Article

An Optimized Approach to Translate Technical Patents from English to Japanese Using Machine Translation Models

Maimoonah Ahmed [1], Abdelkader Ouda [1,*], Mohamed Abusharkh [2], Sandeep Kohli [3] and Khushwant Rai [3]

1   Department of Electrical and Computer Engineering, Western University, London, ON N6A 5B9, Canada; mahme376@uwo.ca
2   Digital Media Software Engineering, Ferris State University, Grand Rapids, MI 49307, USA; mohamedabusharkh@ferris.edu
3   Village Centre Pl Unit 209, Mississauga, ON L4Z 1V9, Canada; s.kohli@xlscout.com (S.K.); khushwant.rai@xlscout.com (K.R.)
*   Correspondence: aouda@uwo.ca

Abstract: This paper addresses the challenges associated with machine translation of patents from English to Japanese. This translation poses unique difficulties due to their legal nature, distinguishing them from general Japanese-to-English translation. Furthermore, the complexities inherent in the Japanese language add an additional layer of intricacy to the development of effective translation models within this specific domain. Our approach encompasses a range of essential steps, including preprocessing, data preparation, expert feedback acquisition, and linguistic analysis. These steps collectively contribute to the enhancement of machine learning model performance. The experimental results, presented in this study, evaluate three prominent alternatives considered for the final step of the transformer model. Through our methodology, which incorporates a modified version of NLP-Model-III, we achieved outstanding performance for the given problem, attaining an impressive BLEU score of 46.8. Furthermore, significant improvements of up to three points on the BLEU score were observed through hyperparameter fine-tuning. This research also involved the development of a novel dataset consisting of meticulously collected patent document data. The findings of this study provide valuable insights and contribute to the advancement of Japanese patent translation methodologies.

1. Introduction

Machine translation (MT) is an active and rapidly evolving technology in today's software engineering scene even though the idea of using automatic translation is one that predates the invention of computers by a few hundred years. Notable mathematicians and philosophers such as Leibniz and Descartes put forth the idea of using numerical codes as a universal language in the seventeenth century [1]. Though this idea has been around for a long time, the emergence of machine translation, in the modern sense, is said to have taken place in 1949 when Warren Weaver published a memorandum titled "Translation" [2], in which he formulated specific goals and methods that overcame the substantial limitations created by the straightforward method of word-to-word translation. In 2022, the market for MT was estimated at USD 153.8 million and is expected to grow to USD 230.69 million by 2028 with a compound annual growth rate of 7.3% until 2030 [3].

As a sub-field of natural language processing (NLP), MT aims to translate text or speech from a specific source natural language to another target natural language. When thinking about the process of translation, it may seem ostensibly simple: first, a translator, whether human or machine, must decode the meaning of the original text and then

encode that meaning back into the target language. However, when we break down these two steps, the hidden complexities beneath the surface become clearer. To achieve an accurate translation, the translator must have a comprehensive understanding of the components representing languages, such as syntax, semantics, lexicology, etc., of both the source and target languages as well as the culture of the speakers of both languages. Or at least, that is what traditional, also referred to as rule-based translation, aims at.

Additionally, lexical ambiguity further confounds the translation process. Often, words that are either homonyms or polysemous cause lexical ambiguity because they are words that have various meanings, and a translator must identify the expected meaning from context to choose the right word in the target language [1]. Inflectional morphemes are also a source of lexical ambiguity. For example, the word number in the English language may be a noun, or it may be the inflected form of numb [1]. This makes selecting the correct translated option meticulous work. On the other hand, words that are not ambiguous in the source language may be open to more than one interpretation in its target language. This lack of one-to-one correspondence between words in different languages makes translation and inherently automating it a complex task.

A different origin of complexity in translation is syntactic ambiguity where the syntax of a sentence leads to more than one meaning. This type of ambiguity is particularly more complex to a computer because humans are able to pick up on the intended meaning through context, while computers have a difficult time discerning multiple possible meanings [1]. For instance, it is clear to humans that the sentence, "The stolen wallet was found by the fire hydrant" means that the wallet was found next to the fire hydrant; however, a computer might interpret it as the fire hydrant finding the stolen wallet. Evidently, these ambiguities, among many others, make it very difficult for a computer to represent the structure of a language in the form of rules.

