博士学位論文

論文題目　Application-oriented Machine Translation: Design and Evaluation（実応用を志向した機械翻訳の設計と評価）

提　出　者　東北大学大学院情報科学研究科

システム情報科学　専　攻

学籍番号　C0ID2001

氏　名　阿部 香央莉

Application-oriented Machine Translation: Design and Evaluation

実応用を志向した機械翻訳システムの設計と評価

Kaori Abe

Graduate School of Information Sciences
Tohoku University

This dissertation is submitted for the degree of
Doctor of Information Science

January 2023

Acknowledgements

本研究の遂行にあたり、たいへん多くの方々にご協力・ご支援をいただきました。ここに、深い感謝の念を表します。

指導教官である乾健太郎教授には、学部３年からの長い期間、研究のあり方および研究者としてのあり方や社会との関わり方など、多くのご指導をいただきました。また、研究室の内外問わず様々な方々との出会いおよび自身の成長の場をいただき、今後の自分を形づくる上でかけがえのない財産となる多くのものをいただきました。また、学部の頃から経済的に余裕がない自分の境遇を常に気遣っていただき、積極的に学内のリサーチ・アシスタントの機会をいただくなど、多大なサポートをしていただきました。

ご多忙の中、審査員として本論文を査読してくださいました張山昌論教授、ならびに伊藤彰則教授に深く感謝いたします。

鈴木潤教授には、機械翻訳に関する研究の知識や、実装の方法、また研究の基礎となる考え方など、研究を進める上で必要なたくさんの助言を承りました。また、お忙しい中でも自分自身が関わるもの全てに気を配る姿勢には、大変感銘を受けるとともに強く尊敬の念を抱きました。心より感謝いたします。

東京工業大学の岡崎直観教授、東北大学教育学部の松林優一郎准教授、理化学研究所の松田耕史様、東北大学助教の横井祥助教、愛媛大学の梶原智之助教には、研究の遂行において、数多くの貴重な助言や心身を含めたサポートなど、手厚いご助力をいただきました。深く感謝いたします。

東北大学乾研究室の皆様、ならびに学会等でご縁のあった皆様には、議論を通して様々な助言やコメントをいただき、研究生活を支えていただきました。皆様との温かな交流を経て、無事このように学位論文を大成することができました。心より、感謝の意を表します。

最後に、日々私を応援し支え続けてくれた友人、遠い青森の地から暖かく見守り続けてくれている家族に、深く感謝いたします。

Abstract

Machine translation (MT) performance has improved significantly in recent years. With this improvement, the use of MT in the real world has also expanded. For example, MT users can enjoy foreign content on the web or online platforms such as SNS and YouTube or offer their content to potential foreign customers. However, considering the real-world uses of MT, we are faced with two problems caused by the current MT: (1) the need for customizable MT to the desires of individual users and (2) the potential to spread misunderstanding and miscommunications with errors.

The final goal of this study is to develop an individually customizable MT that does not cause miscommunication for real-world applications. In this study, we show two possible solutions to these two problems, (i) MT design and (ii) robust evaluation, respectively. We present actual works based on these two solutions in this paper.

We describe some possible designs of application-oriented MT to combine multiple optional architectures for real-world problems. Particularly, we show a multi-dialect translation system that translates low-resource and diverse Japanese dialects among examples of the MT design.

For the robust evaluation, we explore appropriate evaluation framework for two specific subjects: semantic similarity evaluation and terminology-focused evaluation. As for the semantic similarity, we analyzed what factors are affected with respect to the currently proposed automatic semantic similarity metrics. We argue that the current metrics are affected by differences in domain and similarity granularity, and that we need to consider the benchmark for exploring a more rigorous similarity evaluation framework. Regarding terminology-focused evaluation, we established a new task as an evaluation framework for constrained models focused on terminology in neural MT. Through analyses of the task results, we discussed the validity of a simple proposed automatic metric and the importance of rigor in human evaluation.

The MT design considering user-individual problem will further improve MT familiarity and convenience and enable support for real-world cross-lingual communications. Further, the robust evaluation is necessary to reduce misunderstandings that can occur in the facilitated cross-lingual communications. We conclude that the development of application-oriented MT requires the exploration of appropriate MT design and robust evaluation as described in this study, which will lead to MT helping people lead sound livelihoods.

Table of contents

List of figures

List of tables

Chapter 1

Introduction

Machine translation (MT) is one of the fields actively studied in Natural Language Processing (NLP). The quality of MT has improved in recent years, which has made the use of MT widespread in our daily lives and industry. In daily life, the MT services on web browsers and SNS have been implemented to encourage users to enjoy enormous foreign language content. Additionally, automatic translated subtitle generation combining MT and speech processing has also been tried on YouTube and online meeting tools such as Zoom. On the other hand, MT also makes providers of contents such as articles and videos to provide pre-translated content to reach foreign users. Also in industry, the introduction of MT technology has been active. The MT market has been and will continue to expand domestically and internationally because of the improvement of the MT performance[1]. Thus, the expanded use of MT systems has made MT an indispensable part of our lives.

However, two issues are critical in the expansion of MT. One point is the need for customizable MT systems for individual problems. Some MT services that provide a customized MT system for individual enterprises are attracting attention, while there are general-purpose MT such as Google Translate and DeepL. In addition, even in general-purpose MT such as DeepL, user-customizable features such as glossaries are implemented. Thus, the actual use of an MT system in the real world needs to be customized to suit the user's preferences and cases. Another critical issue is that the MT output still contains mistranslations that amateurs need help identifying. In the MT research community, some researchers Läubli et al. (2020); Toral et al. (2018) demonstrated that crowd workers' evaluations are insufficient for MT quality evaluation and professional translator-level evaluations are necessary. This problem becomes severe in real-world situations; because it means that most people would not be able to recognize mistranslations in the real-world daily use of MT. This has the possibility of

[1] https://www.gminsights.com/industry-analysis/machine-translation-market-size

leading to misunderstandings and communication errors. In industry, such mistranslations also have been making additional processes, such as post-editing by professional translators for MT output. Thus, in real-world applications of MT, it is necessary to consider MT that is (1) customizable and (2) does not induce miscommunication.

This study attempts to tackle these two critical issues using MT **design** and robust **evaluation**, respectively. We present examples of these approaches with concrete real-world applications. As an example of MT design, we describe an MT system that can treat low-resource and diverse dialects to show the effectiveness and importance of customizing MT to individual problems. For robust evaluation, we focus on two subjects, evaluation for semantically inappropriate examples and inappropriate terminology in the current MT outputs. We analyze the semantic similarity evaluation metrics proposed in the current NLP field, and explore the problems in the current semantic similarity evaluation for the semantically inappropriate examples. In addition, we establish a new translation task focusing on evaluating terminology as an evaluation framework to detect inappropriate terminology.

1.1 Research Issues

In this thesis, we address the following research issues:

- **What is the appropriate design for real-world situations?:** In the MT field, various optional architectures have been proposed to deal with individual problem such as inappropriate terminology. However, how MT should be designed for real-world problems involving a combination of various phenomena has yet to be organized.
- **What are the possible problems with the current semantic similarity evaluation?:** Several metrics that consider semantic similarity instead of BLEU have been proposed as current automatic metrics, but it needs to be clearly analyzed what factors they are susceptible to and what problems they contain.
- **Lack of an evaluation framework to facilitate proper terminology translation:** One problem that undermines the current MT's quality is inappropriate terminology translation. Several models have been proposed to address this, but no clearly shared evaluation framework exists.

1.2 Contributions

This thesis makes the following contributions:

- **Proposed a customized MT that allows efficient learning for a real-world situation (e.g., low-resource Dialect Translation)**: we presented some designs for each scene using MT and our customized MT system that treats dialects to show the effectiveness and importance of customizing MT to individual problems.
- **Comprehensive analysis for current semantic similarity metrics for MT**: we analyzed the generic semantic similarity evaluation metrics and MT-specific similarity metrics proposed in the current NLP field. We explored the problems in the current semantic similarity evaluation.
- **Development of terminology translation evaluation framework**: we established a new translation task focusing on evaluating terminology as an evaluation framework to detect inappropriate terminology examples. We identified additional considerations in the current evaluation framework based on the task results.

1.3 Thesis Overview

An overview of this paper is given as follows:

Chapter 2: Background. we introduce the background of two approaches (MT design and robust evaluation) and related work.

Chapter 3: Japanese Multi-Dialect Translation as Use Case. we introduce an example of designing a real application-oriented MT system for low-resource and diverse Japanese dialects.

Chapter 4: Comprehensive Analysis of Semantic metrics for MT. we discuss the characteristics and biases of current semantic similarity metrics based on performance in two tasks, the semantic textual similarity (STS) task, which is a generic semantic similarity benchmark, and MT evaluation task, which is one of the application-oriented semantic similarity tasks.

Chapter 5: Proposal of Terminology-focused Translation Task. we introduce the terminology-focused MT task in the MT workshop and highlight the problems for evaluating terminology translation that need to be addressed.

Chapter 6: Conclusions. we summarize two approaches, MT design and robust evaluation, that should be considered for application-oriented MT to facilitate cross-lingual communications in the real world.

Chapter 2

Background

2.1 Design: Customizable MT with Optional Architecture

A neural MT (NMT) model, a sequence-to-sequence model with neural networks, has been adopted as a typical architecture in MT. Transitioning from Long-Short Term Memory (Hochreiter and Schmidhuber, 1997) to Transformers (Vaswani et al., 2017) allowed these models to achieve higher translation performance. However, even with the current NMT models using Transformers, translation quality issues are still present, as described in Chapter 1 when considering real-world applications. Many attempts have been made to address these problems by applying a (relatively simple) optional architecture to the general architecture. Indeed, fairseq,[1] an open-source tool for sequence-to-sequence models such as MT, provides a choice of sub-words and characters as the encoding method and constrained decoding as the decoding method. Here, we introduce examples of existing optional architectures that are useful for considering application-oriented MT.

2.1.1 Option Architectures

Encoding Method In general, NMT models have a static list of vocabularies. The encoding process in NMT involves tokenization, which is splitting raw texts into a set of prescribed processing units named tokens. Tokens that do not in the vocabulary list are processed as an unknown token as <*unk*>. As a unit of the vocabulary, words are most traditionally used. However, in the case of word units, several words that do not appear during training (e.g., rare words and domain-specific words) are treated as unknown tokens, which can deteriorate translation performance. So, using a smaller unit as the optional architecture for tokeniza-

[1] `https://github.com/facebookresearch/fairseq`

tion, subwords (Kudo and Richardson, 2018; Sennrich et al., 2016) and characters (Gao et al., 2020; Lee et al., 2017), has been shown to improve performance by alleviating the appearance of unknown tokens. In addition, ideograms, a set of characters that express their meaning, have been used in some languages such as Chinese (*kanji*). To uniformly process these different character types, recent research has been conducted that treats byte sequences (Wang et al., 2020) and visual features (Salesky et al., 2021) as the optional architectures.

Additional Information for Input Representations In recent years, MT has been required to support low-resource languages. A few/zero-shot translation using multilingual MT is attracting attention to solve this problem. Multilingual NMT is a translator that can translate multiple language pairs by learning multiple language pairs simultaneously. Currently, the most common architecture for multilingual NMT is the method proposed by (Johnson et al., 2016), which inserts a special token for each language at the beginning of the input sequence. The special token is used to learn the features of each language as a vector (named language embedding), which is said to enable processing that corresponds to each language better. There are also many examples of applications in which the special token is regarded as "input meta information" and applied as a different representation. Such representations according to special tokens are also used as augmented data signal (Caswell et al., 2019). The special tokens as the augmented data signal are helpful in input data augmentation, which is a promising method for improving MT performance depending on data amounts. In addition to individual special tokens such as language embedding, there is also an attempt to realize context-aware translation by adding context information to input using [SEP] tokens. This context-aware translation is effective for high-context languages such as Japanese, where context information is often omitted, and for colloquial-style texts such as dialogues. Thus, a simple method of additional input information can be considered an optional architecture.

Decoding Method During decoding, we can adopt optional architectures as well as encoding. Constrained models, which output specified phrases or words, have been proposed as an architecture for terminology translation. There are two types of constrained models: the soft constrained method (Chen et al., 2020; Song et al., 2019) that adds constraints by augmentation of training data, and the hard constrained method (Arthur et al., 2016; Hokamp and Liu, 2017; Post and Vilar, 2018) that forcibly adds constraints by signals during decoding.

2.1.2 Examples of MT Design

When we would like to apply MT to individual problems in the real world, we can combine the above optional architectures to suit the problem. This subsection will give several examples of specific real-world problems and discuss possible MT designs for them.

As a first example, we consider the translation of patents of new medicine. Patent translation has been the subject of several studies in the research community (Chu et al., 2018; Goto et al., 2013; Morishita et al., 2022) and is one of the leading applications of MT in the industry. In this case, users would like to convey their patents worldwide. However, there is a regulation that a patent obtained in Japan applies only within Japan. Therefore, in order to protect patents around the world, it is necessary to translate the patents into various foreign languages. In addition, this patent contains information about medicine; in other words, it belongs to a specific biomedical domain, which contains many terminologies. To address these issues, we can adopt three options are as follows: (1) multilingual translation with special tokens, (2) input data augmentation for in-domain data, and (3) constrained models. Multilingual translation using special tokens enables translation into a variety of foreign languages. In addition, to improve the performance of translations in the biomedical domain, data augmentation using augmented data signals or constrained models that has the effect of outputting a specific terminology can be introduced.

Next, as another example, we consider cross-lingual chat support. In recent years, automatic translation tools have been introduced on online platforms such as SNS or YouTube, leading to more interaction with foreign language users. Also, automatic interpretation has been introduced in online meeting tools such as Google Meets and Zoom. Thus, cross-lingual chats via MT have become an everyday occurrence, and several types of research on chat translation have been conducted (Farajian et al., 2020; Li et al., 2022; Liang et al., 2021). However, unlike the translation of official documents, MT systems must cope with colloquial-style expressions, such as omitting some parts of the texts. In response to this, the following options can be introduced: (1) context-aware translation using [SEP] tokens and (2) data augmentation for in-style data. Context-aware translation enables supplementing omitted parts in the source chat text with contextual information. In addition, data augmentation with similarly styled data may effectively improve the translation performance into a colloquial style.

Thus, there are various possible designs for MT systems combining with appropriate optional architectures for each problem. In Chapter 3, as one example, we propose an MT system for Japanese dialects, which needs to handle diverse dialects in low-resource situations.

2.2 Evaluation: Necessity of Robust Evaluation

Another issue of the current MT is the validity of the evaluation. Generally, there are two kinds, automatic and human evaluation, in MT evaluation. The human evaluation with professional translators is the most desirable; however, it will be expensive to conduct the evaluations regularly to develop reliable MT systems. So much research has been conducted to realize inexpensive and appropriate automatic evaluation metrics as an alternative to the human evaluation. In the MT community, Metrics task [2] has been held at the international MT workshop since 2008 to compete for the performance of the automatic metrics. The results of this task have often led to discussions about the reliability of the MT evaluation framework (Bojar et al., 2017; Ma et al., 2019).

One of the major problems for the MT evaluation in recent years is that the improvement of MT performances has made it impossible for amateur annotators to evaluate the MT quality properly (Läubli et al., 2020). This problem becomes a more severe issue when considering real-world use. When MT outputs include mistranslations, users can only do the following ways because they cannot recognize the errors on their own: to use the mistranslated texts as it is or to hire experts to post-edit the texts. In the former case, ignoring mistranslations may lead to misunderstandings in cross-lingual communication. The latter also causes the problem of costs eventually, despite introducing MT as a low-cost way compared to human translations. Thus, we need to consider more rigorous evaluation for MT systems as the use of MT expands in real-world applications.

For more reliable MT in real-world applications, we analyze problems of the automatic evaluation for capturing semantic similarity in Chapter 4 and establish a new MT task specialized for terminology evaluation in Chapter 5. In the following, we briefly describe the background knowledge of the automatic and human evaluation helps to understand those Chapters.

2.2.1 Automatic Evaluation

N-gram based metrics: BLEU, chrF The most commonly used automatic evaluation metric in MT is BLEU (Papineni et al., 2002a), which is based on the n-gram overlap of tokens. Recently, chrF (Popović, 2015, 2017), the F-score of character n-gram overlap, has also been frequently used as a baseline because it is known to correlate higher with human evaluation than BLEU. However, it has long been pointed out that the n-gram-based metrics do not give good scores for semantically acceptable expressions (Banerjee and Lavie, 2005; Zhang et al., 2020a). In addition, Ma et al. (2019) pointed out that the current MT automatic

[2] `https://wmt-metrics-task.github.io/`

evaluation at the segment level[3] does not correlate with the human evaluation. Especially in recent years, some papers strongly argue that BLEU should not be used (Kocmi et al., 2021; Mathur et al., 2020).

Semantic metrics: METEOR, RUSE, and BERTScore In contrast to n-gram-based metrics, METEOR (Banerjee and Lavie, 2005) was proposed as one of the first metrics to capture semantic similarity. However, it spread only a little, mainly because of the cost of adapting it to other languages. With the advent of the Universal Sentence Encoder (USE) (Cer et al., 2018), which was evaluated on a generic task of semantic similarity in NLP, many sentence encoder models have applied for several NLP applications such as MT. Specifically, RUSE (Shimanaka et al., 2018), which applied the sentence encoder to the MT evaluation, became the top system in the WMT18 Metrics task. Recently, BERTScore (Zhang et al., 2020a), a model using Bi-directional Encoder Representations from Transformer (BERT; Devlin et al., 2019) was proposed as the automatic semantic metric. BERTScore is widely used to evaluate semantic similarity in NLG as a whole (Mille et al., 2021), because its performance is significantly higher than that of conventional methods that use BERT as-is, and it does not require fine-tuning. In addition, learned semantic metrics such as BLEURT (Sellam et al., 2020) and COMET (Rei et al., 2020) have earned state-of-the-art in the current MT Metrics task.

2.2.2 Human Evaluation

Likert scale (adequacy and fluency) The most basic human evaluation is a five-point scale based on the two aspects, adequacy and fluency, proposed by (Callison-Burch et al., 2007). This five-point scale is also used in the JPO translation evaluation criteria[4] and the human evaluation of Workshop on Asian Translation (WAT) (Nakazawa et al., 2022, 2021).

Direct Assesment Currently, the most used human evaluation is the direct assessment (DA) framework from Graham et al. (2015). It has 100 evaluation stages, including source-based DA (Cettolo et al., 2017; Federmann, 2018), which evaluates by comparing source and translation, and reference-based DA, which evaluates by comparing reference and translation.

[3] In general, one segment corresponds to one sentence in the datasets of the Metric task.

[4] https://www.jpo.go.jp/system/laws/sesaku/kikaihonyaku/tokkyohonyaku_hyouka.html

Multi-dimensional Quality Metrics (MQM) The overview paper in the recent Metric task (Ma et al., 2019) argued that the current automatic metrics are incapable of comparing superior MT systems at the segment level. Their focus was on the issue of automatic metrics; however, Mathur et al. (2020) pointed out the possibility that there is no difference between the superior MT systems not only in the automatic metrics but also in the current human evaluation framework. At the same time, Läubli et al. (2020) argued that it is a misconception based on the low quality of the human evaluation that the current MT performance reaches the professional translator level. In light of these studies, the MT community recognizes the need to consider the validity of the human and automatic evaluation. As an alternative to the DA framework, Freitag et al. (2021) has proposed Multi-dimensional Quality Metrics (MQM)[5] as a new criterion for the human evaluation. They proposed a method to annotate multi-dimensional errors (e.g., Accuracy, Style, and Terminology.) in three levels: Major, Minor, and None. They showed that the rankings of MT systems differed significantly between the two human evaluation frameworks (i.e., DA and MQM) and that professional human translation still be preferred over MT in the MQM framework. They also presented that automatic metrics learned datasets with the MQM annotations outperformed human amateur annotators. This result suggests that considering a proper human evaluation framework also facilitates the proposal of better automatic metrics.

