DMWE:Differential Testing for Multi-word Expressions in Machine Translation

Anonymous ACL submission

Abstract

With the advancement of deep neural networks, machine translation has greatly improved. Nowadays, people widely use machine translation tools to facilitate tasks such as reviewing foreign documents. However, due to the complexity of neural networks, translation errors can occur, leading to misunderstandings or conflicts. Existing machine translation systems often focus on sentence coherence, neglecting phrase translation accuracy, and most testing methods concentrate on the sentence hierarchy. This paper investigates multi-word expressions, a specific form of phrases prone to errors, and proposes Differential Multi-Word Expression testing method for machine translation (DMWE). We evaluated multi-word expressions by comparing their translation similarity across different translation software, based on the idea that phrase translations within the same sentence should be similar. Using three common types of multi-word expressions—Noun + Noun, Adjective + Noun, and Verb + Noun—we tested 1498, 1372, and 1525 sentences with Google Translate, Microsoft Bing Translator, and Baidu Translate. The results show that DMWE performs well in detecting translation errors with high precision.

1 Introduction

Machine Translation (MT) is a core technology in the field of Natural Language Processing (NLP), aimed at automatically translating text from one natural language to another through computer systems (Hazelwood et al., 2018). The goal is to break down language barriers and promote cross-cultural communication by enabling automatic translation between languages. In recent years, Neural Machine Translation (NMT) has seen significant breakthroughs, especially with the introduction of the Transformer model by Google in 2017 (Yang et al., 2013). As NMT technology continues to advance and mature, some advanced machine translation systems have achieved human-level performance in both quality scores and human evaluations, and their application scope has expanded widely (Hassan et al., 2018). Machine translation now plays a critical role not only in cross-lingual communication but also in various industries and fields (Bahdanau et al., 2015; Gehring et al., 2017; Devlin et al., 2019; Zhang et al., 2018), such as e-commerce (Tan et al., 2020), linguistic research, language education (Lee, 2023), and healthcare (Manchanda and Grunin, 2020).

Despite the significant progress in NMT technology, translation errors still occur. Research has shown that while current machine translation systems have improved fluency and naturalness, they often prioritize sentence coherence over the accurate translation of certain sentence components, particularly phrases with specific cultural or contextual meanings. This excessive focus on fluency and grammatical structure can lead to the loss or distortion of information, affecting both translation accuracy and semantic clarity. For example, as shown in Table 1, both Microsoft Bing Translator and Baidu Translate fail to correctly translate the idiomatic expression “kick the bucket” as “to die”, resulting in inaccurate translations. Similarly, as illustrated in Table 2, the sentence “She bought a new designer brand handbag” was translated by Microsoft Bing Translator as “设计师品” (designer brand), which, although not entirely incorrect, sounds unnatural within the context of the sentence.

Currently, most machine translation testing methods focus on sentence-hierarchy translations and do not effectively assess the accuracy of phrase translations. To address the shortcomings of current machine translation software in translating phrases, we propose DMWE and test it on Google Translate, Microsoft Bing Translator, and Baidu Translate. Using differential testing, we evaluate machine translation results by comparing

Table 1: Mis-translation of Google Translate and Microsoft Bing Translator

Software	Source Language	Target Language
Google	She kicked the bucket yesterday, but she had a lot of friends who helped her throughout her illness.	她昨天去世了，但是她有很多朋友在她患病期间给予了她帮助。
Bing		她昨天踢了水桶，但她有很多朋友在她生病期间帮助过她。
Baidu		她昨天放弃了，但她有很多朋友在她生病期间帮助了她。

Table 2: Inaccuracy Translate of Microsoft Bing Translator

Translation Software	Source Language	Target Language
Google	She bought a new designer brand handbag.	她买了一个新的名牌手提包。
Bing		她买了一个新的设计师品牌手袋。
Baidu		她买了一个新的名牌手提包。

phrase translations at different software and hierarchies, assessing the accuracy of machine translation. Compared to existing testing methods, our approach identifies more errors and demonstrates improved precision. This is the first application of differential testing to phrase-hierarchy machine translation testing.

In summary, the main contributions of this paper are as follows: (1)We introduce DMWE, a method for machine translation testing, which addresses existing gaps in phrase translation evaluation. (2)We implement and experimentally evaluate DMWE, demonstrating that it achieves high accuracy in machine translation testing. (3)We provide three commonly used multi-word expression types—Noun + Noun, Adjective + Noun, and Verb + Noun—along with datasets containing 1498, 1372, and 1525 sentences, respectively, thereby enriching the multi-word expression datasets.

2 Background

2.1 Differential Testing

Differential testing is a commonly used software testing technique designed to identify potential errors or inconsistencies by comparing the output results of two or more programs, modules, or versions under the same input conditions. The core idea of this method is to examine output differences based on input variations, which effectively uncovers latent defects, especially in complex algorithms or when comparing multiple implementations. In the context of machine translation, differential testing is primarily used to identify potential translation errors, inconsistencies, or performance issues by comparing the outputs of different machine translation systems or the same system across varying versions, configurations, or training conditions. Unlike traditional software testing methods, differential testing is particularly important in machine translation due to the diverse and semantically rich nature of translation outputs, making it difficult to define a single, absolute "correct" output. Differential testing offers a novel approach to evaluating translation quality, particularly for verifying consistency between different translation systems or conducting regression testing.