Translation becomes a more complex task to do when patent documentation is introduced into the mix, which involves patent conditions, correspondence with lawyers, and a unique style of writing [4]. Deep knowledge of technical terms and a comprehensive understanding of the legal language is required by the translator to achieve high-quality translations, and as powerful documents that are tools for encouraging innovation, patents require excellent language pair expertise. Compromising on the accuracy of machine translation used for patent documents can create a case for fraudulence or can interrupt the patent filing process, which may lead to other consequences such as theft of inventions, expenses, and delays, among many others [5]. The challenges of patent document translation from English to Japanese are detailed in Section 3.

These challenges have called on machine learning specialists to develop novel solution ideas that go beyond the attempt of replicating and automating steps that are used by expert human translators, especially for special-domain translation tasks such as patent translation. The latter is the aim of this paper. The paper will cover the challenges of translating patents from English to Japanese using machine translation, current applications of machine translation, and commonly used evaluation metrics in the field of NLP. These sections lead to the proposed methodology as well as the findings of the experiments conducted. The contributions of this work can be summarized as follows:

- To solve the problem of translating technical multi-domain patents, from English to Japanese, the performance of different MT models with varying parameters are evaluated.
- Start from the best-performing model to develop and implement a novel multi-step approach to fine-tune models to improve accuracy in patent translation.
- The offered model is able to overcome consistent problems for the problem at hand, including lexical ambiguity and sentence structure challenges, as indicated by superior BLEU score performance compared to previous solutions to this problem.

2. Machine Translation Approaches

MT solutions can be classified into three main approaches: rule-based machine translation (RBMT), statistical machine translation (SMT), and neural machine translation (NMT).

2.1. Rule-Based Machine Translation

RBMT is the earliest method of machine translation. It is a system that relies on linguistic information about the target and source languages that are retrieved from traditional language sources such as dictionaries, grammar rules, etc. This information includes lexical, syntactic, semantic, and morphological congruities of both languages. RBMT then performs a morphological analysis on the grammatical structure of both the source and target languages using a parser to construct an output [6]. Due to the number of language rules and variations of forms for linguistic structures, building a complete rule-based system is a prohibitively long and rigid process.

Moreover, RBMT requires manual placement of some linguistic information, and significant post-amendments made by human experts are needed to obtain a sufficient output. As discussed earlier, many types of ambiguities also make it infeasible when working with large systems, and improvements depend on more hard-coded rules and human involvement [6]. With the emergence of SMT and NMT, RBMT has become the least common method of translating text, though there are still many online translation systems existing that use a rule-based approach. Some depend on less complex and ambiguous sentence structures such as inputs to perform well and thus are not reliable for more general use cases.

2.2. Statistical Machine Translation

Until recently, SMT had been predominantly used in MT research and industry MT systems, such as Google Translate, given its more precise translation results compared to RBMT [7]. Because meaning inherently stems from the way words are used together, using a statistical approach that can detect regularities in text and context of use increases the quality of translation significantly [7]. The SMT approach analyzes very large amounts of linguistic data taken from previous translations and assesses their statistical probability of being the most appropriate translation. The data used are typically in the form of a corpus containing source and target sentences in pairs [1], referred to as bilingual corpora. The analysis of the statistical probability of the data first occurs in the translation model, in which the probability of the correspondence between sentences in the two languages is estimated. Adjacent to the translation model, the system learns to calculate the probability of word sequences being valid in what is called the target-language model [1]. The model must also consider the extent to which each language differs in syntax, so a distortion model is learned [1]. This model takes into account the probability that a word placed in a specific position within the source sentence will move to a different position in the target sentence [1]. For SMT to be able to translate text, a crucial third component must be used—the decoder. A decoder takes the source sentence and determines the different probabilities of each word within the translation model, which then outputs possible words and phrases. The output is used as an input within the target-language model, which then calculates and outputs the most likely translation according to complex mathematical and statistical methods [1]. SMT enhances the RBMT method, but its performance is hindered when we consider the fact that words between languages do not have a one-to-one correspondence.