[5] `https://themqm.org/`

Chapter 3

Design: Japanese Multi-Dialect Translation as Use Case

3.1 Introduction

With the use of automated personal assistants (e.g., Apple's Siri, Google Assistant, or Microsoft Cortana) and smart speakers (e.g., Amazon Alexa or Google Home) becoming increasingly widespread, the demand to bridge the gap between the standard form of a given language and its dialects has also enlarged. The importance of dealing with dialects is particularly evident in a rapidly aging society, like that in Japan, where older people use them extensively.[1]

To address this issue, we consider a system for machine translation (MT) between Japanese dialects and standard Japanese. If such a system can yield correct dialect-to-standard translations, then other natural language processing systems (e.g., information retrieval or semantic analysis) that adopt standard Japanese as the input could also be applied to dialects. In addition, if a standard-to-dialect translation system becomes available, then smart speakers could respond to the native speakers of a dialect using that dialect. We believe that sympathetic interactions of this type might lead to such systems gaining more widespread acceptance of in the Japanese society.

In this paper, we present a multi-dialect neural MT (NMT) system tailored to Japanese. Specifically, we employ *kana*, a Japanese phonetic lettering system, to provide the basic units in the encoder–decoder framework to avoid the followings: ambiguity in converting *kana* to *kanji* (characters in the Japanese writing system), difficulties in identifying word boundaries

[1] 本研究は既発表論文「Multi-dialect Neural Machine Translation for 48 Low-resource Japanese Dialects」『自然言語処理』27 巻 4 号 (CC BY 4.0) に基づく.

especially for dialects, and data sparseness problems due to dealing with numerous words originating from different dialects. Because Japanese dialects almost always use the same word order as standard Japanese, we employ *bunsetsu* (a Japanese phrase unit) as a unit of sequences, instead of a sentence, which is more commonly used in NMT.

One issue for Japanese dialects is the lack of training data. To deal with this, we build a unified NMT model covering multiple dialects, inspired by the studies on multilingual NMT (Johnson et al., 2016). This approach utilizes *dialect embeddings*, i.e., vector representations of Japanese dialects, to inform the model of the input dialect. An interesting by-product of this approach is that the dialect embeddings that the system learns illustrate the difference between different dialect types from different geographical areas. In addition, we present an example of using these dialect embeddings for dialectometry (Guggilla, 2016; Kumagai, 2016; Nerbonne and Kretzschmar, 2011; Rama and Çöltekin, 2016).

Another advantage of adopting a multilingual architecture for multiple related languages is that it can enable gaining knowledge of their lexical and syntactic similarities. For example, Lakew et al. (2018) reported that including several related languages in supervised training data can improve multilingual NMT. Our results confirm the effectiveness of using closely related languages (i.e., Japanese dialects) in multilingual NMT.

3.2 Related Work

Dialectal text is scarcely available because dialects are generally spoken, instead of being written. For this reason, many dialect MT researchers study in low-resource situations (Hassan et al., 2017; Scherrer and Ljubešić, 2016; Zbib et al., 2012).

The use of similar dialects has been found to be helpful in learning translation models for particular dialects. Several previous studies have investigated the characteristics of translation models of closely related dialects (Honnet et al., 2018; Meftouh et al., 2015). For example, Honnet et al. (2018) reported that a character-level NMT model trained on one Swiss-German dialect performed moderately well for translating sentences in closely related dialects.

Therefore, in view of the above, we use multilingual NMT (Johnson et al., 2016) to learn the parameters that encode the knowledge of the shared lexical and syntactic structures of dialects. Some researchers (Arivazhagan et al., 2019; Gu et al., 2018) demonstrated that multilingual NMT could be useful for low-resource language pairs, additionally Lakew et al. (2018) found that a multilingual NMT system trained on multiple related languages showed an improved zero-shot translation performance. We believe that multilingual NMT can be effective for closely related dialects, and can compensate for the lack of translation data for the different associated dialects.

Multilingual NMT can also assist in analyzing the characteristics of each considered language. Östling and Tiedemann (2017) found that clustering the language embeddings learned by a character-level multilingual system provided an illustration of the language families involved. In the light of this, we also examine our dialect embeddings to investigate whether our multi-dialect model can capture the similarities between the dialects (Section 5).

Previous studies reported that character-level statistical machine translation (SMT) using words as translation units is effective for translating between closely related languages (Nakov and Tiedemann, 2012; Scherrer and Ljubešić, 2016). There are two reasons for this: character-level information enables the system to exploit lexical overlaps, whereas using words as translation units takes the advantage of the syntactic overlaps of the related languages. To utilize these overlaps, Pointer Networks (Gülçehre et al., 2016; Vinyals et al., 2015), which can copy some parts of input sequences to output sequences, also seem to be effective for dialect translation. In this study, we conduct an experiment with a simple long short-term memory (LSTM) architecture to train a multilingual NMT model for multiple dialects. Adopting a copy architecture similar to Pointer Networks will be conducted in a future study.

In this study, we present a method of translating between Japanese dialects by combining three ideas: multilingual NMT, character-level NMT, and using base phrases (i.e., *bunsetsu*) as translation units. We believe this enables our approach to fully exploit the similarities among dialects and standard Japanese, even under low-resource settings.

3.3 Data: Japanese Dialect Corpus

Japanese is a dialect-rich language, with dozens of dialects used in everyday conversations in most Japanese regions. They can be characterized in terms of the differences in their content words (vocabulary) and regular phonetic shifts, mostly in their postpositions and suffixes. Specifically, they share most words with standard Japanese, and mostly use common grammatical rules, such as for the word order, syntactic marker categories, and connecting syntactic markers.[2] Some dialects also share the dialect-specific vocabulary. For example, the word, しゃっこい (*shakkoi*, meaning "cold"), is shared among some dialects in the Tohoku region, such as Aomori and Akita.

[2] For details, *Linguistic Atlas of Japan Database* (`https://www.lajdb.org/TOP.html`) published by NINJAL provides an overview of the Japanese dialect distribution.

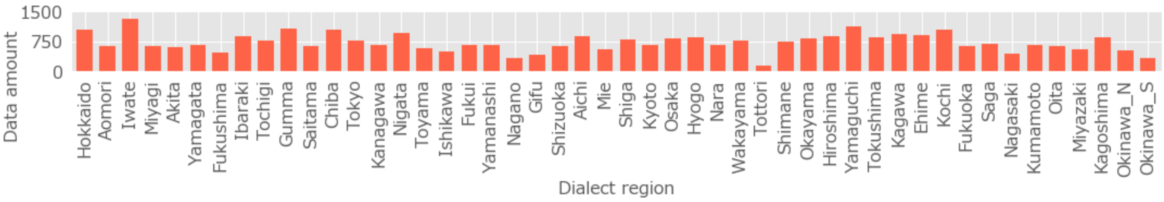


Figure 3.1 Number of sentences in each dialect in *National Dialect Discourse Database Collection*.

In this study, we used a collection of parallel textual data for the dialects and standard Japanese, called as the *National Dialect Discourse Database Collection* (National Institute for Japanese Language and Linguistics, 1980). This corpus includes 48 dialects, one from each of the 47 prefectures and an additional dialect from the Okinawa Prefecture. For each dialect, the texts consist of transcribed 30-minute conversations between two native speakers of that dialect. The total number of dialect sentences (each paired with a translation into standard Japanese) is 34,117. Figure 3.1 shows the number of sentences in each dialect. The amount of each dialect data is less than 1500 sentences and vary.

Japanese texts are generally written in a mix of *kanji* and *kana*; therefore, we converted the *kanji* in the sentences into *kana*, and subsequently, segmented them into *bunsetsu*s.[3] In this study, we used the *bunsetsu* segmentation annotated in the original corpus. After pre-processing, the average sentence lengths were of 14.62 and 15.57 characters for the dialects and the standard Japanese, respectively. The average number of *bunsetsu*s per sentence was 3.42.[4]

3.4 Model: NMT Model with Three Options

Figure 3.2 presents an overview of our network structure of the multi-dialect NMT system. Because our focus is on examining the effectiveness of the multi-dialect NMT and its detailed behavior, rather than on creating a novel translation model, we used OpenNMT (Klein et al., 2017). OpenNMT is a stacking LSTM encoder–decoder model with a multilingual extension similar to that of Johnson's method (Johnson et al., 2016). However, to improve its direct translation accuracy, we introduce the following three modifications.

Dialect labels: Following a previous multilingual NMT study (Johnson et al., 2016), we train the unified model that deals with all the 48 dialects simultaneously using the dialect

[3] This is the smallest Japanese phrase unit, containing a single content word and attached postpositions.

[4] The total number of *bunsetsus* is 116,928.

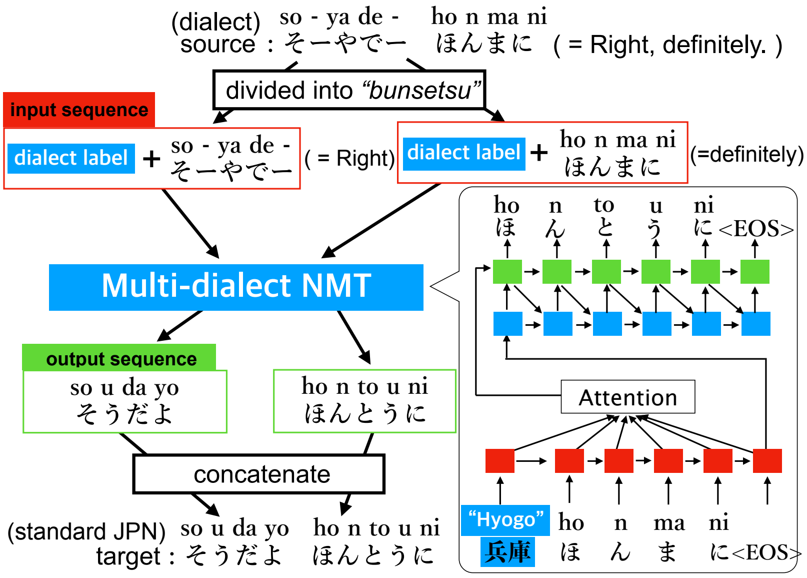

(dialect) source : so - ya de - そーやでー ho n ma ni ほんまに (= Right, definitely.)
divided into *“bunsetsu”*
input sequence
dialect label + so - ya de - そーやでー (= Right)
dialect label + ho n ma ni ほんまに (=definitely)
Multi-dialect NMT
output sequence
so u da yo そうだよ
ho n to u ni ほんとうに
concatenate
(standard JPN) target : so u da yo そうだよ ho n to u ni ほんとうに
ho ほ n ん to と u う ni に <EOS>
Attention
“Hyogo” 兵庫 ho ほ n ん ma ま ni に<EOS>

Figure 3.2 Proposed multi-dialect NMT model.

Table 3.1 Dialect label input order variants. For example, when we translate Aomori dialect into standard language, we test using input sentence that replaced <SRC> with 青森 (*Aomori*) and <TGT> with 標準語 (*Hyoujungo*).

ID	Encoder input order
a	<SRC>, sequence
b	<TGT>, sequence
c	<SRC>, <TGT>, sequence
d	<SRC>, sequence, <TGT>

embeddings including auxiliary dialect labels. Johnson et al. (2016) added a label to the beginning of each sequence to specify the output language. We modify this approach to specify both the input and output dialects of the model, and examine the four different placements for these labels, as listed in Table 3.1.

Syllable-to-syllable translation: As mentioned in Section 3.3, a key to translate between two closely related languages, especially in our case, dialects, is modeling the phonetic correspondences between them. Thus, to consider syllable-level translation rules that may be shared by similar dialects, we defined our translation task as a syllable-to-syllable translation.

We realize syllable-to-syllable translation by representing the inputs and the outputs as *kana* sequences and performing character-based MT. A similar approach was used to normalize Japanese text from Twitter, where the main issue was phonological transliteration (Saito et al., 2017). In our dataset, all the dialect expressions are transcribed using *kana*; however, the standard Japanese translations use a mix of *kanji* and *kana* characters. Therefore, in order to conform to the syllable-to-syllable task, we also convert them into *kana* sequences by automatically analyzing the pronunciation of each *kanji* character and replacing it with the corresponding *kana* sequence.

Translation without distortion: Finally, we attempt to remove the word-order distortion modeling from NMT. In a standard MT, systems adopt a single sentence as the input and yield a translated sentence in an appropriate word order for the target language. However, in the dialect translation, the input and output word orders are mostly the same. To test this, we manually checked 100 randomly-selected sentence pairs from the training set, and found no differences in the ordering (distortion). This fact suggests that we do not require sentence-by-sentence supervision data, because it does not need to learn a distortion model. Based on this intuition, we split each input sentence into base-phrase parts, i.e., *bunsetsu* sequences, translate each chunk from the source to the target language and, subsequently, output the translated chunks in the same order.

3.5 Experiments

Using parallel text data (standard Japanese and 48 regional dialects), we trained both a single dialect-to-standard translation model and a reverse (standard-to-dialect) model, measuring the translation quality using BLEU scores (Papineni et al., 2002b). In addition, we analyzed the trained dialect embeddings in detail and conducted data ablation tests.

3.5.1 Experimental Setup

For these experiments, we split the corpus into training, development, and the test sets in an 8:1:1 ratio. We oversampled the translation pairs to ensure that every dialect had the same amount of training data, because there were different numbers of training and test instances for each dialect (in Figure 3.1). For the oversampling, we randomly sampled the existing sentence in each dialect dataset. Finally, all the training sets for each dialect consisted of 1,042 sentences, the same size as the largest original training set of the Iwate dialect.

Because Japanese dialects mostly share the same vocabulary and there are few distortions (word order changes), we expect that the translation between a Japanese dialect and standard Japanese is relatively easy compared to that between other languages. Thus, the main focus of the following experiments was to evaluate how well the model captured the

Table 3.2 Experimental settings of OpenNMT such as hyper parameters.

Train	
epoch	20
layer (encoder / decoder)	2
batch size	64
valid batch size	32
word embedding dim.	500
hidden dim.	500
dropout rate	0.3
optimizer	SGD
learning rate	1.0
Decode	
Beam-size	5

phonological shifts between the dialects and the standard Japanese. Therefore, we employed syllable-level (i.e., character-level) BLEU scores as the evaluation measures. We calculated the syllable-level BLEU for each sentence by concatenating the chunk-wised translations. Note that this evaluation measure generally yields higher scores than those calculated at the word level. Finally, we macro-averaged the scores over all the dialects. For Multi NMT or SMT models, we generated the translation output for the entire test set, which contained all the dialects, and we divided it into the 48 dialect test sets. Subsequently, we calculated the macro-averaged BLEU scores using the 48 local BLEU scores obtained on the test set for all the dialects. For Mono NMT or SMT models, we trained 48 local NMT/SMT models using a local training/valid set (also a subset of the entire training/valid set) and subsequently evaluated it with a local test set. For all the settings, for the evaluation, we used the test set written as one sentence per line. The difference between the dialect-to-standard and standard-to-dialect translations is simply the exchange of the source and target languages. In fact, the macro-averaged BLEU score reached 35.10 even when we simply output the dialect sentences without translation.

We used OpenNMT-py[5] with its default hyper-parameter settings, except for the number of training epochs (which we set to 20), and selected the model that performed best on the development set. For the details, we list the hyperparameter settings in Table 3.2. In addition, we employed Moses[6] (Koehn et al., 2007) as the baseline SMT model and set the distortion limit to 0. The standard Japanese language model used in Moses was trained with

[5] https://github.com/OpenNMT/OpenNMT-py

[6] http://www.statmt.org/moses

Table 3.3 Descriptions of each model.

System	Translation unit	Model	Identify dialects
None (w/o translation)	-	-	-
Mono NMT	*bunsetsu*	local NMT ×48	True
Multi NMT (w/o labels)	*bunsetsu*	multilingual NMT	False
Multi NMT-sentence (w/ labels)	sentence	multilingual NMT	True
Multi NMT (w/ labels)	*bunsetsu*	multilingual NMT	True
Mono SMT	*bunsetsu*	local SMT ×48	True
Multi SMT (w/o labels)	*bunsetsu*	multilingual SMT	False

KenLM (Heafield, 2011). For the syllable-to-syllable translation, we used MeCab 0.996[7] to analyze the pronunciations of the *kanji* characters.

Regarding the dialect label order used for the input, our preliminary experiments on the validation set indicated that the best models were obtained using input sequence (d) (Table 1) for the dialect-to-standard translation and input sequence (b) for the standard-to-dialect translation.[8]

A brief description of each model we used in our experiments is provided in Table 3.3. Except for the multi-sentence NMT models, we used each *bunsetsu* as a translation unit. Note that, as we mentioned in Sec. 3.4, it is practically unnecessary to model the word order distortion in Japanese dialect translation. The individual dialects are distinguished by two methods: addition of dialect labels to the multilingual NMT, and training the local models for each dialect separately. In comparison, the multilingual NMT and SMT systems without the labels (Multi NMT w/o labels, Multi SMT w/o labels) do not distinguish dialects, and therefore, are disadvantageous.

3.5.2 Multi-Dialect NMT Model Performance

Table 3.4 summarizes the results of the dialect translation performance of all the considered models, with the first row group comprising their scores for dialect-to-standard translation under different input settings.

Monolingual vs. multilingual: For comparison, we first considered a model that was trained using only a single set of dialect-standard parallel data (Mono NMT). It performed quite poorly compared to the other models that used data for all the dialects (Multi NMT)

[7] http://taku910.github.io/mecab/

[8] See Appendix B for more details.

Table 3.4 Syllable-level BLEU scores of all models.

System	BLEU
dialect-to-standard	
None (w/o translation)	35.10
Mono NMT	22.45
Multi NMT (w/o labels)	71.29
Multi NMT-sentence (w/ labels)	69.74
Multi NMT (w/ labels)	**75.66**
Mono SMT	52.98
Multi SMT (w/o labels)	73.54
standard-to-dialect	
Multi NMT (w/ labels)	65.30

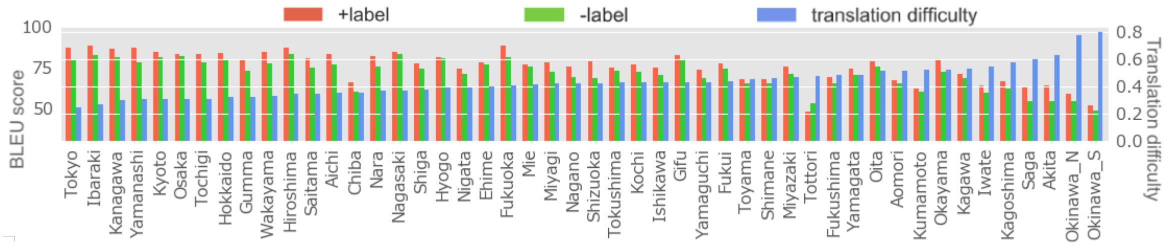


Figure 3.3 BLEU scores of Multi NMT models and translation difficulty for all dialects.

and was even worse than simply outputting the dialect sentences unchanged (35.10). This indicates that training independent NMT models for each language pair with a limited amount of training data is extremely inefficient. In contrast, the multi-dialect model presented a drastically improved the translation performance.

Dialect labels: Including dialect labels improved the Multi-NMT BLEU score by 4.37 points (fifth row of Table 3.4) compared to that of the same model without the dialect labels (third row). Figure 3.3 shows for these two models the BLEU scores for all the dialects in ascending order of translation difficulty. Here, the translation difficulty is defined as the average normalized Levenshtein distance over all the sentence pairs (dialect and standard Japanese) for a given dialect. As expected, the BLEU scores for all the dialects present a strong negative correlation ($\rho = -0.82$) with the translation difficulty. In addition, we can observe that the model with language labels consistently outperforms that without the labels, except for the Tottori dialect, for which there is an extremely small amount of text data

(Figure 3.1). This result indicates that explicit information of the source and target dialects with dialect labels can improve the encoding and decoding accuracy.

Fixed-order translation: Comparing the proposed model (Multi NMT) with the same model trained via the standard approach of using entire sentences as input/output sequences (Multi NMT-sentence) shows that Multi NMT outperforms Multi NMT-sentence by 5.92 points. One disadvantage of the chunk-wise translation is that it cannot capture the context beyond the boundary of each chunk; however, despite this disadvantage, our Multi NMT model can still outperform the model with an access to a broader context (Multi NMT-sentence). This indicates that our fixed-order translation approach is suitable for translating Japanese dialects, despite its limited context sensitivity.

NMT vs. SMT: Zoph et al. (2016) found that SMT models largely outperformed state-of-the-art NMT models for low-resource languages. The second-row group in Table 3.4 summarizes the results for a fixed-order character-based SMT baseline. In these experiments, the NMT model trained using a single dialect (Mono NMT) yielded the poorest performance; however, the one with dialect labels outperformed the baseline Multi SMT model, achieving the best performance overall.