2.2 Word Alignment

Word alignment is a critical task in machine translation (Yang et al., 2013), aimed at establishing correspondences between words in the source and target languages within bilingual parallel corpora. It is an active area of research in natural language processing (NLP), and accurate alignment is essential for improving machine translation performance. Word alignment ensures that the translation system correctly interprets the source language and maintains semantic consistency between the source and target languages. Effective word alignment helps translation models handle lexical transformations more efficiently, thereby enhancing translation quality and accuracy. Early word alignment methods were primarily based on statistical models (Brown et al., 1990), such as the IBM Model Series (Brown et al., 1993). These models assumed a probabilistic relationship between source and target language words and used the Expectation-Maximization (EM) algorithm for parameter estimation. While these methods laid the theoretical foundation for early machine translation systems, they faced data sparsity issues when handling low-frequency vocabulary or long sentences. With the advent of deep learning, neural network-based word alignment methods have gradually become mainstream. The Seq2Seq model (Sutskever et al., 2014) employs an encoder-decoder architecture to encode source language sentences into fixed-length vectors, and the attention mechanism improves alignment accuracy. This

mechanism assigns weights to each word in the input sequence, allowing the model to focus on relevant parts, thereby enhancing translation performance. Currently, various word alignment models have been developed (Sabet et al., 2020; Liu et al., 2019). In this paper, we use AWESOME (Dou and Neubig, 2021), a deep learning-based model that combines the Transformer architecture and self-attention mechanism, supporting five language pairs. As an open-source tool, AWESOME performs well on large-scale datasets, marking a shift from statistical models to more flexible and efficient neural network-based approaches for word alignment.

2.3 Multi-word Expressions

Multi-word expressions (MWEs) are phrases composed of multiple lexical units combined through specific grammatical and semantic rules. MWEs are common across many languages and present significant challenges for language modeling and automated processing (Sag et al., 2002; Kim, 2008). The primary difficulty lies in their semantic non-compositionality, meaning that the overall meaning of an MWE cannot be inferred solely from its syntactic structure or the meanings of its individual components. For example, the phrase "go the extra mile" literally means "walk farther", but its actual meaning is "make an extra effort". Thus, correctly identifying and understanding MWEs is critical for handling such expressions. In NLP, one of the key tasks for effectively processing MWEs is their automatic identification. Solutions for automatic MWE recognition have been proposed in the PARSEME shared tasks (Savary et al., 2017, 2023), which have laid the foundation for further advancements in language modeling and NLP applications. Understanding and handling MWEs—along with their structural and semantic properties—are essential not only for language learning but also for the development of machine translation and NLP systems.

3 Methodology

The overall framework of DMWE is illustrated in Fig. 1. It consists of five major steps designed to evaluate the accuracy of translation systems in handling multi-word expressions (MWEs). To ensure consistency within the same translation software, DMWE pairs MWEs with sentences containing these expressions to form source language groups. These groups are then used to generate test inputs, which are verified using a differential testing approach.

The input to DMWE is a list of unannotated source language groups, and the output is a list of suspicious groups. The detailed workflow is as follows:

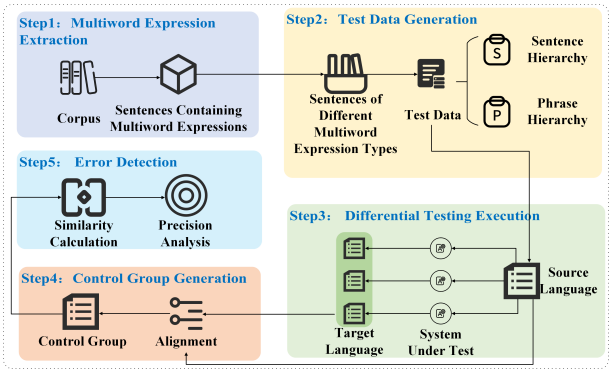

Figure 1: Flowchart of DMWE

3.1 Multi-word Expression Extraction

In the Multi-word Expression (MWE) extraction phase, the process of identifying and extracting MWEs involves several steps, with the core being the effective preprocessing of the sentences in the corpus and the use of specialized tools to identify MWEs.

To accurately identify MWEs, the first step is to preprocess the sentences in the corpus and convert them into a standardized CoNLL format. The specific process is as follows:(1) Tokenization: The sentences are split into words or subwords using a tokenization tool. (2) Part-of-Speech Tagging: A part-of-speech tagging model is used to assign corresponding part-of-speech labels (e.g., noun, verb, adjective) to each word. (3) Dependency Parsing: A dependency parsing tool is used to annotate the syntactic dependencies between each word and others in the sentence. After completing the preprocessing steps, all information is organized and output in CoNLL format. The advantage of the CoNLL format is its structured representation of each word's basic information.

