework enhances the efficiency and precision of malicious text recognition by assimilating global semantic features. Central to this framework are an AMR parser dedicated to construct global semantic information. These graphs and original texts are subsequently encoded to represent both word and sentence level features. The model then facilitates features interactions of two levels. Additionally, we innovate a semantic consistency-based data augmentation strategy that capitalizes on real-world data to create a balanced amalgam of positive and negative samples. This strategy diversifies the dataset and refines the model's efficacy in discerning semantic consistency via contrastive learning.

Moreover, as shown in Table 5, advanced models like ERNIE2.0 and ZEN2.0 are outperformed by our GCSN framework. ERNIE 2.0's performance, for example, is commendable, particularly in LCQMC and showed an accuracy of 87.4, yet it does not reach the benchmark set by GCSN. The ZEN2.0 model, while impressive with 88.71 on LCQMC, also falls short of GCSN's results. It is important to note that while these models are effective, they exhibit a gap in the nuanced understanding of malicious content within the context of social media's diverse and innovative text structures.

Model	**BQ**		**LCQMC**	
	ACC	**F1**	**ACC**	**F1**
BiMPM	81.85	81.73	76.1	84.9
ESIM	81.93	81.87	82.58	84.49
COIN	-	-	86.20	87
ERNIE3.0	84.67	-	87.40	-
ZEN2.0	85.42	-	88.71	-
LET-BERT	84.80	84.98	87.60	88.85
GMN-BERT	85.60	85.5	87.3	88
HME	85.92	85.53	88.71	88.92
CBM-BERT	86.16	87.44	88.8	89.1
KSTM	87.22	88.64	89.00	89.17
GCSN(our)	**87.53**	**88.65**	**89.2**	**89.37**

Table 5: **Main Result**

The empirical scrutiny conducted on two pivotal datasets, BQ and LCQMC, affirms the exceptional performance of the GCSN model. As denoted in the provided Table 5, our model achieves an accuracy (ACC) of 87.53% and an F1 score of 88.65% on the BQ dataset, while on the LCQMC dataset, it attains an accuracy of 89.2% and an F1 score of 89.37%. These results are not only superior to other models but also epitomize the advanced capabilities of the GCSN model in understanding and processing semantically complex queries.

However, it is worth noting that we were unable to report the ACC and F1 metrics for some baseline methods (COIN, ERNIE3.0, ZEN2.0) due to the absence of comprehensive experimental results in their publicly available implementations or related literature. Additionally, we could not reproduce these metrics during our experiments because the necessary resources or details were not accessible from open sources.

In comparison with other robust models, such as BiMPM and ESIM, which present respectable accuracies and F1 scores, our model transcends in capturing the nuanced semantic variations specific to malicious content. While COIN demonstrates strong F1 performance, it does not match the holistic accuracy presented by our model. ERNIE2.0 and ZEN2.0, despite their advanced linguistic understanding, still fall short when confronted with the intricate task of semantic alignment in malicious interrogations, a task where GCSN excels. Furthermore, BERT-based adaptations like LET-BERT and GMN-BERT show satisfactory results but do not exhibit the same level of semantic discernment as GCSN. Models such as HME and CBM-BERT are marginally outperformed by our framework, highlighting the efficacy of the GCSN's integrated approach to semantic graph understanding.

We randomly selected 50 erroneous samples and invited three experts to analyze them. The analysis revealed that the primary source of errors, accounting for approximately 15%, originated from the construction tool. The second major source, contributing about 20%, was associated with the process of constructing negative samples.

4.3 Ablation Study

4.3.1 Global Semantic Network Ablation Study

We designed four sets of ablation experiments to ascertain the efficacy and essentiality of each component within the proposed Global Semantic Network:

- GCSN(DP+word-sentence) employs dependency syntax analysis to construct semantic graphs, utilizing our global graph matching network to validate the effectiveness of our global semantic graph.
- GCSN(GSN only) solely builds a global semantic graph and computes semantic consistency from the feature vectors of both graphs, serving to verify the entire global interaction matching network.
- GCSN(Word-level interaction) engages only word-level interactions within the global graph matching network, demonstrating the necessity of sentence-level interactions.
- GCSN(Sentence-level interaction), relies exclusively on sentence-level interactions, demonstrating the necessity of word-level interactions.

The ablation study presented in the Table 6 analyses the individual components of the GCSN framework on two datasets, BQ and LCQMC, to evaluate their respective contri-

butions to the model's performance. GCSN (DP+word-sentence), which uses dependency syntax analysis instead of AMR for constructing semantic graphs coupled with our global senmantic interaction network, demonstrates a significant improvement over GCSN (GSN only). This variation alone constructs a global semantic graph and computes semantic consistency based on the feature vectors of the two graphs, yielding a slightly reduced performance, indicating that the integration of the global senmantic interaction network plays a crucial role in capturing nuanced semantic relationships.