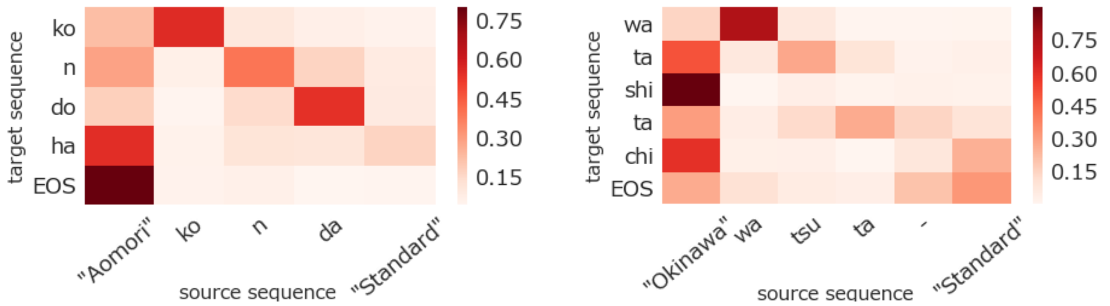
3.5.3 Example of Translation Results

To demonstrate how each of the proposed component contributes to generating accurate translations, we now present some concrete examples of the translation results of our models for the Hyogo, Kagoshima, and Nigata dialects (Table 3.5).

Comparing the Multi NMT models with and without dialect labels, we noted that adding labels enables the models to better translate the chunks that required dialect-specific knowledge. In Example 1, the source sentence includes a local name, おー (*O* -), for a certain area, あいおい (*Aioi*), in Hyogo, which only the model with dialect labels can successfully translate. In addition, in Example 2, the dialect labels enable the model to capture a dialect-specific transliteration rule for the functional suffix (“ta ra”), a conditional-mood marker in the last chunk of the reference sentence (i.e., “ta ya” to “a ra”).

Similarly, because the Multi SMT model could not take advantage of the dialect labels, it failed to capture dialect-specific translation rules.

In the previous section, we noted that our fixed-order translation approach is suitable for translating Japanese dialects, despite its limited context sensitivity. However, this becomes a problem in Example 3, where the chunk-wise translation models cannot correctly translate



(a) Aomori-to-standard

(b) Okinawa-to-standard

Figure 3.4 Attention weight examples. (a) *Aomori-to-standard* translation of "next time". Aomori word *konda* is formed by linking syllables from two standard Japanese words, *kondo* (next time) and *ha* (topic marker). (b) *Okinawa-to-standard* translation of "we". Okinawa word *watta-* combines two standard Japanese words, *watashi* (I) and *tachi* (plural marker), with *watt* and *ta-* roughly corresponding to *watashi* and *tachi*, respectively.

a *bunsetsu* owing to the lack of context. Here, none of the models, except the Multi NMT-sentence, can translate the *bunsetsu*, "mi zu n", in the Nigata dialect to the correct standard Japanese *bunsetsu* "mi zu no." Because the translation of the functional word, "n," in the Nigata dialect is ambiguous, it can be translated as either "ga" (nominative marker) or "no" (of) depending on the following context. This example exposes the limitations of our chunk-wise translation models and suggests potential future directions: extending the fixed-order translation to incorporate contextual information.

3.5.4 Visualizing Attention Weights

Here, to investigate how the proposed model translated the *kana* sequences in various dialects, we visualized the attention weights of the best-performing model for some correctly-translated examples.

Figure 3.4(a) shows the attention history of the model for an example where a part of the target language (standard Japanese) *bunsetsu* changes from the source language (Aomori dialect) according to a simple regular rule. In such cases, the model tends to weight the dialect label heavily when applying the rule ("da" → "do ha"). Conversely, Figure 3.4(b) shows the attention history for an example where almost all the syllables are transcribed. In these cases, the model needs to disambiguate the morpheme-level definitions to create a correct translation, and thus, tended to focus on the entire sequence of semantically- or grammatically-related morphemes.

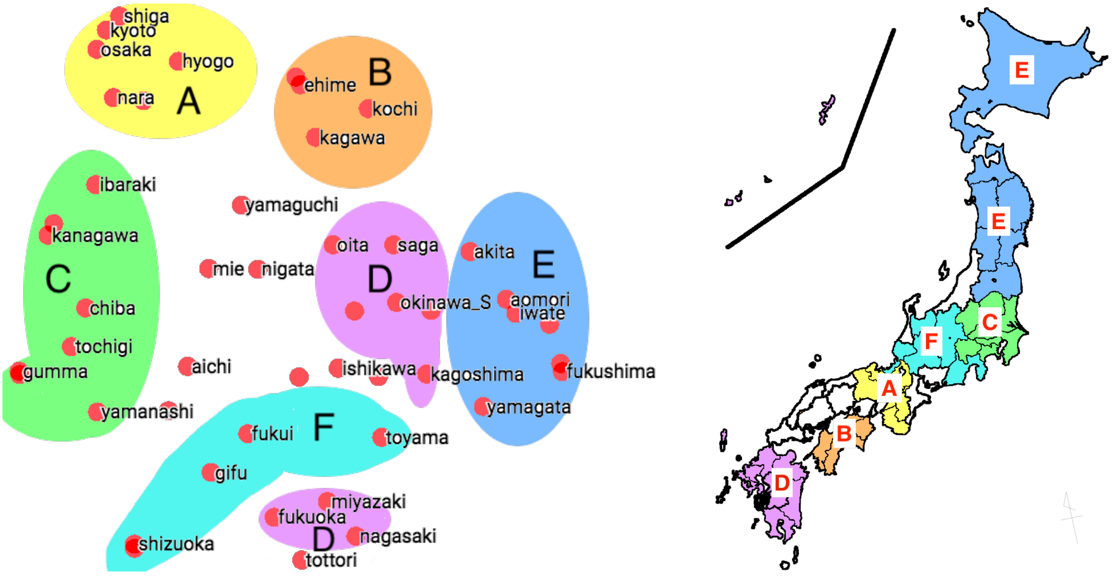


Figure 3.5 T-SNE projection of dialect label vectors. Dialects belonging to same region are shaded using same background color.

3.5.5 Visualizing Dialect Embeddings

Östling and Tiedemann (2017) reported that clustering the language embeddings used to train a multilingual language model produced a language cluster structure similar to those of established relationships among the language families. Motivated by their work, we decided to examine the relationships between the dialect embeddings and the typology of the dialects.

Figure 3.5 shows the t-Distributed Stochastic Neighbor Embedding (t-SNE) projection of the dialect embeddings. It indicates that dialects from neighboring regions tend to form a single cluster. Furthermore, we can observe an interesting agreement between the cluster distances and the predictions of the dialectological typology theory, known as *center versus periphery* (Yanagida, 1980), wherein new language use trends gradually propagate from the cultural center (the old capital, Kyoto) to less culturally influential areas. This potentially explains why the dialects in the Tohoku region (E) are similar to those in the Kyushu region (D), despite their large geographical separation.

3.5.6 Effect of the Nearby Dialects

To investigate in more detail how jointly learning multiple dialects contributed to the dialect-to-standard translations for each dialect, we performed an ablation study on all the dialect regions. As presented in the previous section, the dialects in geographically close regions are generally more similar to each other than those in other regions. Therefore, we assumed that the impact of sharing data from other dialects would differ depending on their geographical distances from the target dialect.

To investigate this assumption, we prepared two Multi NMT models per dialect, trained on the data that excluded the five geographically nearest or farthest dialects[9] for the given dialect region, and calculated the differences in the BLEU scores of these models and the original model, for all the dialect regions. For example, the −*nearest 5* model for the Tokyo dialect is the Multi NMT model learned with the training data excluding the Chiba, Kanagawa, Saitama, Gumma, and Ibaraki dialects. Subsequently, we compared this BLEU score of the model for the Tokyo dialect in the test set to that of the original model trained with full data.

Table 3.6 lists the average results over all the 48 models for both the cases. Both the models trained without the nearest five dialects, and those without the farthest five dialects yielded lower average BLEU scores for their target dialects than the full models. This suggests that even very distant dialects still assist in training other dialects. In addition, we note that removing the nearest five dialects had a more significant impact than removing the farthest five dialects, indicating that similar dialects contribute more to assisting a multi-dialect NMT to learn effectively.

3.6 Conclusion

We have examined the effectiveness of a multilingual, syllable-based, fixed-phrase-order NMT model for translating Japanese dialects into standard Japanese. The results showed that each component of our multi-dialect NMT model successfully improved the translation accuracy when using a limited amount of supervised training data. In addition, we demonstrated the potential benefit of analyzing dialect embeddings for dialectological analysis applications, and have also analyzed how the multi-dialect NMT leverages the training data involving similar dialects to translate a given dialect.

One limitation of the proposed model is that it cannot consider longer-range dependencies beyond the chunk level. Therefore, our future research plans include incorporating contextual information, e.g., n-to-1 translation (Tiedemann and Scherrer, 2017), into fixed-order translation models and investigating the characteristics of the dialect embeddings further.

[9] The distances between the dialect pairs were calculated using the Euclidean distances between the points where the dialogs were recorded.

3.7 Appendix

Data size for each dialect We show the details of the dataset size in Figure 3.6. In our experiments, we used the largest available Japanese dialect dataset; however, the number of sentences in some dialects (e.g., Tottori) was small in our dataset. Owing to the lack of the training or test dataset, the BLEU score of the Tottori dialect is actually inconsistent with those of the other dialects, as shown in Figure 3.3. However, this result suggests that the proposed method is effective for almost all the dialects, except in a very low-resource scenario.

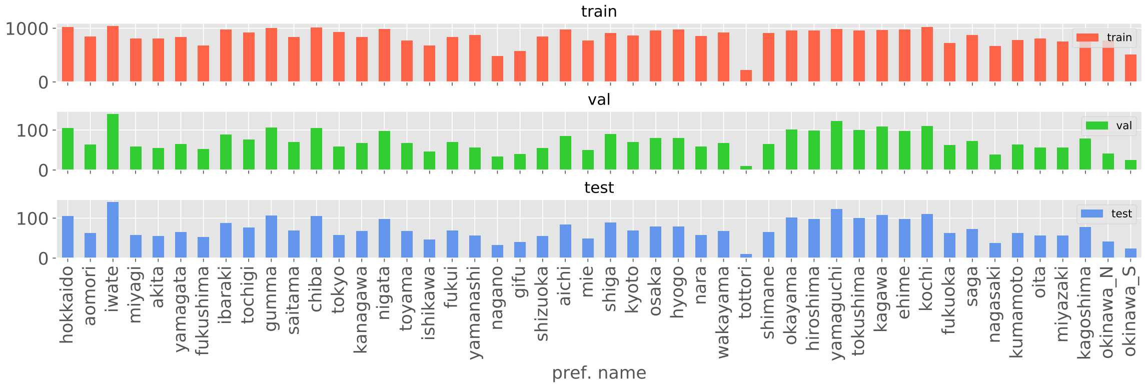


Figure 3.6 Size of each dataset (training/valid/test) in our experimental setting per dialect.

Experiments of dialect label order variants We summarize the results of the preliminary experiments to examine the best setting of the dialect labels in Tables 3.7 (validation set) and Table 3.8 (test set). As can be seen from both the tables, the best models are obtained using input sequence (d) for the dialect-to-standard translation and input sequence (b) (in Table 3.1) for the standard-to-dialect translation. Label set (b) in the dialect-to-standard translation and set (a) in the standard-to-dialect translation does not contain the dialect information, in contrast with the other settings. They present quite lower performances.

Example 1	Hyogo region
(Meaning	Yes, until then, in Aioi ...)
Source	n - / so re ma de / o - ni wa (んー / それまで / おーにわ)
Reference	u n / so re ma de / a i o i ni ha (うん / それまで / あいおいには)
Multi NMT (w/o label)	u n / so re ma de / o o ni ha (うん / それまで / おおには)
Multi NMT-sentence (w/ label)	n - / so re ma de / a t ta n da (んー / それまで / あったんだ)
Multi NMT (w/ label)	n - / so re ma de / a i o i ni ha (んー / それまで / あいおいには)
Multi SMT (w/o label)	u n / so re ma de / o o ni ha (うん / それまで / おおには)
Example 2	Kagoshima region
(Meaning	After a few days, then it was...)
Source	so i ga / mo / na n ni k ka / shi ta ya (そいが / も / なんにっか / したや)
Reference	so re ga / mo u / na n ni chi ka / shi ta ra (それが / もう / なんにちか / したら)
Multi NMT (w/o label)	so re ga / mo u / na n ni chi ka / shi ta da (それが / もう / なんにちか / しただ)
Multi NMT-sentence (w/ label)	so re ga / mo u / na n ni tsu ka / shi ta yo (それが / もう / なんにっか / したよ)
Multi NMT (w/ label)	so re ga / mo u / na n ni chi ka / shi ta ra (それが / もう / なんにちか / したら)
Multi SMT (w/o label)	so re ga / mo u / na ni ka / shi ta de (それが / もう / なにか / したで)
Example 3	Nigata region
(Meaning	I want to go to the water park as soon as possible, but...)
Source	ha yo - / mi zu n / do ko e / i ko - to / o mo u ke do (はよー / みずん / どこえ / いこーと / おもうけど)
Reference	ha ya ku / mi zu no / to ko ro he / i ko u to / o mo u ke re do (はやく / みずの / ところへ / いこうと / おもうけれど)
Multi NMT (w/o label)	ha ya ku / mi zu ga / do ko he / i ko u to / o mo u ke do (はやく / みずが / どこへ / いこうと / おもうけど)
Multi NMT-sentence (w/ label)	ha ya ku / mi zu no / to ko ro he / i ko u to / o mo u ke do (はやく / みずの / ところへ / いこうと / おもうけど)
Multi NMT (w/ label)	ha ya ku / mi zu ga / do ko he / i ko u to / o mo u ke re do (はやく / みずが / どこへ / いこうと / おもうけれど)
Multi SMT (w/o label)	ha ya ku / mi zu ga / do ko he / i ko u to / o mo u ke do (はやく / みずが / どこへ / いこうと / おもうけど)

Table 3.5 Example dialect-to-standard translations for Hyogo, Kagoshima, and Nigata dialects.

Table 3.6 Impact of excluding nearest or farthest five dialect regions from training data when calculating BLEU score for each dialect region. "Avg. Δ" denotes the average BLUE score difference compared to that using all the data.

Dataset	Avg. Δ	#Regions BLEU decreased
−nearest 5	−0.94	34 / 48 (71%)
−farthest 5	−0.22	31 / 48 (65%)

Table 3.7 Syllable-level BLEU scores for each dialect label in Multi NMT in both translation directions on the validation sets. We showed the details of dialect labels (a)–(d) in Table 3.1.

Seed	(a)	(b)	(c)	(d)
dialect-to-standard				
0	90.16	89.30	90.28	**91.03**
1	90.92	89.45	90.93	**91.16**
standard-to-dialect				
0	79.32	**85.37**	84.35	84.68
1	79.36	**85.34**	84.39	84.83

Table 3.8 Syllable-level BLEU scores for each dialect label in Multi NMT in both translation directions on the test sets.

Seed	(a)	(b)	(c)	(d)
dialect-to-standard				
0	74.35	71.59	74.25	**75.57**
1	75.31	71.44	75.43	**75.50**
standard-to-dialect				
0	51.30	**65.30**	64.18	63.17
1	51.61	**65.13**	64.06	63.19

Chapter 4

Evaluation (1) : Comprehensive Analysis of Semantic Metrics for MT

4.1 Introduction

Computing the semantic similarity between two texts is crucial in various NLP tasks. One prominent cluster of application examples is the use of semantic similarity as a metric for evaluating automatically generated text (e.g., machine translation and text summarization) considering gold reference texts (Rei et al., 2020; Sellam et al., 2020; Zhang et al., 2020a). Such semantic similarity metrics are also reported effective as a loss function for training language generation models (Wieting et al., 2019; Yasui et al., 2019). Another common application of the semantic similarity can be seen in text/sentence retrieval, where estimating the relevance between a given query and retrieved texts is an essential component (Chen et al., 2017; Gao et al., 2021a; Karpukhin et al., 2020; Qu et al., 2021).

For more than a decade, a framework, known as Semantic Textual Similarity (STS) has been widely used to test computational models of semantic similarity (Agirre et al., 2012). Over the last decade, STS has emerged as the de-facto standard task for evaluating semantic similarity models, and numerous studies have been published to propose semantic similarity models over a decade (Chuang et al., 2022; Gao et al., 2021b; Giorgi et al., 2021; Lan and Xu, 2018; Li et al., 2020; Reimers and Gurevych, 2019; Severyn et al., 2013; Yan et al., 2021; Zhang et al., 2020b, etc.).

The STS evaluation framework assumes that a model that performs well for the general STS task should also perform well for specific application-oriented tasks. Based on this assumption, models proposed for and evaluated on STS have been applied to application-oriented tasks. For example, in machine translation (MT) evaluation, for the model incorporating several universal sentence encoders (USE) (Cer et al., 2018; Conneau et al., 2017; Logeswaran and Lee, 2018), which performed well on STS, had the highest performance in WMT18 (Shimanaka et al., 2018). These studies appear to provide empirical evidence supporting the assumption that STS performs well as a general proxy for specific application-oriented tasks.

However, in this study, we question this widely accepted assumption. Specifically, we empirically investigated whether semantic similarity models superior to the general STS task perform better on specific application-oriented tasks. In the experiments, we chose MT Evaluation (MTE) as the representative application-oriented task of STS, and investigated the correlation of the performance of numerous (> 20) sampled models between STS and MTE. From the results, we gained several findings as follows:

- Semantic similarity models exhibited a non-negligible gap in performance on STS and the specific task (i.e., MTE) (Fig. 4.1).
- The discrepancies appeared to be caused by the discrepancies between the STS and MTE datasets, including (i) sentence length distribution, (ii) vocabulary coverage, and (iii) granularity of gold-standard similarity scores.

The identified gap, which we refer to as **the evaluation gap**, indicates that the assumption in question does not necessarily hold and demonstrates the potential dangers of relying solely on the current STS-based evaluation alone in studying the semantic similarity. We believe that our findings will be considered in future research on the crucial components of NLP.

4.2 Related Work

The necessity of the semantic similarity in application-oriented tasks. Semantic similarity is required in various NLP application tasks, and STS was motivated by being a surrogate task for such application-oriented tasks (Agirre et al., 2012; Cer et al., 2017). These tasks comparing similarity can be categorized into two types, namely, (1) reference-based evaluation and (2) semantic retrieval. For example, the reference-based evaluation is commonly used in the natural language generation (NLG) fields such as MT, summarization, and simplification. Semantic retrieval includes PR, dialog retrieval, as well as machine reading comprehension. Among these application-oriented tasks, we selected MT evaluation.

In fact, MT evaluation have several examples that incorporate STS-based models. For example, Castillo and Estrella (2012); Shimanaka et al. (2018) applied STS model for MT evaluation and demonstrated the effectiveness of those models. However, relying on the STS evaluation for semantic similarity models could be risky when there is no sufficient correlation between the evaluation of STS and the application-oriented task such as MTE. We investigates the evaluation gap between STS and MTE, to identify vulnerabilities in the STS evaluation in the real world.

Validity of NLP evaluation protocol. Recently, the validity of evaluation protocols, such as benchmark datasets (Bowman and Dahl, 2021) or metrics (Durmus et al., 2022; Mathur et al., 2020) has been questioned on various NLP tasks. Many studies have identified the bias or lack of certain factors in the evaluation protocol. Søgaard et al. (2021); Varis and Bojar (2021) investigated the effects of differences in the sentence length distribution between train and test sets. Additionally, a difference in vocabulary distribution (domain mismatch) is also often mentioned as an important factor affecting the evaluation (Wang et al., 2022; Zhang et al., 2020b). In terms of an STS-specific factor, Reimers et al. (2016) highlighted the difference in the granularity of similarity between STS and downstream tasks. They focus on appropriate task-intrinsic evaluation metrics for STS-based models, considering different downstream tasks; however, their thought is also based on the assumption that the STS-based models are useful for the downstream tasks. In our study, we question this assumption. Based on these previous studies, we analyze the effects of three factors, **sentence length**, **vocabulary**, and **similarity granularity**, contributing to the evaluation gap between STS and the application-oriented task such as MTE.