A crucial part of statistically based MT systems is the phrase transition model [8], which depends on either the conditional probability of generating a target sentence given the source sentence, or the joint probability of the concurrence of the source and target sentences [8]. The nature of such models makes them susceptible to viewing phrases that are paraphrased or different on the surface as distinct units. This prohibits the distinct units, though they may share linguistic properties, from sharing parameters of the model while translating, which leads to the problem of sparsity that affects morphologically rich languages [8], including the Semitic languages Hebrew and Arabic [9]. Because of these shortcomings in SMT, the need for novel algorithms that utilize machine learning became more pressing. NMT provides an alternative that aims to tackle these challenges, and hence it has risen in popularity recently.

2.3. Neural Machine Translation

NMT, put simply, uses a single neural network to directly translate a source sentence to its intended target sentence [10]. Both SMT and NMT rely on a large corpora of pairs of sentences in the source language and their corresponding translated target sentences, but NMT implements continuous vector representations of linguistic units, unlike SMT, which implements discrete symbolic representations [11], i.e., propositions represented as discrete objects of varying sizes [12]. Collobert et al. used continuous representations for words and were successful in showing results that overcame the problem of scarcity in which the morphological, semantic, and syntactic properties of a language were represented [13].

Although there are other language models in the literature such as recurrent neural networks (RNNs) and connectionist temporal classification (CTC) [14], this work focuses on discussing the encoder–decoder architecture in this section as it is the base model for most NLP sequence-to-sequence use-case models, especially in NMT [15]. It is also used in NMT systems, which will be discussed in Section 3. The encoder–decoder architecture is a standard method used to learn the conditional distribution of a sequence given another sequence. The encoder and decoder are trained together in order to maximize the conditional log-likelihood [13], and the encoder–decoder utilizes a series of recurrent neural network cells.

A successful example can be seen in [16], where a transformer model that uses the attention mechanism to determine which parts of the input source text are of the most importance, to extract mutual dependencies between the input and the output, was implemented. The attention mechanism refers to a function that maps a query and a set of key–value pairs, represented as vectors, to an output, which is also represented as a vector [16]. To compute the output, first, a compatibility function of the query, along with the designated key, calculates a weight that is assigned to each of the values. The weighted sum of the values is then presented as the output [16].

The architecture of the model is made up of an encoder and a decoder. The encoder is composed of six identical layers, where each layer has two sub-layers: a multi-head self-attention mechanism and a fully connected feed-forward network. A residual connection is then applied around each of the two sub-layers, which is followed by layer normalization. Similarly, the decoder is made up of six identical layers with a third sub-layer inserted that takes the output of the encoder and performs multi-head attention on it [16].

3. English-to-Japanese Patent Machine Translation

The translation problem we are tackling focuses on translation from English to Japanese, specifically on translating technical patents in different domains from English to Japanese. The purpose of this is to understand the complexities of language translation for relevant use cases such as patent translation. The vast dissimilarity (lexically, syntactically, morphologically, and semantically) between the two languages will aid in highlighting some of these complex problems that are encountered in MT with the purpose of studying to mitigate them. One source of distinctions other than the obvious difference in characters of both languages is the word order in English and Japanese. In English, the word order is in the form of subject–verb–object (SVO), while the word order in Japanese is SOV [17]. This means that topics are at times expressed in entirely different sentence structures, and thus there is a lack of correspondence between the structures of sentences, which poses a challenge for the machine to translate accurately [18]. Additionally, being a high-context language, it is common in Japanese for much of the information to be implied [19]. For example, the subject is usually dropped in situations where the context is clear [20]. If a person were to say "I am going to bake a cake" in English, the direct Japanese translation would be "bake a cake" since it is understood by the listener that the speaker is the subject, and thus "I" is implied. In contrast, English, as a low-context language, communicates content explicitly, and writing is understood very literally. This demonstrates that there is more to translation than ensuring the machine translator chooses the correct words; the cultural context must also be taken into account to make sure that the intended meaning is

delivered correctly [21]. This could prove an unfeasible task for computers; being a cultural mediator requires a certain level of refined reasoning where one must be able to deduce what meanings might be extracted by a reader so that the translation may be adjusted as needed [21]. Without the explicit specification of a large number of language and cultural context-based rules, this task is difficult for machines, and if such intricate rules, which may be in the tens of thousands, were to be constructed and documented with the system, the effort expended may be prohibitive and contradictory to the purpose of utilizing machine learning to automate the MT task altogether. A system that overcomes the need to explicitly set language-dependent or culture-dependent rules is the ultimate goal here. MT using machine learning is a candidate to solve this problem.