Model	BQ	LCQMC
GCSN(DP+word-sentence)	87.16	88.42
GCSN(GSN only)	85.72	87.96
GCSN(Word-level interaction)	86.63	88.5
GCSN(Sentence-level interaction)	86.22	88.64
GCSN(our)	**87.53**	**89.2**

Table 6: **Global Semantic Graph Ablation Study**

Moreover, when examining GCSN (word-level interaction), which employs only the word-level interactions within the global senmantic interaction network, there is a noticeable decline in performance compared to the complete model. This suggests that word-level interactions are significant yet insufficient to capture the full scope of semantic nuances. In contrast, GCSN (Sentence-level interaction), which solely relies on sentence-level interactions, fares slightly better, underscoring the importance of holistic graph analysis for semantic understanding. The most comprehensive performance is observed in the complete GCSN framework (our model), which integrates both word and sentence-level interactions along with the global senmantic interaction network. It outperforms all other ablations, confirming the hypothesis that a synergistic approach that combines multiple levels of interaction paramount for accurate semantic consistency. This superior performance exemplifies the effectiveness of our proposed GCSN framework, highlighting its potential as a robust tool for semantic graph-based applications in natural language processing.

4.3.2 Combine Ablation Study

We design three sets of ablation experiments to ascertain the efficacy and essentiality of each component within the proposed data augmentation methodology:

- Firstly, to evaluate the word-level data augmentation's effectiveness, we use algorithms for entity substitution informed by entity replacement (ER), synonym replacement and random insertion predicated on the Masked Language Model(MLM), and word noise augmentation(NOISE). These algorithms are applied to the original dataset, and the resultant enhanced datasets are evaluated for their question matching accuracy.

- Secondly, we formulate data augmentation strategies (ER+MLM+NOISE) at the word level and (BACKTRANS) at the sentence level. These are implemented to verify the utility of both modules and compare the enhancement outcomes given by each method..

- Lastly, the effectiveness of both the word and sentence-level modules is validated through a comparative analysis of the composite method's enhancement outcomes against those derived from individual word-level and sentence-level augmentations.

Table 7 presents the ablation study results, indicating that proposed model achieved high macro accuracy when applying the COMBINE method. Notably, the NOISE and MLM techniques showed superior performance in the spelling and syntactic structure metrics, respectively. The NER method boosted lexical understanding, spelling accuracy, and syntactic configuration by 1.38%, 1.91%, and 3.21%. Conversely, the MLM technique augmented these metrics by 2.75%, 1.92%, and 5.22%, respectively. Such enhancements underscore the efficacy of NER and MLM in broadening the sample space, thus bolstering the model's interpretative capacity. While the NOISE method led to a significant 16.46% improvement in spelling precision, it concurrently engendered a reduction in lexical comprehension and syntactic structure by 2.11% and 1.25%, which is attributable to the model assimilating erroneous examples due to an overly high noise ratio.

The BACKTRANS strategy highlights improvements in the model's comprehensive metrics, particularly elevating lexical understanding and spelling precision by 2.83% and 12.77%. This improvement stems from Back-translation's potential to enrich sample diversity and its inherent ability to rectify errors partially. Moreover, the tripartite NER+MLM+ NOISE approach, by consolidating three word-level data augmentation techniques, outperformed the BACKTRANS method, particularly enhancing syntactic and spelling metrics by 3.92% and 2.71%, respectively.

Methods	**Lexical Understanding**	**Spelling Errors**	**Syntactic Structure**	**Speech**	**Dialogue**	**Macro Accuracy**
Original	83.14	71.23	84.37	93.32	87.84	83.982
ER	85.53	73.14	87.58	93.37	87.60	85.25
MLM	85.90	73.15	89.60	94.49	88.35	86.29
NOISE	81.02	87.69	83.11	93.98	89.05	86.97
BACKTRANS	85.98	84.00	83.99	94.32	88.27	87.31
ER-MLM-NOISE	85.72	86.71	87.91	94.13	89.36	88.77
COMBINE	86.98	85.27	88.56	94.80,	89.88	89.10

Table 7: COMBINE Ablation Study

Implementing the COMBINE method amalgamates four augmentation algorithms, resulting in an unparalleled macro accuracy of 89.10% and an overall enhancement of 5.12%. This method acquired substantial gains in lexical comprehension, colloquial text, and dialogic text, marking increments of 3.83%, 1.48%, and 2.04%, respectively. The experimental results advocate that the fusion of these algorithms is instrumental in resolving diverse robustness issues, with their integration manifesting the most robust efficacy in the model.