Discussion of the problems of STS benchmark. While many models have been proposed using the STS evaluation, some studies have also questioned the STS or conducted an additional evaluation for specific factors that are not captured by the STS evaluation. Wang et al. (2021) argue that previous studies rely on the STS evaluation and argues that STS lacks domain adaptability. Futhermore, Liu et al. (2021) did not adopt the STS evaluation because of the lack of domain coverage and lack of consideration for context, so they created a new contextual dialog domain STS dataset. In addition, Wieting et al. (2020) extracted a more difficult subset which contains the examples with low word overlap by focusing on a specific factor such as word overlap. Wang et al. (2022) focused on the discrepancy between the evaluation of STS and single-sentence downstream tasks in SentEval, highlighting the problems of domain mismatch and ambiguous annotations. In comparison, we investigated

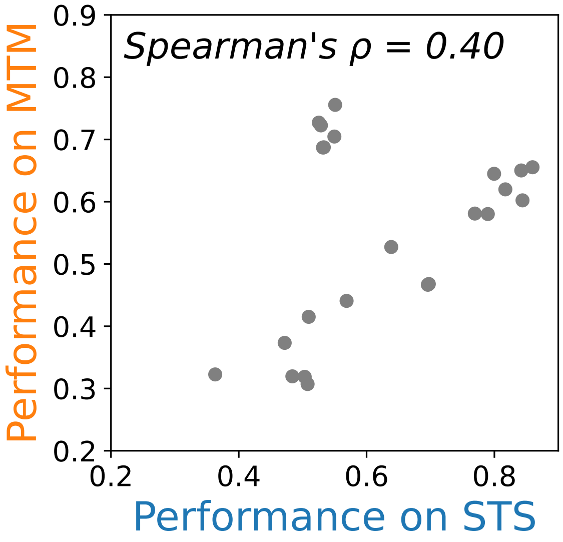


Figure 4.1 Correlation between evaluation using STS and that using a task-specific dataset, such as MT Evaluation (MTE).

whether STS satisfies the original motivation for application-oriented tasks *practically using semantic similarity* (Agirre et al., 2012; Cer et al., 2017).

In summary, we shed the light on the specific factors such as sentence length, vocabulary, and similarity granularity to make the relationship to the evaluation gap explicit. We provided the first evidence that STS has a considerable evaluation gap even from the MT evaluation task, that have been considered representative applications since the inception of STS.

4.3 Is There a Gap between Evaluation of STS and Application-oriented Tasks?

STS dataset (Agirre et al., 2012; Cer et al., 2017) was proposed as a semantic similarity benchmark that can be directly applied to several NLP tasks and is currently the de-facto standard for evaluating semantic similarity models. In this study, to validate the STS benchmark, we conducted comprehensive experiments to examine whether there is a sufficient correlation between the evaluation results on STS and that on the specific application-oriented task dataset such as MTE.

4.3.1 Tasks and Datasets

General settings. We present the definitions of two tasks—STS and MTE—that must capture the semantic similarity addressed in this study. The main structure of the two tasks is comparing a sentence pair (s, s') and predicting the semantic similarity score between the two sentences. We selected MTE as the example of application-oriented tasks of STS. MTE compares relatively similar sentence pairs and provides a gradation score as the gold standard.

STS (STS-b). STS (Agirre et al., 2012) is a task that compares a sentence pair (s1, s2) and predicts a similarity score between the two sentences. The gold-standard similarity score is provided in the range of 0-5. Model prediction scores are evaluated using Pearson or Spearman correlations with the gold standard. In this study, we used Pearson correlation. We used the STS-b dataset (Cer et al., 2017) with image captions, news articles, and forum domain data over a 5-year pilot task (STS12-17).

MT Evaluation (WMT17). MT Evaluation (MTE) is a task that compares a (model hypothesis, reference) pair and predicts the adequacy scores of the model hypothesis relative to the reference. In this study, we use the segment-level Direct Assessment dataset (to-English) in WMT17 (Bojar et al., 2017).[1] We selected this because of the reliability of the manual scores (Sellam et al., 2020; Zhang et al., 2020a). The gold standard score is the normalized value of scores manually evaluated with 100 scales to the pair (model hypothesis, reference). The Pearson or Kendall correlation between the gold standard and the model prediction score is usually used in the evaluation. In this study, we used the Pearson correlation.

Statistics of datasets. Table 4.1 shows statistics of two datasets (STS and MTE) employed in this paper. The dataset size of STS is larger than that of MTE, whereas the total word counts are comparable between STS and MTE.

4.3.2 Semantic Similarity Prediction Model

A semantic similarity prediction model usually involves the following two steps: (i) obtaining a sentence representation and (ii) calculating the similarity between two representations.

To determine whether there is an evaluation gap between various models, we measured the correlation between the evaluation results on STS and that of MTE. In this study, we used

[1] We use cs–en, de–en, fi–en, lv–en, ru–en, tr–en and zh–en datasets, which are sourced from news domain texts. `https://www.statmt.org/wmt17/results.html`

	STS (s1, s2)	MTE (hyp, ref)	
#sentence pairs	8,628	3,793	
#sentences ({s, s'})	15,487	4,261	
#words	186,134	170,565	
#words / {s, s'}	**11.443±6.143**	23.381±11.215	
#words / s	**11.450±6.188**	23.296±11.290	
#words / s'	**11.437±6.099**	23.467±11.138	
	STS-news (s1, s2)	STS-forum (s1, s2)	STS-image-captions (s1, s2)
#sentence pairs	4,299	1,079	3,250
#sentences	8,268	1,913	5,306
#words	107,957	25,456	52,721
#words / {s, s'}	12.927±7.506	12.642±4.978	9.0823±2.910
#words / s	12.949±7.564	12.677±5.007	9.0585±2.906
#words / s'	12.905±7.448	12.608±4.949	9.1062±2.914

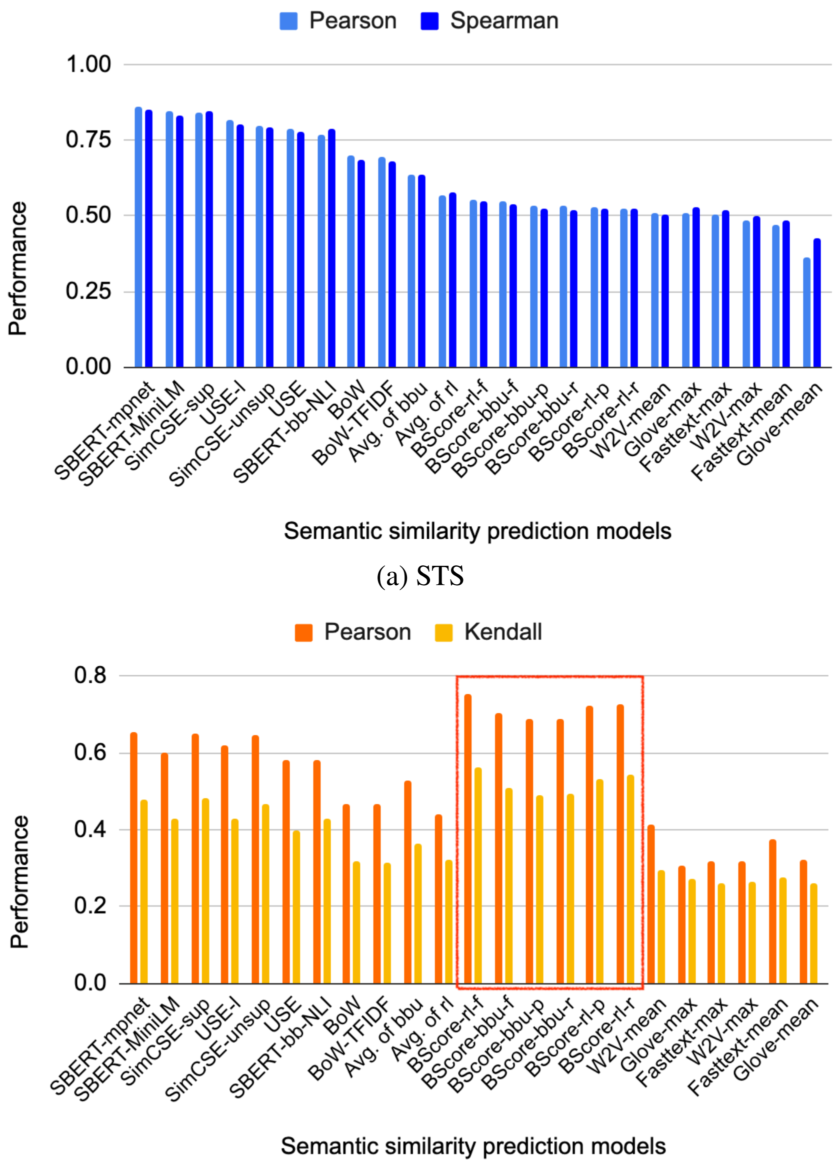
Table 4.1 Stats. of sentences and words and average of sentence length for STS (all and sub-domain sets) and MT Evaluation: MTE).

the following 23 semantic similarity prediction models: **BoW**-{raw, TFIDF}-sum, **BoV**-{Word2vec*, Glove, Fasttext}-{mean, max}, **USE**-{normal, large}, **Avg. of BERT**-{BERT-base-uncased (bbu), RoBERTa-large (rl)}, **BERTScore (BScore)**-{BERT-base-uncased, RoBERTa-large}-{precision, recall, F1-score}, **Sentence-BERT (SBERT)**-{bertbase-NLI-mean, MiniLM, mpnet}, and **SimCSE**-{supervised, unsupervised}.

Table 4.2 shows the descriptions of the models used in this paper.

	model	dim	similarity function	pooling	others
SimCSE-sup	princeton-nlp/sup-simcse-bert-base-uncased	default	cos		
SimCSE-unsup	princeton-nlp/unsup-simcse-bert-base-uncased	default	cos		
SBERT-bb-NLI-mean	bert-base-nli-mean-tokens		cos	mean	
SBERT-MiniLM	all-MiniLM-L6-v2	384	cos	mean	
SBERT-mpnet	all-mpnet-base-v2	768	cos	mean	
BERTScore-rl-p	roberta-large	default	precision		
BERTScore-rl-r	roberta-large	default	recall		
BERTScore-rl-f	roberta-large	default	f1-score		
BERTScore-bbu-p	bert-base-uncased	default	precision		
BERTScore-bbu-r	bert-base-uncased	default	recall		
BERTScore-bbu-f	bert-base-uncased	default	f1-score		
avg. of BERT-bbl	bert-base-uncased	768	cos	mean	
avg. of BERT-rl	roberta-large	768	cos	mean	
BoV-Word2Vec (mean)	GoogleNews-vectors-negative300.magnitude	300	cos	mean	
BoV-Word2Vec (max)	GoogleNews-vectors-negative300.magnitude	300	cos	max	
BoV-Glove (mean)	glove.840B.300d.magnitude	300	cos	mean	
BoV-Glove (max)	glove.840B.300d.magnitude	300	cos	max	
BoV-fasttext (mean)	crawl-300d-2M.magnitude	300	cos	mean	
BoV-fasttext (max)	crawl-300d-2M.magnitude	300	cos	max	
BoW (sum)	CountVectorizer (sklearn, use smooth idf, stopwords)	vocab size	cos	sum	
BoW-TFIDF (sum)	TfidfVectorizer (sklearn, stopwords)	vocab size	cos	sum	norm=L2
USE	universal-sentence-encoder	512	cos		norm=L2
USE-l	universal-sentence-encoder-large	512	cos		norm=L2

Table 4.2 Semantic similarity model descriptions.



(a) STS

(b) MT Evaluation

Figure 4.2 Performance of semantic similarity models on STS and MT Evaluation.

4.3.3 Experimental Procedure and Results

Fig. 4.2 compares the evaluation for each semantic similarity prediction model on STS and MTE. The x-axis represents the semantic similarity prediction models, which are ordered by decreasing the performance on STS from left to right. Compared with STS, the performance of each model differs largely in MTE. For the STS evaluation, SBERT (mpnet: 0.86) outperforms BScore (RoBERTa-large, F1-score: 0.55); however, in MTE, those performances are inverse as SBERT (0.66) < BScore (0.76). Both STS and MTE, both correlation measures have a similar trend for model ranking in each task (Fig. 4.2), thus we used the Pearson correlation in each task's evaluation. In addition, we calculated Spearman correlation coefficients between the performance on STS and that on each task to precisely visualize these performance gaps (Fig. 4.1). Here, we define these correlation coefficients as the value of the evaluation gap. A lower correlation value indicated a larger evaluation gap. In Sec. 4.4, we examine changes in the evaluation gap when the explanatory variables (e.g., sentence length, vocabulary coverage, similarity granularity) are changed.

4.4 What Factors Cause the Evaluation Gap?

As mentioned in Sec. 4.3, there is a large gap between the specific application-oriented task and STS used as frameworks for evaluating the sentence similarity prediction models. In this section, we discuss three potential factors contributing to the gap between evaluation frameworks, as well as the dataset features that should considered to when using STS for evaluation.

4.4.1 Factor 1: Difference in Sentence Length

In the following, we discuss the sentence length (i.e., the number of words in a sentence). Words are commonly used as the basic unit in NLP models. This is also true when making predictions of semantic similarity measures. We focused on the large variance in the number of words (i.e., sentence length) in the target text for similarity measurement. Some studies reported that differences in the sentence length distributions produce different scores on different test sets (Søgaard et al., 2021; Varis and Bojar, 2021). Therefore, we hypothesize that differences in the distribution of sentence lengths by task may result in an evaluation gap.

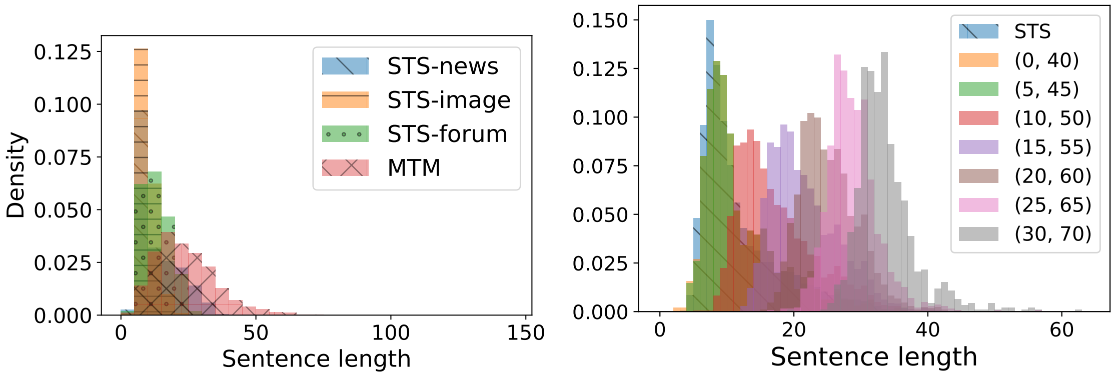
Short sentence length in STS benchmark

Here, we demonstrate that the *STS dataset has shorter sentence lengths than the datasets for other specific tasks*, such as MTE. Histograms of the sentence length distribution for each dataset are presented in Fig. 4.3(a). Compared with the sentence length distribution of the MTE task, STS has a biased sentence length distribution consisting of short sentences.

Also in Table 4.1, the sentence length distribution (the number of of words / {s,s'}) shows that STS has very few words per sentence compared to the MTE task. As for the STS sub-domain sets, the three sets have different sentence length distributions. We additionally describe the histograms of the sentence length distributions for the three STS sub-domain

	size	avg. sent len
[0, 40)	481	11.610±5.794
[5, 45)	481	11.790±5.979
[10, 50)	1225	16.841±5.747
[15, 55)	1484	21.086±5.015
[20, 60)	1112	24.722±4.286
[25, 65)	715	28.260±3.733
[30, 70)	465	33.184±4.462

Table 4.3 Statistics of sentence length subsets for MTE. The “size” means the number of sentence pairs and the “avg. sent len” means the average of sentence length for each subset.



(a) original datasets (STS and MTE)

(b) MTE subsets

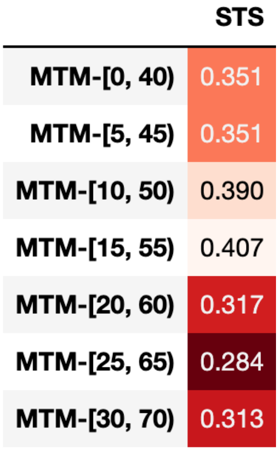
Figure 4.3 Histogram of sentence length in original datasets (STS and MT Evaluation: MTE) and MTE subsets according to sentence length.

sets in Fig. 4.1. As illustrated here, the average sentence length of the image-caption domain is particularly highly biased for shorter sentence lengths.

Does the sentence length gap cause the evaluation gap?

There is a difference in the sentence length distribution between STS and the application-oriented task. Here, we investigate whether eliminating the difference in sentence length between the STS and the application task (MTE) alleviates the evaluation gap.

Settings. We created subsets of the MTE dataset to match or differ the STS sentence length distribution, and then, compared the correlations between the STS evaluation result and MTE result for the different models. The subset [x, y) was drawn from a range of sentence lengths [x, y) according to the STS distribution. In MTE, the subsets were split based on the average sentence length of the sentence pairs. Statistics of the subset of sentence length is shown in



	STS
MTM-[0, 40)	0.351
MTM-[5, 45)	0.351
MTM-[10, 50)	0.390
MTM-[15, 55)	0.407
MTM-[20, 60)	0.317
MTM-[25, 65)	0.284
MTM-[30, 70)	0.313

Figure 4.4 Spearman correlations between performance with STS and that with the subsets split according to sentence length with MT Evaluation task. The darker color represents the lower correlation (= the larger evaluation gap). [x, y) means that the subsets consist of the examples of the sentence length from x to y.

Table 4.3. Histograms of the created subsets according to sentence length distribution are shown in Fig. 4.3(b). We created MTE subsets from [0, 40) to [30, 70). The shorter MTE subsets, such as [0, 40) and [5, 45), had nearly the same distribution as the STS set. We investigated whether correlations were lower in the task-specific datasets (i.e., the evaluation gap was amplified) when their sentence length distribution was more different from that of STS.

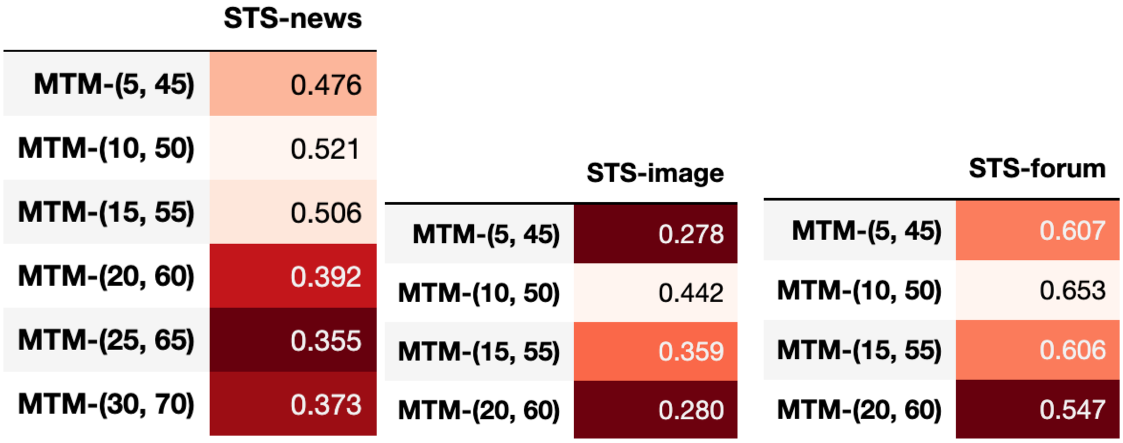
Results. Figs. 4.4 present the Spearman correlations between the performance of the models on STS and that on the MTE subset with adjusted sentence length distributions, respectively. For MTE, the greater the difference in the sentence length distribution, the lower the correlation (i.e., the larger the evaluation gap). This result indicates that the difference in the sentence length distribution contributes to the evaluation gap between STS and MTE.

Analysis: In-domain vs. Out-of-domain. The STS dataset is sourced from three different domains (news, image captions, and forum), and the sentence length distribution actually differs for each domain. We conducted additional experiments for three sub-domain sets following the same procedure using subsets, and found that the similar trends that the evaluation gap increases with the larger sentence length subset.