Once the CoNLL format file is generated, MWEs can be extracted using the tool mwetoolkit3. mwetoolkit3 is specifically designed for processing MWEs and is capable of automatically identifying, analyzing, and handling MWEs in the CoNLL format. By combining the CoNLL format with mwetoolkit3, it becomes possible to efficiently identify and extract multi-word expressions from the corpus. This tool enhances the process by leveraging

the structured information in the CoNLL format, enabling precise and reliable extraction of MWEs for further analysis.

3.2 Test Data Generation

In the test data generation phase, it is necessary to extract target multi-word expressions by type from sentences containing these expressions and pair them with the complete sentence that includes the expression to form a source language group, thus constructing the test dataset. The source language group consists of both sentence-hierarchy and phrase-hierarchy components. For example, the sentence "She bought a new designer brand handbag" represents the sentence-hierarchy component, while the multi-word expression "designer brand" constitutes the phrase-hierarchy component. This forms a noun + noun type source language group.

3.3 Differential Testing Execution

In the differential testing phase, multi-word expressions are evaluated using the differential testing method. Specifically, the translation results from different translation software are compared to assess how they handle multi-word expressions, and the strengths and weaknesses of each software are analyzed. This study selects three mainstream machine translation tools: Google Translate, Microsoft Bing Translator, and Baidu Translate. These translation tools are widely used in various practical scenarios and represent the current mainstream level of machine translation technology.

As shown in Figure 2, the source language groups generated in the previous phase are input sequentially into each of the translation software being tested. The results from each software are then combined to form the corresponding target language groups. In the figure, the sentence-hierarchy "She bought a new designer brand handbag" and the phrase-hierarchy "designer brand" from the source language group serve as input, and all three translation tools output translations at both hierarchies. Each translation software corresponds to one target language group, resulting in three target language groups generated by the three software systems.

3.4 Control Group Generation

In the reference group generation phase, the alignment tool AWESOME (Dou and Neubig, 2021) is used to align the multi-word expressions between

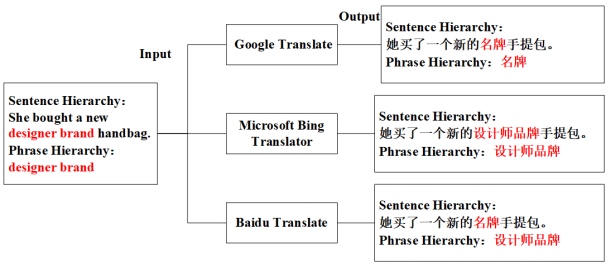

Figure 2: Differential Testing Execution

the source language sentence and the target language sentence, obtaining the corresponding translations of the multi-word expressions within the sentences. Each of the three translation tools generates a target language group, and after alignment, a total of six translation results for the multi-word expressions are obtained. These translation results form the reference group.

3.5 Error Detection

In the error detection phase, the translation accuracy of multi-word expressions by each translation software is further analyzed through similarity calculations. Each set of translation results includes translations at both the sentence and phrase hierarchies. To assess translation accuracy, DMWE employs two types of comparison: within-group comparison and between-group comparison. The within-group comparison measures the similarity of multi-word expression translations at different hierarchies by the same translation software, while the between-group comparison compares the similarity of translations of the same multi-word expression at the same hierarchy by different translation software.

This method uses BertScore (Zhang et al., 2020) for similarity calculations. BertScore is a text similarity evaluation method based on the BERT model, which is particularly effective for assessing the similarity of Chinese texts and can compute and return similarity scores between two phrases. DMWE sets a threshold to evaluate translation accuracy: if the similarity score is below the threshold t, the translation result is flagged as "suspicious".

$$\left[\forall s' \neq s, B_s(s, s') < t\right] \quad (1)$$

$$\left[W(s) < t \ \wedge \ \forall s' \neq s, B_p(s, s') < t\right] \quad (2)$$

Suspicious group identification is divided into two types. At the sentence hierarchy, if the translation of a multi-word expression by the selected software has a similarity lower than the threshold t when compared to the translations from the other

two software systems, it is considered suspicious. At the phrase hierarchy, where context is limited, if the similarity between the translation of a multi-word expression and the other two translations is below the threshold t, and the similarity with the same software at the sentence hierarchy is also below t, it is considered suspicious. A multi-word expression is deemed a suspicious group if it satisfies any of these conditions.

4 Experiments

This section addresses three aspects: precision, detection diversity and method effectiveness. Empirical evaluation of DMWE is conducted on the three translation software systems using the compiled dataset, with the experimental results analyzed to answer the research questions.

RQ1: What is the precision of DMWE in detecting errors in different types of multi-word expressions? The objective of this question is to assess the applicability of DMWE to different types of multi-word expressions and perform a precision analysis.

RQ2: What types of multi-word expression errors are detected by DMWE?The objective of this question is to evaluate the practicality of DMWE by analyzing the types of multi-word expression translation errors it can detect. The test results will be classified, and the corresponding error types and quantities will be counted.