4.3.3 Robustness And Generalizability Study

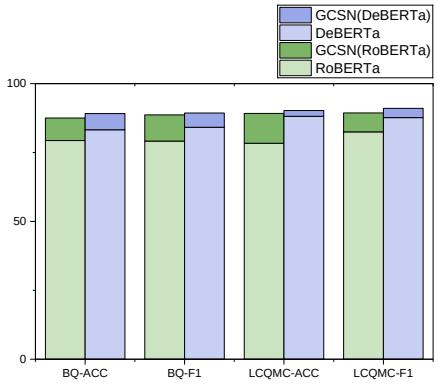


Figure 5: **Robustness and Generalizability Study**

We compared the experimental results of GCSN (DeBERTa) with DeBERTa and GCSN (RoBERTa) with RoBERTa on two datasets. As shown in Figure 5, the GCSN (DeBERTa) model consistently achieved the highest ACC and F1 scores in all scenarios, indicating its superior generalization ability when handling these types of data. In contrast, RoBERTa demonstrated the weakest performance, particularly on the BQ dataset, where both its ACC and F1 scores were lower than those of the other three models. Additionally, it is worth noting that for the same task type (e.g., BQ or LCQMC), the performance differences between the models were relatively small. This suggests that they might share certain common features or pattern recognition strategies to address these problems, thereby demonstrating the models' robustness and generalizability.

5. Conclusion

In this study, we present the GCSN, a contrastive pretraining framework tailored for the semantic alignment of questions, and the COMBINE method, a novel data augmentation technique predicated on semantic consistency. Through the deep integration of global semantic knowledge of natural language and the application of our proposed data augmentation via contrastive learning, our model significantly improves its proficiency in recognizing semantic consistencies. This enhancement, in turn, boosts the efficiency and accuracy of malicious text identification. The efficacy of our approach has been tested on two benchmark Chinese datasets, demonstrating superior performance. Furthermore, ablation studies highlight the critical importance of leveraging global semantic information and effective data augmentation for enhancing text matching models.

References

Banarescu, L., Bonial, C., Cai, S., Georgescu, M., Griffitt, K., Hermjakob, U., Knight, K., Koehn, P., Palmer, M., & Schneider, N. (2013). Abstract meaning representation for sembanking. In *Proceedings of the 7th linguistic annotation workshop and interoperability with discourse*.

Chen, J., Chen, Q., Liu, X., Yang, H., Lu, D., & Tang, B. (2018). The bq corpus: A large-scale domain-specific chinese corpus for sentence semantic equivalence identification. In *Proceedings of the 2018 conference on empirical methods in natural language processing*.

Chen, L., Zhao, Y., Lyu, B., Jin, L., Chen, Z., Zhu, S., & Yu, K. (2020). Neural graph matching networks for chinese short text matching. In *Proceedings of the 58th annual meeting of the Association for Computational Linguistics*.

Chen, M. Y., Jiang, H., & Yang, Y. (2022). Context enhanced short text matching using clickthrough data. *CoRR*.

Chen, Q., Zhu, X., Ling, Z.-H., Wei, S., Jiang, H., & Inkpen, D. (2017). Enhanced LSTM for natural language inference. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.

Computation, N. (2016). Long short-term memory. *Neural Comput*.

Han, M., Zhang, X., Yuan, X., Jiang, J., Yun, W., & Gao, C. (2021). A survey on the techniques, applications, and performance of short text semantic similarity. *Concurrency and Computation: Practice and Experience*.

Heinecke, J., & Shimorina, A. (2022). Multilingual abstract meaning representation for celtic languages. In *Proceedings of the 4th Celtic Language Technology Workshop within LREC2022*.

Hu, Z., Fu, Z., Yin, Y., & de Melo, G. (2021). Context-aware interaction network for question matching. In Moens, M., Huang, X., Specia, L., & Yih, S. W. (Eds.), *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*. Association for Computational Linguistics.

Kapanipathi, P., Abdelaziz, I., Ravishankar, S., Roukos, S., Gray, A., Astudillo, R. F., Chang, M., Cornelio, C., Dana, S., Fokoue-Nkoutche, A., et al. (2021). Leveraging abstract meaning representation for knowledge base question answering. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*.

Lan, W., & Xu, W. (2018). Neural network models for paraphrase identification, semantic textual similarity, natural language inference, and question answering. In *Proceedings of the 27th International Conference on Computational Linguistics, COLING 2018, Santa Fe, New Mexico, USA, August 20-26, 2018*. Association for Computational Linguistics.

Li, Q., Wu, C., Chen, J., Zhang, Z., He, K., Du, R., Wang, X., Zhao, Q., & Liu, Y. (2024). Privacy-preserving universal adversarial defense for black-box models. *arXiv preprint arXiv:2408.10647*.