Settings. We create subsets from the MTE dataset to match the sentence length distribution for each of three STS sub-domain sets. Notably, the forum and image caption domains have

	STS-news-based		STS-forum-based		STS-image-captions-based	
	size	avg. sent len.	size	avg. sent len.	size	avg. sent len.
[0, 40)	503	12.898±6.971	400	9.491±3.183	816	12.348±4.347
[5, 45)	506	13.238±7.259	398	9.521±3.162	867	13.106±4.855
[10, 50)	2150	19.356±6.201	676	13.024±2.620	1229	15.444±3.879
[15, 55)	1902	22.082±5.192	778	17.648±2.457	911	18.337±3.013
[20, 60)	1185	24.935±4.332	650	22.185±2.548	658	22.251±2.620
[25, 65)	715	28.260±3.733	-	-	-	-
[30, 70)	465	33.184±4.462	-	-	-	-

Table 4.4 Stats. of sentence length subsets for MTE according the sentence length distribution of STS sub-domain sets. The "size" means the number of sentence pairs and the "avg. sent len" means the average of sentence length (the average of {s, s'}) for each subset.



	STS-news
MTM-(5, 45)	0.476
MTM-(10, 50)	0.521
MTM-(15, 55)	0.506
MTM-(20, 60)	0.392
MTM-(25, 65)	0.355
MTM-(30, 70)	0.373

(a) news

	STS-image
MTM-(5, 45)	0.278
MTM-(10, 50)	0.442
MTM-(15, 55)	0.359
MTM-(20, 60)	0.280

(b) image captions

	STS-forum
MTM-(5, 45)	0.607
MTM-(10, 50)	0.653
MTM-(15, 55)	0.606
MTM-(20, 60)	0.547

(c) forum

Figure 4.5 Spearman correlations between performance on sentence length subsets of STS-news, image captions, forum and MT Evaluation (MTE) . The darker color indicates the lower correlation (= the larger evaluation gap).

relatively small sentence length distributions (in Fig. 4.4, we thus reduced the range of the subsets from [0, 40) to [20, 60). Statistics of the subset of sentence length are shown in Table 4.4.

Results. Fig. 4.5 shows the correlation with MTE when sentence length subsets are created separately for each domain. We observed a similar tendency for all sub-domain sets that the evaluation gap increases for subsets of longer sentence lengths. This suggests that the evaluation results differ due to different sentence length distribution even within the same domain, which is consistent with a previous study's report in a different benchmark (Søgaard et al., 2021).

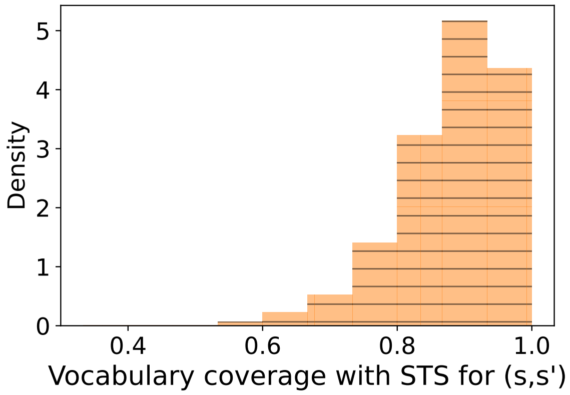


Figure 4.6 Histogram of the ratio of the vocabulary covered with the vocabulary of STS in the MT Evaluation (MTE) for each sentence pair.

4.4.2 Factor 2: Difference in Vocabulary Coverage

Beyond sentence length, there are still other factors that may contribute to the evaluation gap between STS and the application-oriented tasks. Here, we discuss the vocabulary coverage of the MTE task dataset using STS. One reason for focusing on this factor is that the text domains represented in the datasets are distinct. Some studies have highlighted the strong dependence of the STS-based models on domains (Zhang et al., 2020b), as well as mismatch with a dialog domain (Liu et al., 2021). Therefore, we hypothesize that differences in vocabulary coverage due to domain differences may influence the evaluation gap.

Low vocabulary coverage with STS for vocabulary in the applications

Here, we demonstrate that *the STS vocabulary does not adequately cover task vocabulary (MTE)*. For each sentence pair, we calculate the vocabulary coverage, which is the recall of vocabulary in STS (V_{sts}) to the vocabulary in the sentences in the MTE task (s, s'), as follows:

$$\mathrm{Recall}(s, s') = \frac{|(s \cup s') \cap \mathscr{V}_{\mathrm{STS}}|}{|s \cup s'|} \tag{4.1}$$

Fig. 4.6 shows the histograms of $\mathrm{Recall}(s, s')$ for each sentence pair in MTE. In both tasks, most sentence pairs have a vocabulary coverage of less than 1, i.e., they contain vocabulary not covered by STS. Thus, STS vocabulary does not sufficiently cover the vocabulary of the other task.

	STS-based		STS-news-based		STS-forum-based		STS-captions-based	
	size	avg. Recall	size	avg. Recall	size	avg. Recall	size	avg. Recall
(all)	3,793	0.882±0.084	4,299	0.854±0.093	1,079	0.715±0.120	3,250	0.523±0.112
High	100	1.000±0.000	100	1.000±0.000	100	0.980±0.024	100	0.787±0.042
Low	100	0.631±0.060	100	0.588±0.058	100	0.418±0.063	100	0.252±0.062

Table 4.5 Statisitics of vocabulary subsets for MTE.

Does the vocabulary distribution gap cause an evaluation gap?

We investigate whether the low vocabulary coverage with STS examined in Sec. 4.2.1 is indeed a factor contributing to the evaluation gap.

Settings. For the MTE dataset, we extract the top and bottom 100 pairs as the $\mathrm{Recall}(s, s')$-*High* and $\mathrm{Recall}(s, s')$-*Low* subsets, respectively. The MTE $\mathrm{Recall}(s, s')$-*High* subset contains all sentence pairs composed of STS vocabulary. In this experiment, we examine whether higher lexical coverage with the STS vocabulary for the subsets resulted in a higher correlation. Statistics of the subset of vocabulary coverage is shown in Table 4.5.

Results. Table 4.6 presents the Spearman correlation between the performance on STS and those on the $\mathrm{Recall}(s, s')$-*High* and *Low* subsets in MTE. The MTE subsets did not show the hypothesized trend. One reason for the result of MTE is that STS is a mix of three different domains (news, image captions, and forum). In contrast, MTE is a single news domain dataset, which might have caused a divergence in the evaluation of sentence pairs from the same or different domains.

Analysis: In-domain vs. Out-of-domain. To confirm the influence of STS inner domains, we performed an additional analysis. We created vocabulary coverage subsets for the three STS sub-domain sets (news, image captions, and forum) in the same way as for the entire STS, and calculated the correlation between the three STS sub-domain sets and MTE *High/Low* subsets. For an in-domain setting, the MTE subset with *High* vocabulary coverage using STS-news correlated better than that with *Low* vocabulary coverage (0.438 > 0.373), as hypothesized. For out-of-domain settings, the STS-forum set also showed that the *High* subset has a better correlation than the *Low* subset ($0.779 > 0.458$); however, in the image caption set, the correlation of the *Low* subset (0.177) is better than that of *High* subset (0.046). For the image caption domain, the correlation values are extremely low for both the subsets, indicating that the STS image caption set did not play a good role in the evaluation

	Recall(s, s')-*Low*		Recall(s, s')-*High*
MTE	0.276	>	0.272

Table 4.6 Spearman correlations between the performance with STS and that with the subsets split according to higher vocabulary coverage (Recall(s, s')-*High*) and lower one (Recall(s, s')-*Low*) with STS of MT Evaluation task (MTE).

of application-oriented tasks such as MTE. In summary, these results indicate that the vocabulary coverage contributes to evaluating gap between STS and the application-oriented tasks.

Analysis: STS has easier vocabulary STS contains more familiar words than that appear in the application tasks. As quantitative indicators of word familiarity, word frequency (Yimam et al., 2018) and word length (Kincaid et al., 1975) are often used mainly in the text simplification task. Intuitively, the higher the word frequency or the shorter the word length, the more familiar the word. In this case, we use "word frequency (wordfreq)" and "zipf frequency (zipffreq)" scale in `wordfreq` module (Speer et al., 2018).[2] Wordfreq is the normalized frequency in the corpora, and zipffreq is the logarithmically scale of wordfreq. The word length is the number of characters in each word. We use `nltk.word_tokenize()` as word split and filtered out URLs and those with more than 50 characters.

Table 4.7 shows the average word frequency with the wordfreq module and word length for each dataset. In zipffreq, the average of STS is shorter than that of both the application tasks. Also in word length, we could observe that the average of STS is higher than that of MTE. Thus, in both the indicators, word familiarity distribution in STS is higher than in the two application tasks.

Additionally, by comparing between "general" word frequencies (wordfreq) in the `wordfreq` module and actual word frequencies in the corpus (corpus-freq), we can identify words that appear particular high-frequently in the corpus. The words belongs to "corpus-freq − wordfreq > 0.001" for STS and MTE were 43, 18 words, respectively (if excluding stopwords and punctuation, 28 and 3 words, respectively). Examples of higher frequent words in each dataset are shown in Table 4.8. As shown in this, some domain-specific words (STS: image captions, MTE: news) are particularly frequent in each corpus. STS seems to be biased toward certain words (e.g., colors, present progressive forms, relatively abstract nouns such as *man* and *dog*). The results indicate that the STS has a relatively "easier" vo-

[2] A tool to obtain word frequencies from 7 different corpora (Wikipedia, Subtitles, News, Books, Web text, Twitter, Reddit). `https://pypi.org/project/wordfreq/`

	STS	MTE
zipffreq (↑)	**3.59±1.24**	3.45±1.54
length (↓)	**6.97±2.76**	7.34±2.83

Table 4.7 Average of word frequency and word length in STS and MT Evaluation (MTE). The higher (↑) the average for zipffreq (zipf scale of normalized word frequency) or the lower (↓) the average for word length, the higher the word familiarity can be considered.

STS	man, woman, playing, running, guitar, white, black, red, dog, ···
MTE	said, police, olympic(, was, will, which, who, ···)

Table 4.8 Examples of higher frequency words for STS and MT Evaluation (MTE) (stopwords in parentheses).

cabulary (particularly sourced from the image-caption domain) than the application-oriented task.

4.4.3 Factor 3: Difference in Granularity of Gold-standard Scores

Below, we consider the granularity gap of the gold-standard similarity scores between STS and MTE.

We suspect that the granularity of the similarity that was considered in each task varies. The distinction between better or worse hypotheses for high-similarity sentence pairs is an arresting challenge in MTE (Ma et al., 2019). More concretely, the current semantic evaluation model for MTE is unable to finely discriminate the better outputs in highly competitive language pairs such as to-English because of high quality of recent MT output for highly competitive language pairs. Considering this application, we hypothesize that the similarity granularity of STS is insufficient to evaluate such MTE problems.

The discrepancy of the similarity granularity between STS and MTE

The difference in the similarity score between STS and MTE can be seen in some real examples. The actual examples in STS and MTE are illustrated in Table 4.9. STS provides give relatively high scores for the difference between the past and present progressive tenses, and the difference in including proper nouns such as *cholera*, as long as they generally share some elements. However, in MTE, the first example is given a relatively higher score (0.49) for the different actions between *continues to take* and *is already given*, whereas the second example (*Fresh fruit* ...) is assigned a lower score (-0.83), sharing almost similar elements

source		s1 (ref)	s2 (hyp)	gold	BScore	SimCSE
STS	(i)	A man **is riding** a mechanical bull.	A man **rode** a mechanical bull.	4	0.98	0.96
	(ii)	A total of 17 cases have been confirmed in the southern city of Basra, the Organization said.	A total of 17 confirmed cases of **cholera** were reported yesterday by the **World Health** Organisation in the southern **Iraqi** city of Basra.	3.6	0.93	0.74
MTE	(i)	This drug **continues to take** 12 months after a heart attack, which can reduce the risk of a stroke or heart attack.	The drug **is already given for** 12 months after a heart attack, reducing the risk of a stroke or another attack.	0.49	0.94	0.90
	(ii)	Fresh fruit **was replaced with** cheaper dried fruit.	Fresh fruit **is** cheap dried fruit **instead**.	**-0.83**	0.94	0.82

Table 4.9 Actual examples of STS and MT Evaluation (MTE). The gold scores of MTE are normalized in the range (-1.81, 1.44) from with manually evaluated 100-scale scores. “BScore” and “SimCSE” mean prediction scores with BERTScore (RoBERTa-large, F1-score) and SimCSE (supervised), respectively.

but the hypothesis is somewhat difficult to understand. Can this similarity granularity gap cause the evaluation gap?

Does the gap in the granularity of similarity cause an evaluation gap?

Here, we investigate whether the difference in the similarity granularity mentioned in Sec. 4.3.1 results in the evaluation gap.

Settings. For the STS and MTE datasets, we create subsets according to the similarity scores for a sentence pair. We divide the STS dataset into five subsets by considering six labels from 0 to 5. For the MTE dataset, we separated four subsets (*Sim-{Low, MidLow, MidHigh and High}*) by quartiles for human-rated golden scores. We determined the gap between the evaluations using STS and MTE subsets to confirm which range of the similarity granularity impacts the gap in the evaluation. Specifically, the correlation might be higher between the narrower range of the similarity band of STS and the wider range of that of

STS								
	(all)		(news)		(forum)		(image captions)	
	size	avg. similarity	size	avg. similarity	size	avg. similarity	size	avg. similarity
[0, 1]	1182	0.655±0.280	594	0.522±0.393	275	0.472±0.420	931	0.360±0.353
(1, 2]	1348	1.631±0.285	640	1.631±0.283	248	1.687±0.286	460	1.601±0.283
(2, 3]	1672	2.653±0.291	876	2.678±0.291	232	2.656±0.292	564	2.615±0.286
(3, 4]	2317	3.614±0.287	1378	3.599±0.280	189	3.692±0.303	750	3.622±0.292
(4, 5]	1491	4.619±0.304	811	4.613±0.301	135	4.686±0.311	545	4.612±0.306

MTE		
	size	avg. similarity
Sim-Low: [-2, -0.47]	950	-0.820±0.266
Sim-MidLow: (-0.47, -0.03]	948	-0.240±0.126
Sim-MidHigh: (-0.03, 0.42]	943	0.193±0.127
Sim-High: (0.42, 1.5]	952	0.683±0.183

Table 4.10 Dataset size (#sentence pairs) and average & standard derivation of gold-standard similarity scores on STS and MTE subsets.

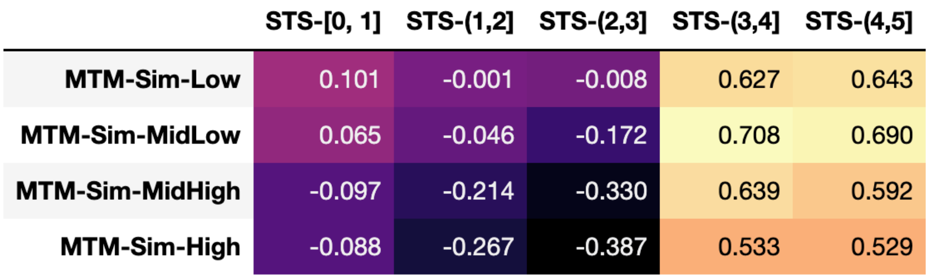
MTE. We anticipate that the higher similarity band in STS only correlates with the MTE dataset, to consider the demand of the MTE that must distinguish higher similarity pairs.

Statistics of the subset of the granularity of similarity is shown in Table 4.10.

Results. Fig. 4.7 shows the Spearman correlations between the similarity granularity subsets of STS and that of the MTE. As hypothesized, only the high-similarity subsets of STS, *STS-(3,4]* and *STS-(4,5]*, were highly correlated with all the MTE subsets. These results significantly show that STS is unable to evaluate discrimination performance in the fine-grained higher similarity bands.

In Fig. 4.8, we describe one interpretation of the above result. We suspect that STS cannot capture fine-grained granularity at higher similarity bands, as discussed (Sec 4.3.1). Not only is the evaluation of the high-similarity band of STS is higher correlated with that of MTE, but the low-similarity band of STS and MTE are nearly uncorrelated or inversely correlated (Fig. 4.7). We should consider introducing finer granularity in high similarity bands for STS, while also considering exclusion examples in ineffective low similarity bands as a widely applicable benchmark.

Analysis: No or low correlations in any domain. As in the previous analyses, we investigated the differences in each domain's tendencies. The correlations between subsets and MTE similarity subsets in each STS sub-domain sets are shown in Fig. 4.5. For the in-



	STS-[0, 1]	STS-(1,2]	STS-(2,3]	STS-(3,4]	STS-(4,5]
MTM-Sim-Low	0.101	-0.001	-0.008	0.627	0.643
MTM-Sim-MidLow	0.065	-0.046	-0.172	0.708	0.690
MTM-Sim-MidHigh	-0.097	-0.214	-0.330	0.639	0.592
MTM-Sim-High	-0.088	-0.267	-0.387	0.533	0.529

Figure 4.7 Spearman correlations between performance on subsets according to gold-standard similarity scores of STS and MT Evaluation (MTE). The darker color represents the lower correlation (= the larger evaluation gap).

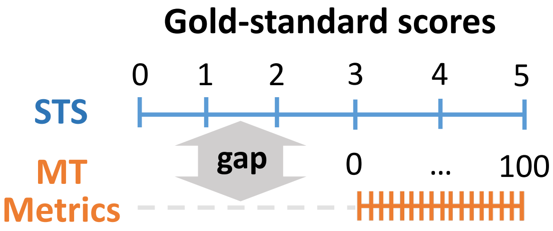
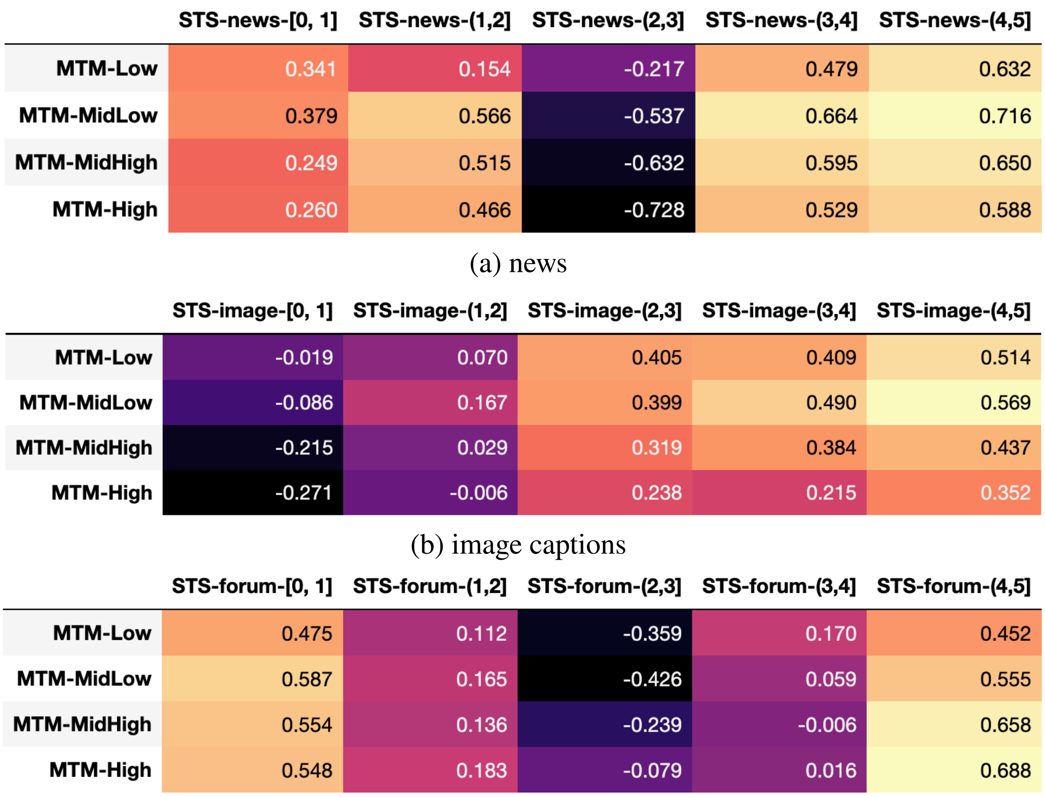


Figure 4.8 The relationship of the granularity of similarity scores between STS and MT Evaluation.

domain setting (STS-news ↔ MTE), only the middle similarity band showed a strong negative correlation with the MTE. For out-of-domain settings, the image caption set showed no correlation with MTE at lower similarity levels. In contrast, the forum domain set showed correlation only at very high or low similarity levels. Thus, the evaluation gap caused by similarity granularity was found to be a cross-domain problem.