The problem of context sensitivity is more or less avoided when considering the source and target text that is shared by readers with the same background knowledge [21]; for example, readers may be part of the same scientific discipline or industry. The premise of this work will focus on the translation of technical patents from English to Japanese using machine translation. Although the complexity of context sensitivity is reduced, we are faced with another challenge: patents of scientific nature contain many technical words that are domain-dependent and may be homonyms with more than one interpretation depending on the subject. Instead of running into the problem of deciphering cultural context, the machine must determine the correct translation of a word depending on the domain it is used in. For example, the word "arm" is a homonym for both the biological human arm as well as a technological robotic arm in English. In Japanese, it becomes sensitive to the domain it is used in; "腕 " refers to a human arm, while "アーム" refers to a mechanical arm.

The world has grown increasingly connected and more technologically advanced, and this is reflected by the increase in patent applications in new and different technology spaces over the years. In 2020, the reported number of patents filed worldwide increased by 1.6% with 3.3 million patents filed, where approximately 85% of the total number of patents filed were accounted for by five national/international patent offices [22]. The National Intellectual Property Administration of the People's Republic of China (CNIPA) received upwards of 1.5 million applications, followed by the United States Patent and Trademark Office (USPTO), which received 597,172 applications. Ranked third, the Japan Patent Office (JPO) had 288,472 applications, the Korean Intellectual Property Office (KIPO) had 226,759, and finally, the European Patent Office (EPO) had 180,346 [22]. Figure 1 below demonstrates the growth of patent applications from 2006 to 2020 worldwide. MT of patents becomes an important problem as it is useful, in terms of industrial use, for countries to be able to file patents in foreign languages. The translation must also be accurate as possible since even a small variation of the intended meaning may lead to legal loopholes that will be taken advantage of to exploit intellectual properties [23]. Efficient MT of patents is a direct step towards having unified human knowledge and sharing patent information for the human race in the general domain or at least on certain predefined domains, such as efforts related to fighting climate change.

Although commercially available translators such as Google Translate and Microsoft Translator offer machine translation with a high level of accuracy for generic and non-technical texts, more specialized domains/scopes rely on industry-specific training data to translate the text so that the translation may be relevant in their context [24]. While Google's Cloud Translation and Microsoft's Translator allow for custom translation and model training to counter this issue, the cost of using them risks getting expensive since more complex, and large training data as well as custom translations lead to high costs while using those products, which may be prohibitive. Custom translations with Microsoft are performed using their C2–C4 instances, where the cost of using those instances ranges from 2055 USD/month to 45,000 USD/month, respectively [25]. This incurs a large expense, especially for companies with many patent grants. For instance, IBM, ranked as 2021's most innovative company in the US IP space, had 9130 and 8682 patent grants in 2020 and 2021, respectively [26]. Since IBM offers a range of products and patents corresponding to

different industries, the use cases would require multiple model training, which further increases the cost. Given the difficulty of MT and the high cost associated with it, continued research in the field of MT is extremely important, and therefore, the aim of this paper is to investigate and analyze existing machine translation models used to translate patents containing highly technical words from a range of domains while also working to increase the models' level of accuracy using training data.
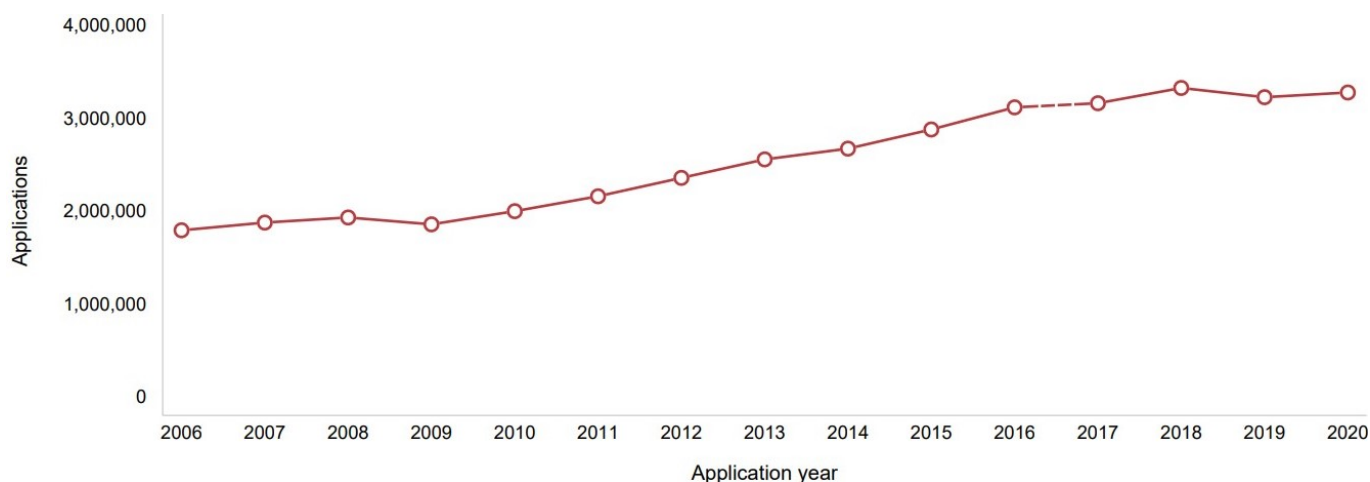