RQ3: What are the advantages of DMWE? The objective of this question is to evaluate the effectiveness of DMWE by comparing it with other similar testing methods.

4.1 Datasets

This study extracted the relevant datasets from UM-Corpus (Tian et al., 2014), selecting three common types: Noun + Noun, Adjective + Noun, and Verb + Noun. To minimize false positives, sentence length was restricted to between 10 and 30 words. As a result, 1498, 1372, and 1525 sentences were extracted for each type, respectively.

4.2 Evaluation Metrics

This method uses BERTScore to calculate the similarity between two words, which is a standard cosine similarity formula to measure the similarity between two embedding vectors, c_i and r_j. Here, c_i represents the candidate translation's word vector, and r_j represents the reference translation's word vector.

$$\text{sim}\left(r_j, c_i\right) = \frac{c_i, r_j}{\|c_i\| \, \|r_j\|} \tag{3}$$

The precision of the results is calculated as follows: TP represents the number of groups that actually contain translation errors among all the error groups. FP represents the number of groups that were incorrectly identified as errors among all the error groups.

$$\text{Precision } = \frac{TP}{TP + FP} \tag{4}$$

4.3 Experimental Results and Analysis

4.3.1 Analysis Results for RQ1

Experiment 1 aims to evaluate the precision of DMWE. Using differential testing, a custom dataset was applied to assess the translation performance of three different machine translation software systems on multi-word expressions. The experiment involved manually verifying suspicious translations, with a translation being marked as an error only when both authors unanimously confirmed the mistake, ensuring the reliability of the results.

Before the experiment, 1,000 sentences were randomly selected for multi-word expression identification. The results showed that Noun + Noun, Adjective + Noun, and Verb + Noun were common types of multi-word expressions. These three types were tested separately, with a threshold range set between 0.6 and 0.75 for analysis. For the two hierarchies of translation content in the same translation software, a translation was considered problematic if there was an error in any hierarchy. The testing results are shown in Table 3, which displays the precision of the three translation software systems under different thresholds. For example, when testing the Noun + Noun type with a threshold of 0.7, the precision for Google Translate was 83.3%, for Microsoft Bing Translator was 85.1%, and for Baidu Translate was 80.6%.

We selected the Noun + Noun type of multi-word expression and created Figure 3 to show the relationship between threshold, precision, and true positives. The figure demonstrates that as the threshold decreases, the number of translation errors detected by DMWE decreases, while precision increases. This is expected, as lower thresholds make the translation software more conservative in processing translations, thereby reducing errors. However, excessively low thresholds may result in

Table 3: DMWE Precision

Type	Threshold	Google	Bing	Baidu
Noun+Noun	0.75	73.1%（49/67）	72.6%（53/73）	71.4%（36/49）
	0.7	83.3%（35/42）	85.1%（40/47）	80.6%（25/31）
	0.65	89.3%（25/28）	88.9%（24/27）	87.5%（14/16）
	0.6	90.1%（20/22）	93.0%（14/15）	91.6%（11/12）
Adjective + Noun	0.75	72.7%（101/139）	71.3%（97/136）	70.7%（70/99）
	0.7	82.1%（78/95）	80.2%（73/91）	79.3%（46/58）
	0.65	91.0%（51/56）	89.9%（62/69）	85.7%（30/35）
	0.6	92.9%（26/28）	92.1%（35/38）	94.4%（17/18）
Verb + Noun	0.75	70.7%（152/215）	71.7%（160/223）	72.3%（115/159）
	0.7	81.4%（114/140）	80.8%（126/156）	81.8%（85/104）
	0.65	90.4%（85/94）	90.6%（96/106）	88.5%（54/61）
	0.6	94.1%（48/51）	91.8%（45/49）	94.9%（37/39）

missing some true errors. Therefore, in practical applications, it is necessary to find the optimal balance between precision and the number of errors. The data in the figure shows that setting the default threshold to 0.7 allows for the detection of more translation errors while maintaining high precision.

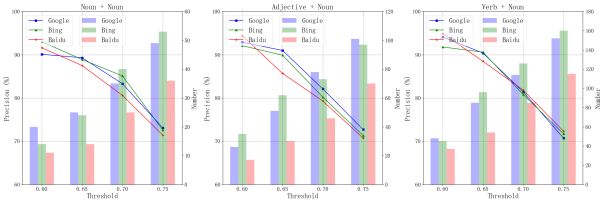

Figure 3: Threshold, Precision, and Quantity Chart for Different Types of Multi-word Expressions

This experiment demonstrates that DMWE is applicable to different types of multi-word expressions and helps identify the most suitable threshold during the testing process. At this threshold, DMWE can detect more translation errors while maintaining high precision, thereby improving the comprehensiveness and practicality of the testing.