Analysis: Ambiguity of similarity criteria in STS and MTE One of the possible reasons for this strange phenomenon is the ambiguity of criteria for similarity in both STS and MTE. Regarding MTE, each similarity score has no concrete criterion. Therefore, the degree of penalty for a particular error depends on the discretion of each annotator. This lack has the potential problem of creating unexpected bias in the annotation. In addition, STS annotations are also ambiguous due to label criteria, discussed in (Wang et al., 2022). For example, there is a large gap between the definitions of 2 (*not equivalent but share some details*) and 3 (*roughly equivalent*) in terms of semantic equivalence. These ambiguities in criteria can be attributed to the evaluation gap.

We described histograms of sentence pairs that differ in some elements (tense, named entities, and pronouns) in the Fig. 4.10.pronouns) in the STS and MTE datasets. We used the `spacy` module to identify verb tenses, POS tags, and named entities. We define the examples



	STS-news-[0, 1]	STS-news-(1,2]	STS-news-(2,3]	STS-news-(3,4]	STS-news-(4,5]
MTM-Low	0.341	0.154	-0.217	0.479	0.632
MTM-MidLow	0.379	0.566	-0.537	0.664	0.716
MTM-MidHigh	0.249	0.515	-0.632	0.595	0.650
MTM-High	0.260	0.466	-0.728	0.529	0.588

(a) news

	STS-image-[0, 1]	STS-image-(1,2]	STS-image-(2,3]	STS-image-(3,4]	STS-image-(4,5]
MTM-Low	-0.019	0.070	0.405	0.409	0.514
MTM-MidLow	-0.086	0.167	0.399	0.490	0.569
MTM-MidHigh	-0.215	0.029	0.319	0.384	0.437
MTM-High	-0.271	-0.006	0.238	0.215	0.352

(b) image captions

	STS-forum-[0, 1]	STS-forum-(1,2]	STS-forum-(2,3]	STS-forum-(3,4]	STS-forum-(4,5]
MTM-Low	0.475	0.112	-0.359	0.170	0.452
MTM-MidLow	0.587	0.165	-0.426	0.059	0.555
MTM-MidHigh	0.554	0.136	-0.239	-0.006	0.658
MTM-High	0.548	0.183	-0.079	0.016	0.688

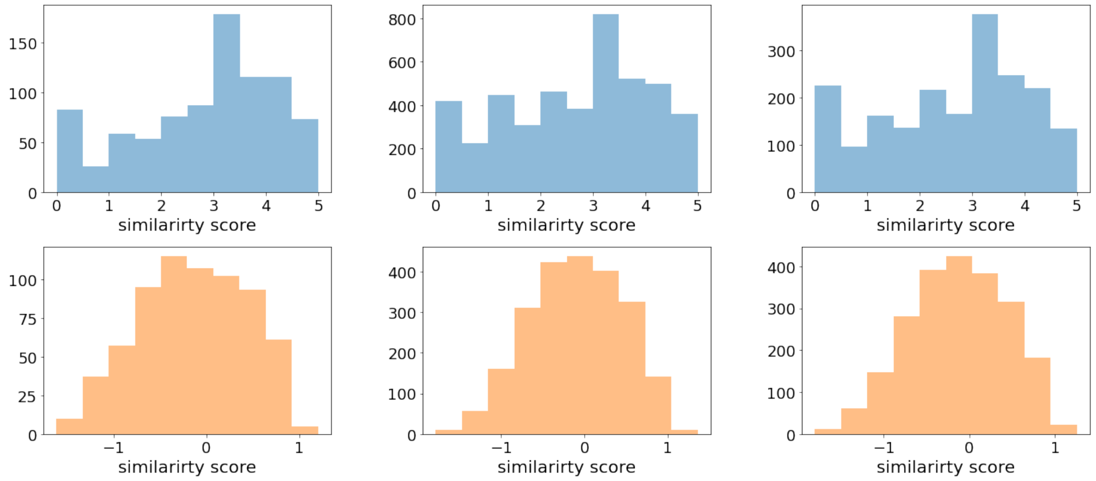
(c) forum

Figure 4.9 Spearman correlations between performance on subsets divided according to gold-standard similarity scores of each STS domain (news, forum, image captions) and MT Evaluation (MTE). The darker color represents the lower correlation (= the larger evaluation gap).

of different tenses as a pair of one sentence containing the verb whose POS tag is VBD and another sentence not containing such a verb. Besides, we defined examples with different named entities as a pair in which an entity list was obtained for each sentence, and both its text and the corresponding label in the list did not match exactly. We also made a list of pronouns (POS = PRON) for each sentence and regarded a pair in which the two lists of pronouns did not match precisely as examples of different pronouns.

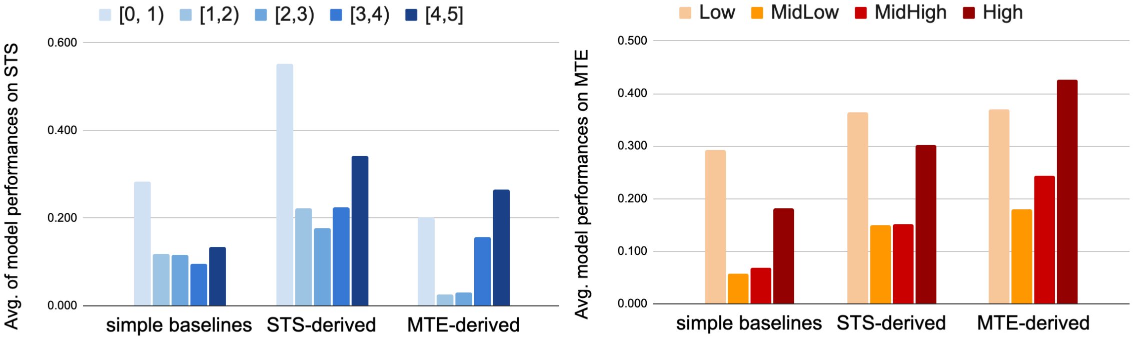
This figure shows that MTE annotations give relatively lower scores for tense differences than other elements, although the criteria are not explicitly stated. Thus, the lack of criteria for each label may result in unexpected annotation bias, contributing to the evaluation gap between STS and MTE.

Analysis: MTE-derived models fail high surface and low semantic similarity pairs We observed that the primary cause of the correlation gap regarding similarity granularity is the difference between MTE-derived models, such as BERTScore, and STS-derived models,



(a) Tense (b) Named entities (c) Pronouns

Figure 4.10 Histograms of #sentence pairs that have difference for each element (tense, named entities, and pronouns) in STS (upper) and MTE (bottom). X-axis represents similarity scores in each dataset.



(a) STS (b) MTE

Figure 4.11 Model performances on STS and MT Evaluation (MTE) for each similarity band.

such as SBERT. We then performed an additional analysis by dividing the models into three types: STS-derived model, MTE-derived model, and simple baselines like BoW, to investigate the tendency induced by this derivation. The type of each semantic similarity prediction model is as follows:

- Simple baselines: BoW, BoV, and the average of BERT embeddings
- STS-derived models: USE, SBERT, and SimCSE
- MTE-derived models: BERTScore variants

Fig. 4.11 shows three-type model performances on STS and MTE subsets for each similarity band. As seen from this figure, the MTE-derived model performs extremely poorly in the low similarity band of STS. In fact, for a simple sentence pair with high surface similarity (e.g., s1: "A woman is slicing tomato.", s2: "A man is slicing onion."). Such MTE-derived models tend to predict high scores for pairs with high surface similarity, even for examples that are easy for humans to discriminate, regardless of their semantic similarity. Our results suggest that this tendency is one reason for the evaluation gap.

4.5 Discussion and Conclusions

We have investigated the gap between evaluation scores on the STS benchmark dataset and those on the evaluation datasets for MT evaluation (MTE). We identified three factors contributing to this evaluation gap, namely, (i) sentence length distribution, (ii) vocabulary coverage ratio, and (iii) similarity granularity. These factors contributed to the evaluation gap, indicating that STS is not currently a directly applicable benchmark for evaluating semantic similarity.

Therefore, what should we do? We must continue to refine the evaluation of semantic similarity alone because of the significant demand for predicting semantic similarity (Sec. 4.1). Generic and task-specific semantic similarity measures have been proposed, which may have overfitted the evaluation dataset. Even though we evaluate a generic semantic similarity metric on only STS, it may not perform satisfactorily in real-world applications due to the influence of multiple factors described in this study. Wang et al. (2021) argued that the evaluation of existing semantic similarity models is biased toward STS and reported evaluation results on several datasets, including STS. One feasible approach is to evaluate and validate model performance on multiple datasets that engage in real-world tasks rather than just STS. In application-oriented semantic similarity metric tasks, e.g., MTE, the problems of evaluation and metric, which derived in the task~~, caused by task specific features~~ were shown. The MTE-derived metrics incorrectly scores high surface and low semantic similarity sentence pairs that could appear in real-world applications. Even though STS has some limitations as a "generic" semantic similarity evaluation, it can be used with a particular application task such as MTE to determine whether the model overfits task-specific tendency. Furthermore, ambiguity in the criteria for similarity scale in both semantic similarity evaluation tasks might lead to unexpected annotation bias. While there have been attempts to make the current criteria more concrete (Park et al., 2021), in recent years, research has emerged in various fields that employ a multi-dimension evaluation for models (Freitag et al., 2021; Singhal et al., 2022; Thoppilan et al., 2022). To mitigate the fear that recent high-performing models may inadvertently deceive humans, we need to be more rigorous in identifying phenomenon-by-phenomenon differences in various model evaluations, including semantic similarity evaluation.

Chapter 5

Evaluation (2) : Terminology-Focused Evaluation

5.1 Background: Lack of Benchmarks

When applying MT, there are many situations in which users want to translate a specific phrase or word into an appropriate representation. For example, when an English sentence, "We have conducted a shared task focusing on terminology consistency since 2021", translated into Japanese using the three off-the-shelf systems, the result is shown in Table 5.1.[1]. Three systems give two different translations for each phrase: 共有タスク and 共同作業 as the translation of "shared task", and 一貫性 and 統一 as the translation of "consistency", respectively. For the"shared task," the phrase 共有タスク is preferred for the translation, since the other phrase 共同作業 would give a different impression of its original meaning. As for the concept of "consistency," the meanings of both translated phrases (一貫性 and 統一) are correct. However, when appearing more than once in the same document, either translation phrase should be used consistently to facilitate better comprehension. As such, there is some motivation for translating a phrase in the source text into coherent and appropriate terminology. We denote such translation as "terminology translation" in this paper.

Previously, terminology translation was performed relatively easily by phrase-based statistical machine translation (PBSMT). Because PBSMT is a method to learn phrase-by-phrase correspondences from the training dataset statistically, we could thus teach a PBSMT model the phrase correspondences using some dictionaries. However, current mainstream NMT models cannot specify phrase correspondences. The reason is that NMT no longer

[1] They are the results as of 12/01/2022.

source	We have conducted a **shared task** focusing on terminology **consistency** since 2021.
System A	2021 年から用語の一貫性に重点を置いた共有タスクを実施しました。
System B	2021 年からの用語の統一に焦点を当てた共有タスクを実施しました。
System C	2021 年から用語の一貫性に焦点を当てた共同作業を行った。

Table 5.1 Motivated examples of terminology translation.

takes explicit phrase correspondences instead of enabling fluent generation via deep learning (in other words, it has become a black box).

In response to this problem, constrained NMT models have been proposed, which aim to output (always) the words or phrases specified as constraints. Existing constrained models can be broadly classified into two types. One model is the soft constrained model, in which constraints are imposed softly by input augmentation for training datasets. For example, Song et al. (2019) proposed a method that replaces constrained phrases in the source input with code-switching to the target language representation. Furthermore, Chen et al. (2020) concatenates a list of constraints directly to the inputs in the Transformer model with `[SEP]` tokens. The soft constrained model is an The soft constrained model is an augmentation of the training data, so the decoding speed is the same as the general MT architecture. However, this has the drawback that it does not always have output constraints. Recently, Kondo and Komachi (2022) proposed a model to address this drawback by using an automatic post-editing method (Susanto et al., 2020). The other model is the hard constrained model, which provides a signal to force the output of constraints during decoding. The hard constrained model has the advantage of always outputting constraints, but its disadvantage is increasing the decoding speed due to additional operations during decoding. Arthur et al. (2016) first established the basis for constrained models. In the last few years (Post and Vilar, 2018) proposed a model with improved decoding speed, but it is still noticeably slower than the soft constrained model. The decoding speed problem is one of the major concerns for real-world applications. The faster response speed is favored by users when considering applications in business scenes.

Though the proposals for the constrained models, there is little benchmark that provides a unified evaluation. There exists a dataset (Thompson et al., 2019) that is created for evaluating the terminology translation but has not been widely used for evaluating constrained models. In addition, Jiang et al. (2022) proposed a document-level evaluation automatic metric, but evaluation frameworks that focus on terminology consistency have been scarcely conducted. Therefore, we proposed the Restriction Translation task in the WAT workshop, created a new evaluation dataset with a scientific domain, one of the real-world application-oriented domains, and set up a simple automatic metric.

5.2 Restricted Translation Task

5.2.1 Task Setting

For the terminology-consistent translation, we re-designed ASPEC scientific-domain translation task (Nakazawa et al., 2016). Restricted vocabulary list containing scientific technical terms in a target language. We do not accept other terms that are semantically similar to the specified ones.

Let $\mathscr{D} = \{(\boldsymbol{x}_n, \boldsymbol{y}_n)\}_{n=1}^{N}$ be an dataset that consists of a pair of sentences $(\boldsymbol{x}_n, \boldsymbol{y}_n)$, where $\boldsymbol{x}_n$ and $\boldsymbol{y}_n$ represent the n-th source and target sentences, respectively. N denotes the number of samples in $\mathscr{D}$. For the terminology consistency evaluation, we need additional information of restricted vocabulary list in the dataset $\mathscr{D}$. We have a list of restricted vocabulary (term pairs) $\mathscr{T} = \{(\boldsymbol{q}_k, \boldsymbol{r}_k)\}_{k=1}^{K}$, such as a bilingual dictionary, where $\boldsymbol{q}_k$ and $\boldsymbol{r}_k$ represent the k-th source and target term pair, and K represents the number of term pairs in $\mathscr{T}$. A sentence pair may contain multiple term pairs. Note that task participants are only given a restricted vocabulary list that is not aligned to a sentence pair. Different restricted vocabulary lists are given for the training, valid, and test datasets, respectively. We evaluate whether all given term pairs in the restricted vocabulary list are correctly translated in whole test dataset.

5.2.2 Dataset

We need to annotate term pairs (in the restricted vocabulary) for each instance to construct a dataset for the restricted translation task. This process can be decomposed into three steps: 1) extracting technical terms in each language, 2) aligning the source and target terms extracted in Step 1, and 3) final check.

We constructed data for English–Japanese language pairs (En–Ja and Ja–En) in 2021 and Chinese–Japanese language pairs (Zh–Ja and Ja–Zh) in 2022, respectively. For the En–Ja and Ja–En directions, steps 1 and 2 were executed in parallel using manual annotation. During these steps, we asked ten bilingual speakers to extract and annotate technical term/phrase pairs for each example. Each annotator annotated around 540 sentence pairs and was paid 10,000 yen as compensation. The final check was done using an additional bilingual speaker. In the final check, we corrected extracted terminology boundaries that were inconsistent between the annotators and corrected alignment errors.

Executing all such procedures manually was a high cost. That is, we attempted to automate the term extraction in Zh–Ja and Ja–Zh datasets by applying `pytermextract`[2] in step 1. In addition, we attempted to automate the alignment acquisition in step 2. But due

[2] http://gensen.dl.itc.u-tokyo.ac.jp/pytermextract/

	En-Ja (#phrase, #char)	Ja-En (#phrase, #word)	Zh-Ja (#phrase, #char)	Ja-Zh (#phrase, #char)
Dev	(2.8, 16.4)	(2.8, 6.6)	(1.2, 4.7)	(1.2, 3.8)
Devtest	(3.2, 18.2)	(3.2, 7.3)	(1.5, 5.5)	(1.5, 4.5)
Test	(3.3, 18.1)	(3.2, 7.4)	(1.4, 5.2)	(1.4, 4.2)

Table 5.2 An average number of annotated phrases and words/characters per sentence pair. Since Japanese and Chinese have no explicit word delimiter, the character count is used as a phrase length indicator, whereas word count is used for English.

to inconsistencies in technical terms boundaries, we failed to obtain the correct alignment between automatically extracted technical terms in each language. So we continued to perform alignments in these directions manually. We asked three bilingual speakers to make the alignments of the terms. Finally, to add the alignment quality information, we asked two additional bilingual annotators to give a translation of 0-100 scores for each term pair. We show in Table 5.2 the average number of term pairs annotated for each sentence pair in the dataset.

5.2.3 Evaluation

Automatic metric: Combination of Exact Match and BLEU BLEU, a standard automatic evaluation metric, looks at n-gram overlap and does not have a feature that explicitly considers terminology. In this study, we designed an automatic evaluation metric that considers terminology errors that we focus on. Our metric is a simple combination of exact matches with given target terms and BLEU scores. We aim to measure both terminology and translation quality in a single metric. For evaluation, we filter out system outputs that do not include terminology with exact matching and then compute BLEU scores. With this filtering, we can consider the brief penalty of BLEU as a demerit of terminology error in the final score calculation.

Human evaluation: Source-based Direct Assessment We used the simple automatic metric described above as a low-cost method to obtain the system's ratings. However, a well-known problem is that automatic evaluation such as BLEU alone does not adequately measure translation quality. To assess the translation quality more accurately, we also performed a human evaluation once the automatic evaluation results are obtained. We adopted source-based Direct Assessment (Cettolo et al., 2017; Federmann, 2018), which asked bilingual evaluators to rate how adequate the output was on 0–100 scores given the source and the corresponding system output.

5.3 Results: What is the Best Model?

Table 5.3 shows the results for the Restricted Translation task in 2021 and 2022. Overall, the combination of soft and hard constrained models (Hard+Soft const) submitted by the NTT team in 2021 performed best (Chousa and Morishita, 2021). This model combines a pre-trained model with large and filtered augmented data and constrained decoding to output terms in the given restricted vocabulary. This model could always generate specified constraints due to the effect of the hard const model. The use of filtered augmented data allowed the model to produce high-quality output comparable to the reference.

Besides, in 2022, the TMU team proposed a method that complements the shortcomings of the soft constrained model with the post-editing model (Kondo and Komachi, 2022). Although this model was slightly inferior to the combination of soft and hard constrained models, it could also output 100% of the constraints by adding omitted constraints using the post-editing model. Another advantage of this model is that it avoids the drawbacks of the hard constrained model, which is slower at decoding speed. [3]. Since the decoding speed of MT is usually an essential issue in real-world applications, this model is a promising substitute for the best model.

5.4 Analysis: Validity of Automatic Metric

Correlation between automatic and human evaluation Below, we investigate the validity of our proposed automatic evaluation metric. To evaluate the validity of automatic metrics, we measured the correlation between our proposed metrics and human evaluation scores following the previous studies (Banerjee and Lavie, 2005; Mathur et al., 2020; Popović, 2017). We measured Spearman correlations between human evaluation (source-based DA) and three automatic metrics: BLEU-only, exact-match-only, and our metric (i.e., a combination of exact-match and BLEU). The results of the Spearman correlations in 12 submitted systems over a 2-year period are shown in Table 5.4. This result indicates that our metric resulted in the highest correlation. Noticeably, although source-based DA did not explic-

[3] Note that there is no rigorous comparison of decoding speed between the Hard+Soft const model by the NTT team and the soft const+post-edit model by the TMU team. However, the TMU team paper shows that the decoding speed of their model (En-Ja: 0.115, Ja-En: 0.126) is faster than that of the NTT-team ablation model excluding the hard constrained part (En-Ja: 0.221, Ja-En: 0.228).

		En-Ja		Ja-En	
	System	Auto	Human	Auto	Human
2021	Hard+Soft const (NTT)	57.2	77.5	44.1	75.6
	Soft const (NHK)	33.9	74.1	37.5	73.9
	Hard+Soft const (NICT)	28.8	73.6	31.8	72.1
	Hard const (TMU)	-	-	22.6	50.2
	(human ref.)	-	73.4	-	74.1
2022	Soft const+Post-edit (ensemble) (TMU)	52.7	76.4	40.8	74.1
	Soft const+Post-edit (TMU)	50.5	76.6	38.1	72
	Soft const (TMU)	37.6	74.9	23.0	73.3
	(human ref.)	-	76.6	-	74.7

Table 5.3 Result of Restricted Translation Task 2021-2022. “Auto” means the evaluation scores with our automatic metric (exact match + BLEU). “Human” means the evaluation scores with Source-based Direct Assesment (DA).