Figure 1. Number of Patent Applications Submitted Each Year From 2006 to 2020 [22].

4. Evaluation Methods

The evaluation of MT models is extremely important as it measures the degree of reliability of the output from the MT model, and it also informs us when a model requires improvement [27]. Over the years, many evaluation methods have emerged, many of which fall into two categories: human evaluation and automatic evaluation. Automatic evaluation metrics work by comparing the output of an MT system to a set of references generated by humans, also called the gold-standard references [28], and then making use of mathematical and statistical calculations to compute how different the machine-translated output is to the reference translation [28]. The quality of the translation is considered better if the difference between the output and reference is smaller. Automatic metrics use n-grams to calculate the precision scores, where an n-gram is a sequence of n words [28]. This section goes over some commonly used automatic metrics that are used to evaluate MT.

4.1. BLEU

The BLEU score, also known as the Bilingual Evaluation Understudy score, is a metric used to assess machine-translated text and evaluate how accurate it is compared to a set of references. More specifically, it is the product of n-gram precision and a brevity penalty (BP). The brevity penalty refers to the penalty applied to the BLEU score when the translated text is much shorter than the reference text, and it restitutes for the BLEU score not having a recall term [29]. The n-gram precision is calculated by counting the total number of word sequences from the MT system output that are also in the set of references [30]. An n-gram, put simply, is the set of n consecutive words within a given sentence [31]. For example, considering the sentence "the wall is white", a 1-gram or unigram is a set that consists of "The", "wall", "is", and "white", and a 2-gram or bigram consists of "The wall", "wall is", and "is white". It is important to note that the words within an n-gram must be taken in consecutive order [31]. The product between the geometric mean of the n-gram precision scores and the BP returns a BLEU score that falls in a range of 0 and 1 (or 0–100), where 0 indicates no overlap between the machine-translated text and the reference text [30]. A score of 1 indicates that the machine-translated text perfectly matches the reference text. Since even linguistic consultants or human translators do not achieve a perfect translation, a BLEU score of 1.0 is almost impossible. However, as a rough guideline, a score between

60 and 70 is generally the best a model can achieve. The n-gram precision of BLEU depends on exact word matches between the output and references. However, since a specific reference may not be the only correctly translated option, a good translation may be scored lower [32]. Despite the transparency of the flaws associated with using the BLEU score, it continues to be widely used in MT research mainly due to its high correlation with human judgment of accuracy [32].

4.2. NIST

Another metric commonly used in MT evaluation is the National Institute of Standards and Technology or NIST. A variant of BLEU, NIST assigns a higher weight to more informative n-grams and uses the arithmetic mean instead of the geometric mean used by BLEU [30]. The calculation of the BP is also where NIST and BLEU diverge; the variation in length between the translated text and the reference text does not affect NIST as it does with the BLEU score [33]. This is because the precision scores that are calculated in BLEU are replaced with the information gained from each n-gram [34]. This enables the system to obtain more credit if the n-gram match is difficult to obtain or less credit if the match is easier [34].