4.3.2 Analysis Results for RQ2

Experiment 2 conducts an in-depth analysis of the translation errors detected by DMWE to identify their types. The experiment categorizes each erroneous result and examines the root causes, revealing specific issues encountered by different translation software when handling multi-word expressions, thus guiding future software improvements. Experiment 1 has already demonstrated that DMWE can effectively detect translation errors in multi-word expressions. During the evaluation process, DMWE successfully identified three main types of translation errors: mistranslation, omission, and non-translation.To gain a more comprehensive understanding of the detected error types, this experiment will provide a detailed description of each error type, aiming to uncover the differences in performance and potential issues among translation software when handling multi-word expressions.

Mis-translation: This occurs when a phrase in the source language is incorrectly translated into the target language. For example, in Table 4, the multi-word expression “lip service”, which means “superficial support or commitment that is not followed by actual action”, is mistranslated by Microsoft Bing Translator. The software’s translation is incorrect.

Table 4: Mis-translation

Software	English	Chinese
Google	lip service	口头承诺
Bing	lip service	唇部服务
Baidu	lip service	口头承诺

Under-Translation: Under-Translation errors occur when a phrase in the source language is not translated into the target language. For example, in Table 5, “key points” means “关键点” in Chinese, but Microsoft Bing Translator failed to translate it, resulting in an omission error.

Non-Translation: Non-translation errors occur when the source language phrase remains unchanged in the target language after translation. For instance, as shown in Table 6, Microsoft Bing Translator translated “let bygones be bygones” as “让bygones 成为bygones”, which is considered a non-translation error because “bygone” remains untranslated.

Under the default threshold t=0.7, the performance of Google Translate, Microsoft Bing Trans-

Table 5: Under-translation

Translation software	Microsoft Bing Translator
English	Some key points are suggested for designing recycled asphalt mixture as blacktop, which may be referable to the design and construction of utilizing recycled asphalt mixture.
Chinese	建议将再生沥青混合料设计为黑顶，这可能与利用再生沥青混合料的设计和施工有关。

Table 6: Non-translation

Translation software	Microsoft Bing Translator
English	Honey, I've made mistakes, too. Let's just let bygones be bygones, and start over.
Chinese	亲爱的，我也犯过错误。让我们让bygones 成为bygones，然后重新开始。

lator, and Baidu Translate in detecting translation error types and their quantities was manually recorded. The statistical results are shown in Table 7. For example, in the case of Noun + Noun type multi-word expressions, Microsoft Bing Translator identified 34 mis-translations, 4 under-translations, and 2 non-translations.

Table 7: Number of Translation Error Types by Software for DMWE

Type	Software	Mis	Under	Non
Noun+Noun	Google	33	2	0
	Bing	34	4	2
	Baidu	23	2	0
Adjective + Noun	Google	72	6	0
	Bing	61	7	5
	Baidu	42	3	1
Verb + Noun	Google	106	8	0
	Bing	104	13	12
	Baidu	75	10	0

Experiment 2 analyzes the translation error types of different software on datasets containing various types of multi-word expressions, demonstrating that DMWE is effective in identifying these issues and provides a comprehensive evaluation approach.

4.3.3 Analysis Results for RQ3

Experiment 3 compares DMWE with existing translation error detection methods to evaluate its effectiveness. For comparison, we selected the differential testing-based DCS, which has been shown to outperform metamorphic testing methods like CIT and CAT. Additionally, the availability of DCS's code implementation facilitates straightforward comparative experiments.

DMWE is specifically designed for testing multi-word expressions (MWEs). Since there is no existing dataset exclusively for MWEs, this experiment uses a custom dataset based on the three types of MWEs mentioned earlier. The translation software tested includes Google Translate, Microsoft Bing Translator, and Baidu Translate. The results are shown in Table 8. For example, when testing Google Translate with a threshold of t=0.7 and focusing on Noun + Noun MWEs, DMWE detected 42 suspicious groups, 35 of which were confirmed as errors, yielding an precision of 83.3%. When the threshold was lowered to t=0.65, DMWE detected 28 suspicious groups, of which 25 were identified as errors, increasing the precision to 89.3%. These results indicate that, under the default threshold, DMWE is slightly more accurate than DCS in most cases.

The primary difference between the two methods lies in their focus: DMWE targets the translation of multi-word expressions, while DCS takes a broader approach, analyzing overall sentence structure. As a result, the two methods complement each other, enabling the identification of more translation errors and improving the overall precision and comprehensiveness of the testing process.

5 Related Work

In machine translation testing, various attributes, such as accuracy and robustness, are typically considered. Some methods primarily focus on the robustness of machine translation, examining whether the system is affected by minor errors or noise in the input sentences. Other methods emphasize testing the accuracy of machine translation. Methods for testing the accuracy of machine translation can be divided into two categories: metamorphic testing-based methods and differential testing-based methods.