Ling, X., Wu, L., Wang, S., Ma, T., Xu, F., Liu, A. X., Wu, C., & Ji, S. (2021). Multilevel graph matching networks for deep graph similarity learning. *IEEE Transactions on Neural Networks and Learning Systems*.

Liu, R., Zhong, Q., Cui, M., Mai, H., Zhang, Q., Xu, S., Liu, X., & Du, Y. (2023). The short text matching model enhanced with knowledge via contrastive learning. *CoRR*.

Liu, X., Chen, Q., Deng, C., Zeng, H., Chen, J., Li, D., & Tang, B. (2018). Lcqmc: A large-scale chinese question matching corpus. In *Proceedings of the 27th international conference on computational linguistics*.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., & Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. *CoRR*.

Lyu, B., Chen, L., Zhu, S., & Yu, K. (2021). Let: Linguistic knowledge enhanced graph transformer for chinese short text matching. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Ma, H., & Guo, H. (2023). A hybrid model based on multi-level external knowledge for chinese semantic matching. In *2023 IEEE International Conference on Big Data (BigData)*.

Melamud, O., Goldberger, J., & Dagan, I. (2016). context2vec: Learning generic context embedding with bidirectional lstm. In *Proceedings of the 20th SIGNLL conference on computational natural language learning*.

Mueller, J., & Thyagarajan, A. (2016). Siamese recurrent architectures for learning sentence similarity. In *Proceedings of the AAAI conference on artificial intelligence*.

Oral, E., Acar, A., & Eryiğit, G. (2024). Abstract meaning representation of turkish. *Natural Language Engineering*.

Othman, N., Faiz, R., & Smaïli, K. (2022). Learning english and arabic question similarity with siamese neural networks in community question answering services. *Data & Knowledge Engineering*.

Qi, L., Zhang, Y., Yin, Q., Zheng, G., Junjie, W., Li, J., & Liu, T. (2022). All information is valuable: Question matching over full information transmission network. In Carpuat, M., de Marneffe, M.-C., & Meza Ruiz, I. V. (Eds.), *Findings of the Association for Computational Linguistics: NAACL 2022*, Seattle, United States. Association for Computational Linguistics.

Reimers, N., & Gurevych, I. (2019). Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*. Association for Computational Linguistics.

Schlichtkrull, M., Kipf, T. N., Bloem, P., Van Den Berg, R., Titov, I., & Welling, M. (2018). Modeling relational data with graph convolutional networks. In *The semantic web: 15th international conference, ESWC 2018, Heraklion, Crete, Greece, June 3–7, 2018, proceedings 15*. Springer.

Sennrich, R., Haddow, B., & Birch, A. (2016). Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.

Shleifer, S. (2019). Low resource text classification with ulmfit and backtranslation. *CoRR*.

Song, Y., Zhang, T., Wang, Y., & Lee, K. (2021). ZEN 2.0: Continue training and adaption for n-gram enhanced text encoders. *CoRR*.

Sun, Y., Wang, S., Feng, S., Ding, S., Pang, C., Shang, J., Liu, J., Chen, X., Zhao, Y., Lu, Y., Liu, W., Wu, Z., Gong, W., Liang, J., Shang, Z., Sun, P., Liu, W., Ouyang, X., Yu, D., Tian, H., Wu, H., & Wang, H. (2021). ERNIE 3.0: Large-scale knowledge enhanced pre-training for language understanding and generation. *CoRR*.

Takhshid, R., Shojaei, R., Azin, Z., & Bahrani, M. (2022). Persian abstract meaning representation. *CoRR*.

Wang, H., Tian, K., Wu, Z., & Wang, L. (2021). A short text classification method based on convolutional neural network and semantic extension. *International Journal of Computational Intelligence Systems*.

Wein, S., & Schneider, N. (2023). Translationese reduction using abstract meaning representation. *CoRR*.

Wein, S., & Schneider, N. (2024). Assessing the cross-linguistic utility of abstract meaning representation. *Computational Linguistics*.

Yang, J., Li, Y., Gao, C., & Zhang, Y. (2021). Measuring the short text similarity based on semantic and syntactic information. *Future Generation Computer Systems*.

Yu, X., Li, G., Chai, C., & Tang, N. (2020). Reinforcement learning with tree-lstm for join order selection. In *2020 IEEE 36th International Conference on Data Engineering (ICDE)*. IEEE.

Zou, Z., Yang, W., Pang, L., & Liang, C. (2022). Interactive context-comparative model for text matching. In *2022 4th International Conference on Frontiers Technology of Information and Computer (ICFTIC)*. IEEE.