4.3. WER

The Word Error Rate (WER) is one of the earlier metrics used for evaluating MT [30], and it examines the accuracy based on the Levenshtein distance. The Levenshtein distance between two words from the translated output and the set of reference text, refers to the minimum number of edits that are required to change a word from the translated output to the word from the reference text [30]. The edits allowed are substitutions (S), insertions (I), and deletions (D). Equation (1) is used to calculate WER, where N is the total number of words in the reference text:

$$WER = (S + I + D)/N \tag{1}$$

4.4. METEOR

As mentioned in Section 1, the precision-oriented nature of BLEU is the source of a few weaknesses, and so the Metric for Evaluation of Translation with Explicit Ordering (METEOR), a recall-oriented metric, is used to tackle these shortcomings [30]. METEOR calculates the harmonic mean, as opposed to the geometric mean, by combining precision and recall with a greater bias towards recall [30]. The computation of the final METEOR score requires multiple stages; the first stage is exact matching where sentences in the translated output and reference text that are completely alike are aligned [30]. The next stage called stem matching refers to the process of aligning words that have the same morphological stem [30]. Finally, in the synonym-matching stage, words that are synonyms of each other (according to WordNet, a lexical database of the English language [35]) are aligned [30]. At each stage, only words that are not aligned are allowed to be matched in the succeeding stage. Furthermore, a fragmentation penalty (FP) is applied to account for the differences in word order [30]. The METEOR score is then calculated by taking the product of the harmonic mean and $(1 - FP)$, which outputs a score in the range of 0–1.

This paper uses the BLEU score metric to compare the performance of various MT models since BLEU is widely used in the literature and this aids in the comparison of results. Table 1 below provides a summary of the advantages and disadvantages of the metrics covered above.

Table 1. Advantages and Disadvantages of MT Evaluation Metrics.

Evaluation Metric	Advantages	Disadvantages
BLEU	• High correlation with human judgment of translation accuracy [32].	• Does not account for synonyms [32]. • Penalizes longer sentences [32].
NIST	• No penalty applied when translation text is shorter in length [33].	• Does not consider synonyms, syntactic structure, or word order [33].
WER	• Not difficult to understand or implement [33]. • Reproducible [33].	• Does not consider semantic similarity, syntactic structure, or word order of sentences [33].
METEOR	• Incorporates some linguistic knowledge using stem and synonym matching [33]. • Does not penalize longer sentences unlike BLEU [33].	• WordNet does not support most languages well except for English [33]. • Relies on WordNet so not suitable for morphologically rich languages [33].

5. Related Works

This section will cover existing work in the field of MT using RBMT, SMT, and mainly NMT models. In [36], the authors propose to extend the use of a rule-based technique to simplify sentences before using an RBMT system to translate those sentences from English to Tamil. Complex sentences are made simpler using connectives such as relative pronouns, coordinating conjunctions, and subordinating conjunctions [36]. Table 2 below shows the words that were used by the system as connectives.

Table 2. Connectives used to Simplify Sentences.

Relative Pronouns	Who, which, whose, whom
Coordinating conjunction	For, and, not, but, or, yet, so
Subordinating conjunction	After, although, because, before, if, since, that, though, unless, where, wherever, when, whenever, whereas, while, why

Additionally, delimiters such as '.' and '?' are used to divide long and complex sentences into sub-sentences where the meaning of the sentence remains the same [36]. The authors chose to ignore the ',' delimiter. The authors lay out the framework as follows: first, the initial splitting of the sentences from paragraphs is performed using the delimiters. Each sentence obtained after the initial splitting is then parsed using the Stanford parser [36]. The next round of splitting is completed using the coordinating and subordinating conjunctions in each sentence. Then, the sentence is further simplified if it contains a relative pronoun [36]. To compare the system's accuracy, 200 sentences were first given to the RBMT system to translate from English to Tamil. Due to syntax and reordering, 70% of the translated sentences were incorrect. Then, the same 200 sentences were simplified using the outlined framework and given again to the RBMT system. After simplification, 57.5% of the sentences were translated correctly. The authors concluded that longer sentences that are given to the MT result in a low translation accuracy while simplifying the sentences increase the accuracy significantly when translating from English to Tamil. Although an accuracy of 100% is not possible to achieve in MT, the authors prove that the splitting and simplification technique can notably improve MT systems.