Metamorphic testing-based approaches define metamorphic relations to describe the dependencies between changes in system inputs and their corresponding outputs. In recent years, Pesu et al. (Pesu et al., 2018) introduced the first metamorphic testing-based method for machine translation, using English as the source language and covering eight target languages. Since then, several metamorphic testing methods for machine trans-

Table 8: Precision Comparison between DMWE and DCS

Type	Method	DMWE		DCS
	Threshold	t=0.7	t=0.65	/
Noun+Noun	Google	83.3%（35/42）	89.3%（25/28）	79.4%（228/287）
	Bing	85.1%（40/47）	88.9%）24/27）	66.7%（265/335）
	Baidu	80.6%（25/31）	87.5%（14/16）	66.2%（319/401）
Adjective + Noun	Google	82.1%（78/95）	91.0%（51/56）	79.6%（214/269）
	Bing	80.2%（73/91）	89.9%（62/69）	80.2%（251/313）
	Baidu	79.3%（46/58）	85.7%（30/35）	80.7%（292/362）
Verb + Noun	Google	81.4%（114/140）	90.4%（85/94）	80.1%（233/291）
	Bing	80.8%（126/156）	90.6%（96/106）	78.6%（250/318）
	Baidu	81.8%（85/104）	88.5%（54/61）	79.0%（286/362）

lation have been developed. For example, He et al. (He et al., 2020) proposed a novel technique called Structural Invariance Testing (SIT), based on the premise that translations of similar source sentences should exhibit similar sentence structures. Furthermore, He et al. (He et al., 2021) introduced the concept of Relative Translation Invariance (RTI), which posits that translations of a text in different contexts should remain similar. By evaluating translations of text pairs sharing the same RTI, they assessed translation similarity to verify accuracy. Ji et al. (Ji et al., 2021) presented Constituent Invariance Testing (CIT), a technique that employs constituent parsing trees to represent sentence structures. Through an efficient data augmentation approach, CIT generates multiple new sentences from a single sentence. Gupta et al. (Gupta, 2020) proposed a testing method called PatInv, based on the principle that sentences with distinct meanings should not yield identical translations. If two sentences with different meanings produce the same translation, this could indicate an error. Cao et al. (Cao et al., 2022) introduced SemMT, an automated testing approach that relies on semantic similarity checking. SemMT performs back-translation and captures semantic similarity using a set of regular expression-based metrics to detect potential issues. Sun et al. (Sun et al., 2020) developed TransRepair, a novel method that combines mutation testing with metamorphic testing to identify inconsistent defects. TransRepair generates mutated sentences by replacing words with contextually similar ones, expecting the translations of the original and mutated sentences to remain consistent despite the word changes. Additionally, Sun et al. (Sun et al., 2022) introduced CAT, a method focused on identifying word substitutions with controlled effects. Finally, Zhang et al. (Zhang et al., 2024) proposed a syntax tree pruning-based metamorphic testing method, hypothesizing that pruned sentences should maintain similar important semantics compared to the original sentences.

Differential testing-based methods (McKeeman, 1998) detect errors by determining whether the outputs for the same input are consistent across implementations based on the same specification. The latest approach in this category is DCS, which introduces a compositional semantics-based differential testing method for evaluating and detecting translation defects and semantic deviations in machine translation. DCS works by decomposing sentences into core and adjunct parts, translating them separately, and identifying errors in the translation process.

6 Conclusion

We propose a Differential Testing Method for Machine Translation of Multi-word Expressions(DMWE). This method targets multi-word expressions, evaluating the accuracy of phrase translations in machine translation through differential testing, thereby enabling more precise identification of translation issues. The experiments assessed DMWE's accuracy, its ability to identify translation errors, its false-positive rate, and its overall effectiveness. The results demonstrate that DMWE offers significant advantages in testing multi-word expressions, providing an innovative approach and methodology for phrase-hierarchy machine translation testing.

Limitations

This method focuses on multi-word expressions as the research target to study phrase translation and identify translation issues. However, multi-word expressions are only a specific form of phrases. To

assess the translation of all types of phrases, the method needs further improvement.

The method has only been tested on English-Chinese translation pairs and does not yet cover other language pairs. As research progresses, DMWE needs to be extended to include additional language pairs.

The accuracy of the method's results is affected by polysemy and word alignment issues. To improve the accuracy further, optimization is required for these two problems, such as enhancing semantic similarity calculations and word alignment algorithms to reduce misjudgments caused by these factors.

Acknowledgments

This work is supported by the Fundamental Research Funds for the Central Universities (No. NT2024020 and No. NJ2024030), and the Open Fund of the State Key Laboratory for Novel Software Technology (No. KFKT2024B27).

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Peter F. Brown, John Cocke, Stephen Della Pietra, Vincent J. Della Pietra, Frederick Jelinek, John D. Lafferty, Robert L. Mercer, and Paul S. Roossin. 1990. A statistical approach to machine translation. *Comput. Linguistics*, 16(2):79–85.

Peter F. Brown, Stephen Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Comput. Linguistics*, 19(2):263–311.