Automatic metrics	Corr.
↔ BLEU	0.795
↔ Exact match	0.498
↔ Ours (Exact+BLEU)	0.836

Table 5.4 Spearman correlations between three automatic metrics (BLEU, exact match, and the combination of exact match and BLEU) and human evaluation (source-based DA).

itly judge terminology errors, the correlation of BLEU scores via exact-match filtering was higher than that of BLEU-only scores. This suggests that integrating terminology-focused evaluation is closer to human translation evaluation.

Reappearance of terminology Indeed, exact matching does not strictly consider reappearance of terms. The reappearance is, e.g., that term A appears twice, and term B including term A appears. Here we analyze how our automatic metric performs on such examples.

First, how many examples containing the reappearance of terms occur? Fig 5.1 shows the percentage of occurrences of reappearance examples in the reference and system outputs that are submitted in 2021. Here, we consider examples where at least one of the terms in RV matched more than once as the reappearance examples. As shown in Fig. 5.1, the reappearance of terms occurs at about 10% in the reference and all systems.

We next investigate how many real erroneous examples are in the reappearance examples. Let us compare the reappearance of an example between the reference and the system output.

	Ja–En		En–Ja	
	#under	#over	#under	#over
Hard+Soft const (NTT)	16	41	19	31
Soft const (NHK)	28	46	36	61
Hard+Soft const (NICT)	17	107	38	86

Table 5.5 Number of examples of under-generation and over-generation for given terms.

Concretely, we focus on under-and over-generation errors. In this analysis, we count those examples that have the reappearance in reference but do not have the reappearance in the system output as under-generation errors. Similarly, we count examples that do not have the reappearance in the reference but have the reappearance in the system output as over-generation. Note that this is a toy setting where we only check the appearance change of a particular term between twice and once.

Table 5.5 shows the number of over- and under-generation errors for each system output submitted in 2021. In Table 5.5, we can observe each system's shortcomings in detail. For example, the soft constrained NHK system tends to under-generate terms compared with the other systems. In contrast, the combination (hard and soft constrained) NICT system, which is inferior to the same NTT architecture, tends to over-generate terms. Thus, over- or under-generation in the reappearance examples was indeed observed; however, the trend in the total error count is the same as the system ranking obtained with the automatic metric. One reason is that the BLEU part of our automatic metric should penalize large under/over-generation. Our combination metric did not significantly affect these errors and thus resulted in a high correlation with human evaluation.

At present, the errors caused by reappearance are not significant enough to affect the ranking results of the system. However, the current metric may not be able to detect errors such as the number of term appearances being correct but their positions needing to be corrected. Our future work is to introduce a more rigorous evaluation metric that can account for a term position.

5.5 Analysis: What examples are rated low in the human evaluation?

With the task results, we have a question; what examples fail to translate correctly to the current top system? To investigate this, we check examples with lower human evaluation

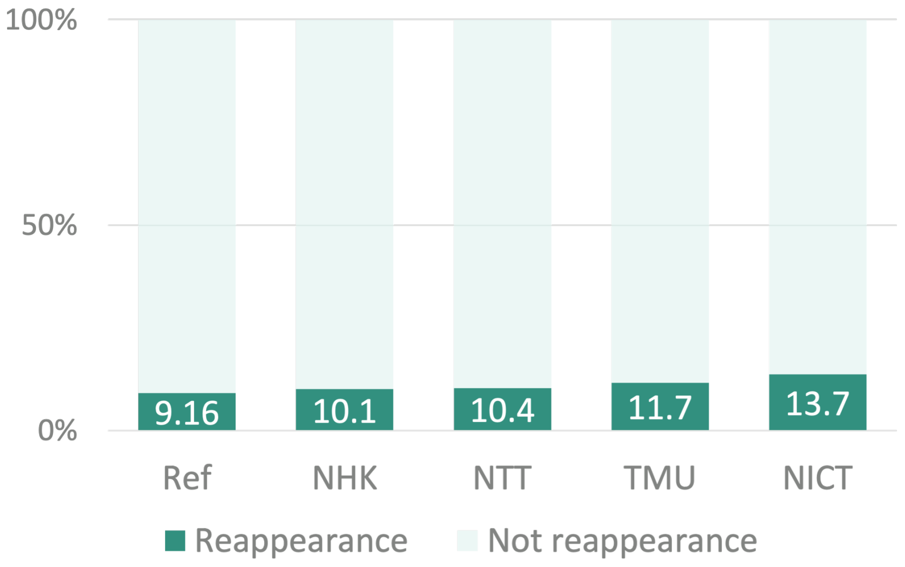


Figure 5.1 Ratio of the numbers of examples that have reappearance of terms. *Reappearance* represents the ratio of examples where at least one of the terms in the RV matched more than once.

scores in the best system (Hard+Soft const. in 2021). Then, we found four error types affected by the quality of the different elements.

1. **Annotation quality**: examples that lack some terms not included in restricted vocabulary list
2. **Translation model quality**: examples that are mistranslated other than terminology
3. **Human evaluation quality**: examples that seem to be annotated unfairly with lower scores due to lack of domain knowledge in annotators
4. **Original translation dataset quality**: examples that seem to overfit ASPEC reference and be annotated with lower scores with source-based evaluation

Type 2 (Translation model quality) and 3 (Human evaluation quality) were the most common of the four error types. More detailed examples of each type are shown in Table 5.6. The first type is an error caused by the annotation quality of the dataset, i.e., restricted vocabulary quality. In example 1, a specific proper noun 北国新聞社 is translated as 北新聞社 by the model. If the phrase 北国新聞社 corresponding “Hokkoku Newspaper Company” is specified as a constraint, this error can be avoided on the current hard constrained model. The second type is a error due to translation model quality. Here, we consider as this type of error all mistranslation examples that are not regarded as terminology errors such as the example 1. In example 2, the phrase "it is omnipresent like God" included in the quotation marks is translated as "God のように不思議である" which is a misinterpretation of the original meaning. So far, there have been many mistranslations considering longer sentences with complex structures or quotation marks. The third type is the error due to human evaluation quality (although whether this can be called "error" is debatable). Example 3 contains a large number of technical terms as the restricted vocabulary (RV). Even though all of them can be output by the constrained model, the human evaluation score is 25 points, which is the lower score among 0-100 scale. One reason is that evaluators were allowed to see only the source sentence (Src) and the output of the system (Out). In other words, they are not given any instruction related to terminology, requiring high domain knowledge of technical terms to evaluate the model correctly. Future improvements could include adopting a different human evaluation method (Multi-dimensional Quality Metrics; Freitag et al., 2021) or incorporating terminology-related criteria into the current framework. The fourth is an example in which a low score is given despite the original reference translation being almost reproduced. In Example 4, the system output sentence reproduces the same meaning as the reference, but misses a portion of "(behind) other countries" when compared to the source meaning. While not as common as in Types 2 and 3, such a few examples were found. One possible reason is that the model overfits ASPEC's reference style leaving out some parts of data, leading to lower scores under the evaluation focusing on corresponding the source meaning.

5.6 Related Work

There is a long history of MT tasks focusing on terminology in real-world application scenarios. Among them, a patent translation task was established in 2013 (Fujii et al., 2008), indicating the demand of MT for specialized technical documents. In recent years, as the use of MT in real-world applications has expanded, there is a need to develop a new terminology-focused task. To meet this need, another terminology task (Alam et al., 2021b) was held at WMT21 as well as our Restricted Translation task. In this task, a terminology-specific evaluation metric (Alam et al., 2021a) was also proposed, which aligns between the source terms and the output terms and focuses on the location of the terms. Also, in WMT22, the terminology task have been adopted as a sub-track of the biomedical translation task (Neves et al., 2022). Thus, there have been some attempts to establish an evaluation framework in the form of task proposals in response to the demand for a system that can correctly translate

Example 1 (annotation quality)	
Src	This paper presents security card renewal construction of Hokkoku Newspaper Company.
Ref	北国新聞社のセキュリティカード更新工事を紹介した。
Out	北新聞社のセキュリティカード更新工事について紹介した。(**score: 37**)
RVs	[]
Example 2 (translation model quality)	
Src	“Ubiquitous” is the meaning of “it is omnipresent like God”.
Ref	ユビキタス（ｕｂｉｑｕｉｔｏｕｓ）は「神のように遍在する」という意味である。
Out	「ユビキタス（ｕｂｉｑｕｉｔｏｕｓ）」とは，「Ｇｏｄのように不思議である」という意味である。(**score: 33**)
RVs	[“ユビキタス（ｕｂｉｑｕｉｔｏｕｓ）”]
Example 3 (human evaluation quality)	
Src	The titled facility consists of equipment such as flue gas desulfurization, stack-gas desulfurization drainage, total drainage, ash disposal, coal transportation, and coal receiving.
Ref	標記設備は，排煙脱硫，排脱排水，総合排水，灰処理，揚運炭，石炭の湾受け入れなどの設備から成る
Out	標記設備は排煙脱硫，排脱排水，総合排水，灰処理，揚運炭，石炭の湾受け入れ等の設備からなる。(**score: 25**)
RVs	[“標記設備”, “排煙脱硫”, “排脱排水”, “総合排水”, “灰処理”, “揚運炭”, “石炭の湾受け入れ”]
Example 4 (original translation dataset quality)	
Src	Japan is behind other countries in taking measures against misconduct.
Ref	日本では不正行為への対応が遅れている。
Out	日本は不正行為への対応が遅れている。(**score: 58**)
RVs	[“不正行為”]

Table 5.6 Examples of outputs of the top system in the Restricted Translation Task.

terminology. In this chapter, we describe a scientific-domain terminology task and proposed a simple metric for English−Japanese and Chinese−Japanese language pairs.

5.7 Conclusions

We established Restricted Translation task as a evaluation benchmark for current constrained models. For this task, we construct English-Japanese and Chinese-Japanese datasets from original ASPEC corpus and proposed a simple automatic metric which capture both translation performance and consistency. The task was actually organized in a workshop, and we pointed out some aspects to consider in future terminology-focused translation. For future work, we need to consider how to increase human evaluation quality. Possible approaches to increase the quality of human evaluation is to provide domain knowledge to annotators to show given terms or to apply a new human evaluation framework.

Chapter 6

Conclusions

In this study, we summarize two approaches to the current MT issues for real-world applications: (1) MT design using multiple optional architectures and (2) improvement of the evaluation framework to find mistranslations that may lead to miscommunications. We then present three real examples of these approaches:

- **Multi-Dialect Translation**: we demonstrated an example of designing MT for a real-world problem of processing various dialects in low-resource scenarios using multilingual, character-level, and fixed-order translation.
- **Comprehensive Analysis of Semantic Metrics for MT**: we investigated potential problems concerning automatic evaluation metrics for semantic similarity and identified factors that influence evaluation results.
- **Proposal of Terminology-focused MT Task**: we introduced an evaluation framework for MT models that handles terminology and identified issues with the current MT model and evaluation frameworks.

This research aims to construct a customizable MT system to individual users' desires that do not induce miscommunications. Fulfilling this goal makes the following developments possible in the real world. The first is the further improvement of familiarity and practicality of MT. By providing an MT system that combines optional architectures to handle more user-specific situations, we can promote readily high-resolution interlingual communications for everyone worldwide. The second is preventing misunderstandings and miscommunications caused by MT, which can occur with the current MT system as it is. In order to realize MT is acceptable for professionals, we focus on the two evaluation frameworks. One is the semantic evaluation to accept semantically correct outputs and reject semantically wrong outputs appropriately. The another is the terminology evaluation to subtract for inappropriate terminology even if it is semantically acceptable. With these works, we have yet to ultimately create a silver bullet to prevent MT from impeding miscommunications. However, exploring appropriate MT design and evaluation frameworks will help real-world MT applications that enrich people's lives.

Finally, we think that there are two policies for future real-world application-oriented MT research. As we have discussed, the first is to develop an appropriate evaluation framework to make professional-level judgments and a high-quality MT system to pass such evaluations. This policy should be continued, but it is not easy to realize it perfectly. Another practical way is to make users aware of what MT can/cannot do or to make users correctly recognize the errors by, for example, adding functions to present wrong errors as “wrong”. In conclusion, we must give insight into MT users' preferences and capabilities in real-world applications.

References

Agirre, E., Cer, D., Diab, M., and Gonzalez-Agirre, A. (2012). SemEval-2012 task 6: A pilot on semantic textual similarity. In *Proceedings of the First Joint Conference on Lexical and Computational Semantics –Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation*, pages 385–393.

Alam, M. M. I., Anastasopoulos, A., Besacier, L., Cross, J., Gallé, M., Koehn, P., and Nikoulina, V. (2021a). On the evaluation of machine translation for terminology consistency. *arxiv preprint arXiv:2106.11891.*

Alam, M. M. I., Kvapilíková, I., Anastasopoulos, A., Besacier, L., Dinu, G., Federico, M., Gallé, M., Jung, K., Koehn, P., and Nikoulina, V. (2021b). Findings of the WMT shared task on machine translation using terminologies. In *Proceedings of the Sixth Conference on Machine Translation*, pages 652–663. Association for Computational Linguistics.

Arivazhagan, N., Bapna, A., Firat, O., Lepikhin, D., Johnson, M. G., Krikun, M., Chen, M. X., Cao, Y., Foster, G., Cherry, C., Macherey, W., Chen, Z., and Wu, Y. (2019). Massively multilingual neural machine translation in the wild: Findings and challenges. *arXiv preprint arXiv:1907.05019.*

Arthur, P., Neubig, G., and Nakamura, S. (2016). Incorporating discrete translation lexicons into neural machine translation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1557–1567. Association for Computational Linguistics.

Banerjee, S. and Lavie, A. (2005). METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72. Association for Computational Linguistics.

Bojar, O., Graham, Y., and Kamran, A. (2017). Results of the wmt17 metrics shared task. In *Proceedings of the Second Workshop on Machine Translation*, pages 489–513.

Bowman, S. R. and Dahl, G. (2021). What will it take to fix benchmarking in natural language understanding? In *Proceedings of the 19th Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4843–4855.

Callison-Burch, C., Fordyce, C., Koehn, P., Monz, C., and Schroeder, J. (2007). (meta-) evaluation of machine translation. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 136–158. Association for Computational Linguistics.

Castillo, J. and Estrella, P. (2012). Semantic textual similarity for MT evaluation. In *Proceedings of the Seventh Workshop on Statistical Machine Translation*, pages 52–58.

Caswell, I., Chelba, C., and Grangier, D. (2019). Tagged back-translation. In *Proceedings of the Fourth Conference on Machine Translation (Volume 1: Research Papers)*, pages 53–63. Association for Computational Linguistics.

Cer, D., Diab, M., Agirre, E., Lopez-Gazpio, I., and Specia, L. (2017). SemEval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation. In *Proceedings of the 11th International Workshop on Semantic Evaluation*, pages 1–14.

Cer, D., Yang, Y., Kong, S.-y., Hua, N., Limtiaco, N., St. John, R., Constant, N., Guajardo-Cespedes, M., Yuan, S., Tar, C., Strope, B., and Kurzweil, R. (2018). Universal sentence encoder for English. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 169–174.

Cettolo, M., Federico, M., Bentivogli, L., Niehues, J., Stüker, S., Sudoh, K., Yoshino, K., and Federmann, C. (2017). Overview of the IWSLT 2017 evaluation campaign. In *Proceedings of the 14th International Conference on Spoken Language Translation*, pages 2–14. International Workshop on Spoken Language Translation.

Chen, D., Fisch, A., Weston, J., and Bordes, A. (2017). Reading Wikipedia to answer open-domain questions. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, pages 1870–1879.

Chen, G., Chen, Y., Wang, Y., and Li, V. O. (2020). Lexical-constraint-aware neural machine translation via data augmentation. In Bessiere, C., editor, *Proceedings of the 24th International Joint Conference on Artificial Intelligence*, pages 3587–3593. International Joint Conferences on Artificial Intelligence Organization.

Chousa, K. and Morishita, M. (2021). Input augmentation improves constrained beam search for neural machine translation: NTT at WAT 2021. In *Proceedings of the 8th Workshop on Asian Translation (WAT2021)*, pages 53–61. Association for Computational Linguistics.

Chu, C., Dabre, R., and Kurohashi, S. (2018). A comprehensive empirical comparison of domain adaptation methods for neural machine translation. *Journal of Information Processing*, 26:529–538.

Chuang, Y.-S., Dangovski, R., Luo, H., Zhang, Y., Chang, S., Soljacic, M., Li, S.-W., Yih, W.-t., Kim, Y., and Glass, J. (2022). DiffCSE: Difference-based contrastive learning for sentence embeddings. In *Proceedings of the 20th Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1–12.

Conneau, A., Kiela, D., Schwenk, H., Barrault, L., and Bordes, A. (2017). Supervised learning of universal sentence representations from natural language inference data. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 670–680.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 17th Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Durmus, E., Ladhak, F., and Hashimoto, T. (2022). Spurious correlations in reference-free evaluation of text generation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, pages 1443–1454.

Farajian, M. A., Lopes, A. V., Martins, A. F. T., Maruf, S., and Haffari, G. (2020). Findings of the WMT 2020 shared task on chat translation. In *Proceedings of the Fifth Conference on Machine Translation*, pages 65–75. Association for Computational Linguistics.

Federmann, C. (2018). Appraise evaluation framework for machine translation. In *Proceedings of the 27th International Conference on Computational Linguistics: System Demonstrations*, pages 86–88. Association for Computational Linguistics.

Freitag, M., Foster, G., Grangier, D., Ratnakar, V., Tan, Q., and Macherey, W. (2021). Experts, errors, and context: A large-scale study of human evaluation for machine translation. *Transactions of the Association for Computational Linguistics*, 9:1460–1474.

Fujii, A., Utiyama, M., Yamamoto, M., and Utsuro, T. (2008). Overview of the patent translation task at the ntcir-7 workshop. In *Proceedings of the Seventh NTCIR Workshop Meeting*, pages 389–400.

Gao, L., Dai, Z., and Callan, J. (2021a). COIL: Revisit exact lexical match in information retrieval with contextualized inverted list. In *Proceedings of the 19th Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3030–3042.

Gao, T., Yao, X., and Chen, D. (2021b). SimCSE: Simple contrastive learning of sentence embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6894–6910.

Gao, Y., Nikolov, N. I., Hu, Y., and Hahnloser, R. H. (2020). Character-level translation with self-attention. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1591–1604. Association for Computational Linguistics.

Giorgi, J., Nitski, O., Wang, B., and Bader, G. (2021). DeCLUTR: Deep contrastive learning for unsupervised textual representations. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 879–895.

Goto, I., Chow, K. P., Lu, B., Sumita, E., and Tsou, B. (2013). Overview of the patent machine translation task at the ntcir-10 workshop. In *Proceedings of the 10th NTCIR Conference*, pages 260–286.

Graham, Y., Baldwin, T., and Mathur, N. (2015). Accurate evaluation of segment-level machine translation metrics. In *Proceedings of the 13th Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1183–1191. Association for Computational Linguistics.

Gu, J., Hassan, H., Devlin, J., and Li, V. O. (2018). Universal neural machine translation for extremely low resource languages. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 185–194. Association for Computational Linguistics.

Guggilla, C. (2016). Discrimination between similar languages, varieties and dialects using cnn- and lstm-based deep neural networks. In *Proceedings of the Third Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 186–194. The COLING 2016 Organizing Committee.

Gülçehre, Ç., Ahn, S., Nallapati, R., Zhou, B., and Bengio, Y. (2016). Pointing the unknown words. *arxiv preprint arXiv:1603.08148*.

Hassan, H., Elaraby, M., and Tawfik, A. Y. (2017). Synthetic data for neural machine translation of spoken-dialects. In *Proceedings of the 14th International Conference on Spoken Language Translation*, pages 82–89, Tokyo, Japan. International Workshop on Spoken Language Translation.

Heafield, K. (2011). KenLM : Faster and Smaller Language Model Queries. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 187–197. Association for Computational Linguistics.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8):1735–1780.

Hokamp, C. and Liu, Q. (2017). Lexically constrained decoding for sequence generation using grid beam search. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1535–1546. Association for Computational Linguistics.