In another paper titled "Rule Based Machine Translation Combined with Statistical Post Editor for Japanese to English Patent Translation", the authors also tackle the problem

of decreased accuracy due to long sentences used in machine translation. Their hypothesized solution is to use a statistical post-editor in conjunction with the RBMT system to improve accuracy. The data used by the authors were collected from the Patent Abstract of Japan, or PAJ, and the abstract of the Patent Publication Gazette (PPG) of Japan, in which the former was used as the Japanese corpus and the latter, were used as the English corpus [37]. The sentences were manually translated. Using the corpi, the training and test datasets were created using the following steps: first, the number of words in sentences was counted, and if the number exceeded 90, the sentence was rejected. Next, if the ratio of the number of words in sentences from both the reference text and the translated source text did not fall between 0.5 and 2, inclusively, the sentence would also be rejected [37]. To evaluate the RBMT system combined with the statistical post-editor (SPE), the authors propose a new metric of evaluation: an n-gram-based NMG measure. NMG, or normalized mean grams, which counts the total number of words within the "longest word sequence matches between the test sentence and the target language reference corpus" [37]. Their results concluded that for patent translation where sentences are long and complex, the RBMT provided an advantage for structural transfer [37]. Additionally, since patents are made up of very technical terms, the SPE provided improved lexical transfer [37].

Before NMT systems became popularized, SMT models were widely used for translation purposes. SMT systems use a phrase-based approach, which allows them to reduce the restrictions of RBMT's word-based approach by translating sequences of words at a time. NMT systems further improve on this because of their ability to learn and improve independently; however, they are still far from perfect. In [38], the authors isolate one of the shortcomings of NMT and propose a novel approach to improve NMT by using SMT. NMT systems have a tendency to forgo accuracy for fluent translation and to improve on this, the authors introduce a hybrid approach using SMT [38]. To attain the goal of improvement, the authors implement the following steps. After the SMT and NMT models have been trained on parallel corpora, the SMT system receives the source text first as input and the translated output of the SMT is encoded. The authors then modify the NMT beam search algorithm, which gives a chance to related SMT tokens during NMT decoding [38]. After the modified beam search has been conducted, the translation sentence from the modified beam algorithm goes into the decoder, which then outputs the translated target text [38]. The intent of this improved algorithm is to increase the probability of SMT tokens that are in lower positions within the beam to be chosen in the NMT decoding step [38]. The algorithm is used in three different approaches in the paper. First, it is applied for a certain number of tokens from the start of the decoding, e.g., the first, second, or third tokens. Second, the algorithm is applied for each sentence as long as the SMT token is discovered in the beam at some part in the decoding. Lastly, it is applied until the decoding ends [38]. The authors use automatic evaluation metrics, specifically BLEU and METEOR, to draw conclusions from the experiment. They conclude that the first and second approaches provide good results in the translation quality, while the third approach, however, performs poorly. Nevertheless, they are able to successfully show that using the phrase-based SMT system can provide improvements in NMT decoding, which ultimately leads to a higher quality of translations [38].

As mentioned previously, NMT models have become a dominant approach to machine translation problems and have improved on several shortcomings of SMT over the years. Though a promising technology, NMT still faces many hurdles as its accuracy in translating several language pairs significantly depends on the availability and use of large parallel corpora [39]. For a large group of languages, however, obtaining a large parallel corpus proves to be difficult. For example, language isolates such as Basque or macaronic languages such as German–Russian [39] do not have enough data available to train an MT model efficiently. Research has been conducted to overcome this problem, and techniques such as triangulation techniques and semi-supervised methods have been proposed; however, there is still a need for strong cross-lingual learning [39]. To overcome the need of using

cross-lingual learning, the authors of [39] propose a novel solution to train NMT models. They suggest relying on a monolingual corpus for entirely unsupervised model training.

The authors used a standard model architecture: an encoder–decoder system with attention mechanisms where the encoder and decoder contained a two-layer bi-directional recurrent neural network, and the attention mechanism used was a global attention method with the general alignment function [39]. Three critical aspects allowed the MT system to be translated with an unsupervised approach, and this included a dual structure, a shared encoder, and fixed embeddings within the encoder. Furthermore, two strategies allowed the NMT system to predict translations in a monolingual corpora, which would have been otherwise impracticable since the authors did not use a parallel corpus. First, they use the principle of denoising autoencoders that train the system to reconstruct a corrupted input to its original form. More specifically, they switch the word order of the input sentences so that the system can learn to retrieve the correct order [39]. Next, they use an adjusted on-the-fly back-translation method so that given a particular input sentence in the source language, the system can use inference mode with greedy decoding to translate the input to the target language [39].