Jialun Cao, Meiziniu Li, Yeting Li, Ming Wen, Shing-Chi Cheung, and Haiming Chen. 2022. Semmt: A semantic-based testing approach for machine translation systems. *ACM Trans. Softw. Eng. Methodol.*, 31(2):34e:1–34e:36.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Zi-Yi Dou and Graham Neubig. 2021. Word alignment by fine-tuning embeddings on parallel corpora. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, EACL 2021, Online, April 19 - 23, 2021*, pages 2112–2128. Association for Computational Linguistics.

Jonas Gehring, Michael Auli, David Grangier, and Yann N. Dauphin. 2017. A convolutional encoder model for neural machine translation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 123–135. Association for Computational Linguistics.

Shashij Gupta. 2020. Machine translation testing via pathological invariance. In *ICSE '20: 42nd International Conference on Software Engineering, Companion Volume, Seoul, South Korea, 27 June - 19 July, 2020*, pages 107–109. ACM.

Hany Hassan, Anthony Aue, Chang Chen, Vishal Chowdhary, Jonathan Clark, Christian Federmann, Xuedong Huang, Marcin Junczys-Dowmunt, William Lewis, Mu Li, Shujie Liu, Tie-Yan Liu, Renqian Luo, Arul Menezes, Tao Qin, Frank Seide, Xu Tan, Fei Tian, Lijun Wu, Shuangzhi Wu, Yingce Xia, Dongdong Zhang, Zhirui Zhang, and Ming Zhou. 2018. Achieving human parity on automatic chinese to english news translation. *CoRR*, abs/1803.05567.

Kim M. Hazelwood, Sarah Bird, David M. Brooks, Soumith Chintala, Utku Diril, Dmytro Dzhulgakov, Mohamed Fawzy, Bill Jia, Yangqing Jia, Aditya Kalro, James Law, Kevin Lee, Jason Lu, Pieter Noordhuis, Misha Smelyanskiy, Liang Xiong, and Xiaodong Wang. 2018. Applied machine learning at facebook: A datacenter infrastructure perspective. In *IEEE International Symposium on High Performance Computer Architecture, HPCA 2018, Vienna, Austria, February 24-28, 2018*, pages 620–629. IEEE Computer Society.

Pinjia He, Clara Meister, and Zhendong Su. 2020. Structure-invariant testing for machine translation. In *ICSE '20: 42nd International Conference on Software Engineering, Seoul, South Korea, 27 June - 19 July, 2020*, pages 961–973. ACM.

Pinjia He, Clara Meister, and Zhendong Su. 2021. Testing machine translation via referential transparency. In *43rd IEEE/ACM International Conference on Software Engineering, ICSE 2021, Madrid, Spain, 22-30 May 2021*, pages 410–422. IEEE.

Pin Ji, Yang Feng, Jia Liu, Zhihong Zhao, and Baowen Xu. 2021. Automated testing for machine translation via constituency invariance. In *36th IEEE/ACM International Conference on Automated Software Engineering, ASE 2021, Melbourne, Australia, November 15-19, 2021*, pages 468–479. IEEE.

Su Nam Kim. 2008. *Statistical modeling of multiword expressions*. Ph.D. thesis, Ph. D. thesis, University of Melbourne, Melbourne.

Sangmin-Michelle Lee. 2023. The effectiveness of machine translation in foreign language education: a systematic review and meta-analysis. *Computer Assisted Language Learning*, 36(1-2):103–125.

Qianchu Liu, Diana McCarthy, Ivan Vulic, and Anna Korhonen. 2019. Investigating cross-lingual alignment methods for contextualized embeddings with token-level evaluation. In *Proceedings of the 23rd Conference on Computational Natural Language Learning, CoNLL 2019, Hong Kong, China, November 3-4, 2019*, pages 33–43. Association for Computational Linguistics.

Sahil Manchanda and Galina Grunin. 2020. Domain informed neural machine translation: Developing translation services for healthcare enterprise. In *Proceedings of the 22nd Annual Conference of the European Association for Machine Translation, EAMT 2020, Lisboa, Portugal, November 3-5, 2020*, pages 255–261. European Association for Machine Translation.

William M. McKeeman. 1998. Differential testing for software. *Digit. Tech. J.*, 10(1):100–107.

Daniel Pesu, Zhi Quan Zhou, Jingfeng Zhen, and Dave Towey. 2018. A monte carlo method for metamorphic testing of machine translation services. In *3rd IEEE/ACM International Workshop on Metamorphic Testing, MET 2018, Gothenburg, Sweden, May 27, 2018*, pages 38–45. ACM.

Masoud Jalili Sabet, Philipp Dufter, François Yvon, and Hinrich Schütze. 2020. Simalign: High quality word alignments without parallel training data using static and contextualized embeddings. In *Findings of the Association for Computational Linguistics: EMNLP 2020, Online Event, 16-20 November 2020*, volume EMNLP 2020 of *Findings of ACL*, pages 1627–1643. Association for Computational Linguistics.

Ivan A. Sag, Timothy Baldwin, Francis Bond, Ann A. Copestake, and Dan Flickinger. 2002. Multiword expressions: A pain in the neck for NLP. In *Computational Linguistics and Intelligent Text Processing, Third International Conference, CICLing 2002, Mexico City, Mexico, February 17-23, 2002, Proceedings*, volume 2276 of *Lecture Notes in Computer Science*, pages 1–15. Springer.