Honnet, P., Popescu-Belis, A., Musat, C., and Baeriswyl, M. (2018). Machine Translation of Low-Resource Spoken Dialects: Strategies for Normalizing Swiss German. In *Proceedings of 11th edition of the Language Resources and Evaluation Conference*, pages 3781–3788. European Language Resources Association.

Jiang, Y., Liu, T., Ma, S., Zhang, D., Yang, J., Huang, H., Sennrich, R., Cotterell, R., Sachan, M., and Zhou, M. (2022). BlonDe: An automatic evaluation metric for document-level machine translation. In *Proceedings of the 20th Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1550–1565. Association for Computational Linguistics.

Johnson, M., Schuster, M., Le, Q. V., Krikun, M., Wu, Y., Chen, Z., Thorat, N., Viégas, F., Wattenberg, M., Corrado, G., Hughes, M., and Dean, J. (2016). Google's Multilingual Neural Machine Translation System: Enabling Zero-Shot Translation. *Proceedings of Transactions of the Association for Computational Linguistics*, 5:339–351.

Karpukhin, V., Oguz, B., Min, S., Lewis, P., Wu, L., Edunov, S., Chen, D., and Yih, W.-t. (2020). Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pages 6769–6781.

Kincaid, J. P., Fishburne, R. P. J., Rogers, R. L., , and Chissom, B. S. (1975). Derivation of new readability formulas (automated readability index, fog count and flesch reading ease formula) for navy enlisted personnel.

Klein, G., Kim, Y., Deng, Y., Senellart, J., and Rush, A. M. (2017). OpenNMT: Open-Source Toolkit for Neural Machine Translation. *arXiv preprint arXiv:1701.02810.*

Kocmi, T., Federmann, C., Grundkiewicz, R., Junczys-Dowmunt, M., Matsushita, H., and Menezes, A. (2021). To ship or not to ship: An extensive evaluation of automatic metrics for machine translation. In *Proceedings of the Sixth Conference on Machine Translation*, pages 478–494. Association for Computational Linguistics.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran Mit, C., Zens, R., Dyer, C., Bojar, O., Constantin, A., and Herbst, E. (2007). Moses: Open Source Toolkit for Statistical Machine Translation. In *Proceedings of 45th Annual Meeting of the Association for Computational Linguistics*, pages 177–180. Association for Computational Linguistics.

Kondo, S. and Komachi, M. (2022). TMU NMT system with automatic post-editing by multi-source Levenshtein transformer for the restricted translation task of WAT 2022. In *Proceedings of the 9th Workshop on Asian Translation*, pages 51–58. International Conference on Computational Linguistics.

Kudo, T. and Richardson, J. (2018). SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (System Demonstrations)*, pages 66–71. Association for Computational Linguistics.

Kumagai, Y. (2016). Developing the linguistic atlas of japan database and advancing analysis of geographical distributions of dialects. In *The future of dialects: Selected papers from Methods in Dialectology XV.*, pages 333–362. Berlin: Language Science Press.

Lakew, S. M., Mauro, C., and Federico, M. (2018). A Comparison of Transformer and Recurrent Neural Networks on Multilingual Neural Machine Translation. *arXiv preprint arXiv:1806.06957.*

Lan, W. and Xu, W. (2018). Neural network models for paraphrase identification, semantic textual similarity, natural language inference, and question answering. In *Proceedings the 24th International Conference on Computational Linguistics*, pages 3890–3902.

Läubli, S., Castilho, S., Neubig, G., Sennrich, R., Shen, Q., and Toral, A. (2020). A set of recommendations for assessing human–machine parity in language translation. *Journal of Artificial Intelligence Research*, 67.

Lee, J., Cho, K., and Hofmann, T. (2017). Fully character-level neural machine translation without explicit segmentation. *Transactions of the Association for Computational Linguistics*, 5:365–378.

Li, B., Zhou, H., He, J., Wang, M., Yang, Y., and Li, L. (2020). On the sentence embeddings from pre-trained language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pages 9119–9130.

Li, Y., Suzuki, J., Morishita, M., Abe, K., Tokuhisa, R., Brassard, A., and Inui, K. (2022). Chat translation error detection for assisting cross-lingual communications. In *Proceedings of the 3rd Workshop on Evaluation and Comparison of NLP Systems*, pages 88–95. Association for Computational Linguistics.

Liang, Y., Meng, F., Chen, Y., Xu, J., and Zhou, J. (2021). Modeling bilingual conversational characteristics for neural chat translation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 5711–5724. Association for Computational Linguistics.

Liu, C., Wang, R., Liu, J., Sun, J., Huang, F., and Si, L. (2021). DialogueCSE: Dialogue-based contrastive learning of sentence embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 2396–2406.

Logeswaran, L. and Lee, H. (2018). An efficient framework for learning sentence representations. In *Proceedings of the Sixth International Conference on Learning Representations*, pages 1–16.

Ma, Q., Wei, J., Bojar, O., and Graham, Y. (2019). Results of the WMT19 metrics shared task: Segment-level and strong MT systems pose big challenges. In *Proceedings of the Third Workshop on Machine Translation*, pages 62–90.

Mathur, N., Baldwin, T., and Cohn, T. (2020). Tangled up in BLEU: Reevaluating the evaluation of automatic machine translation evaluation metrics. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4984–4997.

Meftouh, K., Harrat, S., Jamouss, S., Abbas, M., and Smaili, K. (2015). Machine Translation Experiments on PADIC: A Parallel Arabic DIalect Corpus. In *Proceedings of 29th Pacific Asia Conference on Language, Information and Computation*, pages 26–34. Association for Computational Linguistics.

Mille, S., Dhole, K., Mahamood, S., Perez-Beltrachini, L., Gangal, V., Kale, M., van Miltenburg, E., and Gehrmann, S. (2021). Automatic construction of evaluation suites for natural language generation datasets. In *Proceedings of 35th Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1)*, pages 1–14.

Morishita, M., Suzuki, J., and Nagata, M. (2022). Domain adaptation of machine translation with crowdworkers. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing (Industry Track)*.

Nakazawa, T., Mino, H., Goto, I., Dabre, R., Higashiyama, S., Parida, S., Kunchukuttan, A., Morishita, M., Bojar, O., Chu, C., Eriguchi, A., Abe, K., Oda, Y., and Kurohashi, S. (2022). Overview of the 9th workshop on Asian translation. In *Proceedings of the 9th Workshop on Asian Translation*, pages 1–36. International Conference on Computational Linguistics.

Nakazawa, T., Nakayama, H., Ding, C., Dabre, R., Higashiyama, S., Mino, H., Goto, I., Pa Pa, W., Kunchukuttan, A., Parida, S., Bojar, O., Chu, C., Eriguchi, A., Abe, K., Oda, Y., and Kurohashi, S. (2021). Overview of the 8th workshop on Asian translation. In *Proceedings of the 8th Workshop on Asian Translation*, pages 1–45. Association for Computational Linguistics.

Nakazawa, T., Yaguchi, M., Uchimoto, K., Utiyama, M., Sumita, E., Kurohashi, S., and Isahara, H. (2016). ASPEC: Asian scientific paper excerpt corpus. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation*, pages 2204–2208. European Language Resources Association (ELRA).

Nakov, P. and Tiedemann, J. (2012). Combining Word-Level and Character-Level Models for Machine Translation Between Closely-Related Languages. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 301–305. Association for Computational Linguistics.

National Institute for Japanese Language and Linguistics (1980). *"National Dialect Discourse Database Collection (*全国方言談話データベース 日本のふるさと言葉集成*)"*. Kokushokankokai Inc. (国書刊行会).

Nerbonne, J. and Kretzschmar, W. A. (2011). Dialectometry++. *Literary and Linguistic Computing*, 28(1):2–12.

Neves, M., Yepes, A. J., Siu, A., Roller, R., Thomas, P., Navarro, M. V., Yeganova, L., Wiemann, D., Nunzio, G. M. D., Vezzani, F., Gerardin, C., Bawden, R., Estrada, D. J., Lima-López, S., Farré-Maduell, E., Krallinger, M., Grozea, C., and Névéol, A. (2022). Findings of the wmt 2022 biomedical translation shared task: Monolingual clinical case reports. In *Proceedings of the Seventh Conference on Machine Translation*, pages 694–723. Association for Computational Linguistics.

Östling, R. and Tiedemann, J. (2017). Continuous multilinguality with language vectors. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 644–649. Association for Computational Linguistics.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002a). Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318. Association for Computational Linguistics.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002b). Bleu: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, page 311−318. Association for Computational Linguistics.

Park, S., Moon, J., Kim, S., Cho, W. I., Han, J., Park, J., Song, C., Kim, J., Song, Y., Oh, T., Lee, J., Oh, J., Lyu, S., Jeong, Y., Lee, I., Seo, S., Lee, D., Kim, H., Lee, M., Jang, S., Do, S., Kim, S., Lim, K., Lee, J., Park, K., Shin, J., Kim, S., Park, L., Oh, A., Ha, J.-W., and Cho, K. (2021). Klue: Korean language understanding evaluation. *arxiv preprint arXiv:2105.09680*, pages 1–76.

Popović, M. (2015). chrF: character n-gram F-score for automatic MT evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 392–395. Association for Computational Linguistics.

Popović, M. (2017). chrF++: words helping character n-grams. In *Proceedings of the Second Conference on Machine Translation*, pages 612–618. Association for Computational Linguistics.

Post, M. and Vilar, D. (2018). Fast lexically constrained decoding with dynamic beam allocation for neural machine translation. In *Proceedings of the 16th Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1314–1324. Association for Computational Linguistics.

Qu, Y., Ding, Y., Liu, J., Liu, K., Ren, R., Zhao, W. X., Dong, D., Wu, H., and Wang, H. (2021). RocketQA: An optimized training approach to dense passage retrieval for open-domain question answering. In *Proceedings of the 19th Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5835–5847.

Rama, T. and Çöltekin, Ç. (2016). LSTM autoencoders for dialect analysis. In *Proceedings of the Third Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 25–32. The COLING 2016 Organizing Committee.

Rei, R., Stewart, C., Farinha, A. C., and Lavie, A. (2020). COMET: A neural framework for MT evaluation. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 2685–2702.

Reimers, N., Beyer, P., and Gurevych, I. (2016). Task-oriented intrinsic evaluation of semantic textual similarity. In *Proceedings of the 26th International Conference on Computational Linguistics: Technical Papers*, pages 87–96.

Reimers, N. and 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*, pages 3982–3992.

Saito, I., Suzuki, J., Nishida, K., and Sadamitsu, K. (2017). Improving Neural Text Normalization with Data Augmentation at Character- and Morphological Levels. In *Proceedings of the 8th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 257–262. Asian Federation of Natural Language Processing.

Salesky, E., Etter, D., and Post, M. (2021). Robust open-vocabulary translation from visual text representations. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7235–7252. Association for Computational Linguistics.

Scherrer, Y. and Ljubešić, N. (2016). Automatic Normalisation of the Swiss German ArchiMob Corpus Using Character-level Machine Translation. In *Proceedings of the 13th Conference on Natural Language Processing (KONVENS 2016)*, pages 248–255. Bochumer Linguistische Arbeitsberichte.

Sellam, T., Das, D., and Parikh, A. (2020). BLEURT: Learning robust metrics for text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7881–7892.

Sennrich, R., Haddow, B., and Birch, A. (2016). Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725. Association for Computational Linguistics.

Severyn, A., Nicosia, M., and Moschitti, A. (2013). Learning semantic textual similarity with structural representations. In *Proceedings of the 51th Annual Meeting of the Association for Computational Linguistics*, pages 714–718.

Shimanaka, H., Kajiwara, T., and Komachi, M. (2018). RUSE: Regressor using sentence embeddings for automatic machine translation evaluation. In *Proceedings of the Third Workshop on Machine Translation*, pages 751–758.

Singhal, K., Azizi, S., Tu, T., Mahdavi, S. S., Wei, J., Chung, H. W., Scales, N., Tanwani, A., Cole-Lewis, H., Pfohl, S., Payne, P., Seneviratne, M., Gamble, P., Kelly, C., Scharli, N., Chowdhery, A., Mansfield, P., Arcas, B. A. y., Webster, D., Corrado, G. S., Matias, Y., Chou, K., Gottweis, J., Tomasev, N., Liu, Y., Rajkomar, A., Barral, J., Semturs, C., Karthikesalingam, A., and Natarajan, V. (2022). Large language models encode clinical knowledge. *CoRR*.

Søgaard, A., Ebert, S., Bastings, J., and Filippova, K. (2021). We need to talk about random splits. In *Eroceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1823–1832.

Song, K., Zhang, Y., Yu, H., Luo, W., Wang, K., and Zhang, M. (2019). Code-switching for enhancing NMT with pre-specified translation. In *Proceedings of the 17th Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 449–459. Association for Computational Linguistics.

Speer, R., Chin, J., Lin, A., Jewett, S., and Nathan, L. (2018). Luminosoinsight/wordfreq: v2.2.

Susanto, R. H., Chollampatt, S., and Tan, L. (2020). Lexically constrained neural machine translation with Levenshtein transformer. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3536–3543. Association for Computational Linguistics.

Thompson, B., Knowles, R., Zhang, X., Khayrallah, H., Duh, K., and Koehn, P. (2019). HABLex: Human annotated bilingual lexicons for experiments in machine translation. 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)*, pages 1382–1387. Association for Computational Linguistics.

Thoppilan, R., Freitas, D. D., Hall, J., Shazeer, N., Kulshreshtha, A., Cheng, H., Jin, A., Bos, T., Baker, L., Du, Y., Li, Y., Lee, H., Zheng, H. S., Ghafouri, A., Menegali, M., Huang, Y., Krikun, M., Lepikhin, D., Qin, J., Chen, D., Xu, Y., Chen, Z., Roberts, A., Bosma, M., Zhou, Y., Chang, C., Krivokon, I., Rusch, W., Pickett, M., Meier-Hellstern, K. S., Morris, M. R., Doshi, T., Santos, R. D., Duke, T., Soraker, J., Zevenbergen, B., Prabhakaran, V., Diaz, M., Hutchinson, B., Olson, K., Molina, A., Hoffman-John, E., Lee, J., Aroyo, L., Rajakumar, R., Butryna, A., Lamm, M., Kuzmina, V., Fenton, J., Cohen, A., Bernstein, R., Kurzweil, R., Aguera-Arcas, B., Cui, C., Croak, M., Chi, E. H., and Le, Q. (2022). Lamda: Language models for dialog applications. *CoRR*, abs/2201.08239.

Tiedemann, J. and Scherrer, Y. (2017). Neural machine translation with extended context. In *Proceedings of the Third Workshop on Discourse in Machine Translation*, pages 82–92. Association for Computational Linguistics.

Toral, A., Castilho, S., Hu, K., and Way, A. (2018). Attaining the unattainable? reassessing claims of human parity in neural machine translation. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 113–123. Association for Computational Linguistics.

Varis, D. and Bojar, O. (2021). Sequence length is a domain: Length-based overfitting in transformer models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 8246–8257.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, �. u., and Polosukhin, I. (2017). Attention is all you need. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, volume 30, pages 1–11. Curran Associates, Inc.

Vinyals, O., Fortunato, M., and Jaitly, N. (2015). Pointer networks. In Cortes, C., Lawrence, N. D., Lee, D. D., Sugiyama, M., and Garnett, R., editors, *Advances in Neural Information Processing Systems 28*, pages 2692–2700. Curran Associates, Inc.

Wang, B., Kuo, C.-c., and Li, H. (2022). Just rank: Rethinking evaluation with word and sentence similarities. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, pages 6060–6077.

Wang, C., Cho, K., and Gu, J. (2020). Neural machine translation with byte-level subwords. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence*, volume 34, pages 9154–9160.

Wang, K., Reimers, N., and Gurevych, I. (2021). TSDAE: Using transformer-based sequential denoising auto-encoderfor unsupervised sentence embedding learning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (Findings)*, pages 671–688.

Wieting, J., Berg-Kirkpatrick, T., Gimpel, K., and Neubig, G. (2019). Beyond BLEU:training neural machine translation with semantic similarity. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4344–4355.

Wieting, J., Neubig, G., and Berg-Kirkpatrick, T. (2020). A bilingual generative transformer for semantic sentence embedding. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pages 1581–1594.

Yan, Y., Li, R., Wang, S., Zhang, F., Wu, W., and Xu, W. (2021). ConSERT: A contrastive framework for self-supervised sentence representation transfer. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics*, pages 5065–5075.

Yanagida, K. (1980). *“Kagyuko (蝸牛考)”*. Iwanami Shoten, Publishers (岩波書店).

Yasui, G., Tsuruoka, Y., and Nagata, M. (2019). Using semantic similarity as reward for reinforcement learning in sentence generation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 400–406.

Yimam, S. M., Biemann, C., Malmasi, S., Paetzold, G., Specia, L., Štajner, S., Tack, A., and Zampieri, M. (2018). A report on the complex word identification shared task 2018. In *Proceedings of the 13th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 66–78.

Zbib, R., Malchiodi, E., Devlin, J., Stallard, D., Matsoukas, S., Schwartz, R., Makhoul, J., Zaidan, O. F., and Callison-Burch, C. (2012). Machine Translation of Arabic Dialects. In *Proceedings of the Tenth Conference of North American Chapter of the Association for Computational Linguistics*, pages 49–59. Association for Computational Linguistics.

Zhang, T., Kishore, V., Wu, F., Weinberger, K. Q., and Artzi, Y. (2020a). Bertscore: Evaluating text generation with bert. In *Proceedings of the Eighth International Conference on Learning Representations*, pages 1–43.

Zhang, Y., He, R., Liu, Z., Lim, K. H., and Bing, L. (2020b). An unsupervised sentence embedding method by mutual information maximization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pages 1601–1610.

Zoph, B., Yuret, D., May, J., and Knight, K. (2016). Transfer learning for low-resource neural machine translation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1568–1575. Association for Computational Linguistics.

List of Publications

Journal Papers (Refereed)

1. 藤井諒, 三田雅人, 阿部香央莉, 塙一晃, 森下睦, 鈴木潤, 乾健太郎. 機械翻訳モデルの頑健性評価に向けた言語現象毎データセットの構築と分析. 自然言語処理, Volume 28, Number 2, pp 450-478, June 2021.

2. Kaori Abe, Yuichiroh Matsubayashi, Naoaki Okazaki and Kentaro Inui. Multi-dialect Neural Machine Translation for 48 Low-resource Japanese Dialects. 自然言語処理, Volume 27, Number 4, pp.781-800, December 2020.

International Conference Papers (Refereed)

1. Kaori Abe, Sho Yokoi, Tomoyuki Kajiwara, Kentaro Inui. Why sentence similarity benchmark is not predictive of application-oriented task performance? In Proceedings of the Third Workshop on Evaluation & Comparison of NLP Systems, November 2022.

2. Yunmeng Li, Jun Suzuki, Makoto Morishita, Kaori Abe, Ryoko Tokuhisa, Ana Brassard, Kentaro Inui. Why sentence similarity benchmark is not predictive of application-oriented task performance? In Proceedings of the Third Workshop on Evaluation & Comparison of NLP Systems, November 2022.

3. Riki Fujihara, Tatsuki Kuribayashi, Kaori Abe and Kentaro Inui. Topicalization in Language Models A Case Study on Japanese. In Proceedings of the 29th International Conference on Computational Linguistics, October 2021.

4. Ryo Fujii, Masato Mita, Kaori Abe, Kazuaki Hanawa, Makoto Morishita, Jun Suzuki and Kentaro Inui. PheMT: A Phenomenon-wise Dataset for Machine Translation Ro-

bustness on User-Generated Contents. In Proceedings of the 28th International Conference on Computational Linguistics, pp.5929–5943, December 2020.

5. Takuma Kato, Kaori Abe, Hiroki Ouchi, Shumpei Miyawaki, Jun Suzuki and Kentaro Inui. Embeddings of Label Components for Sequence Labeling: A Case Study of Fine-grained Named Entity Recognition. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics Student Research Workshop, pp.222–229, July 2020.

6. Kaori Abe, Yuichiroh Matsubayashi, Naoaki Okazaki, and Kentaro Inui. Multi-dialect Neural Machine Translation and Dialectometry. Proceedings of the 32th Pacific Asia Conference on Language, Information and Computation, pp.1-10, December 2018.

Awards

1. 言語処理学会第 24 回年次大会若手奨励賞 (9/235 件) 受賞, March 2018.
2. 人工知能学会言語・音声理解と対話処理研究会第 84 回研究会対話システムライブコンペティション 3 位入賞, November 2018.