Agata Savary, Chérifa Ben Khelil, Carlos Ramisch, Voula Giouli, Verginica Barbu Mititelu, Najet Hadj Mohamed, Cvetana Krstev, Chaya Liebeskind, Hongzhi Xu, Sara Stymne, Tunga Güngör, Thomas Pickard, Bruno Guillaume, Eduard Bejcek, Archna Bhatia, Marie Candito, Polona Gantar, Uxoa Iñurrieta Urmeneta, Albert Gatt, Jolanta Kovalevskaite, Timm Lichte, Nikola Ljubesic, Johanna Monti, Carla Parra Escartín, Mehrnoush Shamsfard, Ivelina Stoyanova, Veronika Vincze, and Abigail Walsh. 2023. PARSEME corpus release 1.3. In *Proceedings of the 19th Workshop on Multiword Expressions, MWE@EACL 2023, Dubrovnik, Croatia, May 6, 2023*, pages 24–35. Association for Computational Linguistics.

Agata Savary, Carlos Ramisch, Silvio Cordeiro, Federico Sangati, Veronika Vincze, Behrang QasemiZadeh, Marie Candito, Fabienne Cap, Voula Giouli, Ivelina Stoyanova, and Antoine Doucet. 2017. The PARSEME shared task on automatic identification of verbal multiword expressions. In *Proceedings of the 13th Workshop on Multiword Expressions, MWE@EACL 2017, Valencia, Spain, April 4, 2017*, pages 31–47. Association for Computational Linguistics.

Zeyu Sun, Jie M. Zhang, Mark Harman, Mike Papadakis, and Lu Zhang. 2020. Automatic testing and improvement of machine translation. In *ICSE '20: 42nd International Conference on Software Engineering, Seoul, South Korea, 27 June - 19 July, 2020*, pages 974–985. ACM.

Zeyu Sun, Jie M. Zhang, Yingfei Xiong, Mark Harman, Mike Papadakis, and Lu Zhang. 2022. Improving machine translation systems via isotopic replacement. In *44th IEEE/ACM 44th International Conference on Software Engineering, ICSE 2022, Pittsburgh, PA, USA, May 25-27, 2022*, pages 1181–1192. ACM.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 3104–3112.

Liling Tan, Maggie Yundi Li, and Stanley Kok. 2020. E-commerce product categorization via machine translation. *ACM Trans. Manag. Inf. Syst.*, 11(3):11:1–11:14.

Liang Tian, Derek F. Wong, Lidia S. Chao, Paulo Quaresma, Francisco Oliveira, and Lu Yi. 2014. Um-corpus: A large english-chinese parallel corpus for statistical machine translation. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation, LREC 2014, Reykjavik, Iceland, May 26-31, 2014*, pages 1837–1842. European Language Resources Association (ELRA).

Nan Yang, Shujie Liu, Mu Li, Ming Zhou, and Nenghai Yu. 2013. Word alignment modeling with context dependent deep neural network. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, ACL 2013, 4-9 August 2013, Sofia, Bulgaria, Volume 1: Long Papers*, pages 166–175. The Association for Computer Linguistics.

Biao Zhang, Deyi Xiong, and Jinsong Su. 2018. Accelerating neural transformer via an average attention network. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics,*

ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers, pages 1789–1798. Association for Computational Linguistics.

Quanjun Zhang, Juan Zhai, Chunrong Fang, Jiawei Liu, Weisong Sun, Haichuan Hu, and Qingyu Wang. 2024. Machine translation testing via syntactic tree pruning. *ACM Trans. Softw. Eng. Methodol.*, 33(5):125:1–125:39.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with BERT. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

A Appendix

A.1 Experiment Setup

Table 9: Multi-word expression datasets

Type	# of words/ sentence	Average # of words/sentence	# of words	
			Total	Distinct
Noun+Noun	10-30	20.99	31449	9018
Adjective + Noun	10-30	21.16	32267	8904
Verb + Noun	10-30	20.17	27669	7086

The detailed information of the dataset is shown in Table 9. For example, the average number of words in sentences containing Noun + Noun type multi-word expressions is 20.99, with a total of 31,449 words and 9,018 unique words. To avoid repeated testing, each multi-word expression appears only once in the dataset.

A.2 False Positive

False positives are an inevitable phenomenon in machine translation. During the experiment, several false positive cases were encountered. The causes of false positives are as follows: (1) Word Polysemy: Multi-word expressions may have multiple meanings or their translations may have different ways of expression. BERTScore was used in similarity judgment, but it still has limitations and cannot perfectly match all synonyms. If the similarity between two translation results is low, they may be mistakenly judged as incorrect translations. (2) Word Alignment: AWESOME was used in the word alignment process, which generally performs well, but still has some shortcomings. If the multi-word expression is not correctly aligned with its corresponding position in the target language, the translation result may be mistakenly judged as an error.