After conducting both automatic and human evaluations, the experiments conclude that there is a significant improvement in translation over a baseline system that performed word-by-word substitution. The system was also able to effectively model cross-lingual relations and output accurate and excellent quality [39]. They further showed that moving on from a strictly unsupervised case by incorporating a small parallel corpus has the potential to further improve the quality of translations [39].

Another translator that successfully utilizes neural network architecture for translation is DeepL. DeepL is a neural machine translation service that advertises its enhanced performance compared to competitor tech companies such as Google Translate and Microsoft Translator. It separates itself from the competitors by improving the neural network methodology in four different areas: the network architecture, training data, training methodology, and the size of the network [9].

6. Methodology

The aim of this research is to improve the accuracy of translating technical patents, belonging to different domains, from English to Japanese. Since MT systems such as Google Translate use generic data for translation, translating highly technical and domain-specific sentences often faces many problems. Some of these problems include lexical ambiguity and sentence structure, as discussed earlier in Section 2.

We evaluated different open-source MT models with varying parameters to study and analyze the performances of each model. Analyzing existing models allowed us to gain integral insights into the task of translating complex language structures. Once an initial analysis of models' performance was completed, we implemented a multi-step approach to fine-tune three models to improve accuracy in patent translation. Figure 2 provides a depiction of the activity diagram of our methodology behind fine-tuning a machine translation model.
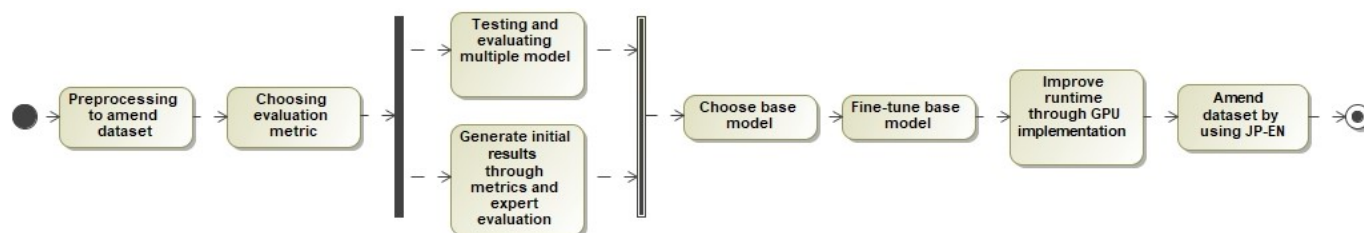

Figure 2. Activity Diagram of Methodology.

6.1. Computing Setup

All training and testing of the MT models covered in the subsequent subsections were completed using 4 NVIDIA Tesla P100-SXM2 GPUs with 17 GB RAM as well as 24 CPUs/cores.

6.2. Approach I: Transformer Model with Attention

Bharadwaj et al. [40] built an NMT system to translate text from Japanese to English. First introduced in [16], Bharadwaj et al. implemented a transformer model that uses the attention mechanism to determine which parts of the input source text are of the most importance, and to extract mutual dependencies between the input and the output. The dataset used to train the model consists of two merged datasets; the first is a corpus that contains approximately 500,000 pairs of sentences that cover the topics of Japanese religion, culture, and history [41], and the second is a collection of bilingual sentence pairs created by [42] that comprises Japanese sentences used in daily conversations. The model was trained using 68,674 rows of the dataset and then evaluated using the BLEU score. The authors were able to achieve a BLEU score of 41.49 [40], suggesting a good accuracy of the translated text.

To use this model for the purpose of patent translation, first, the model was adjusted so that the source text was English and the target text was Japanese. The same dataset used in [40] was then used to train the new model. Once trained, we tested the translation accuracy on sentences extracted from technical patents and then evaluated the results using both automatic (BLEU) and human expert evaluations. Bharadwaj et al.'s translation model will be referred to as NLP-Model-I in this paper from this time forward.

6.3. Approach II: Pre-Trained Hugging Face Models

Generally, the larger a dataset is, the better an NMT system can perform. However, training time is also drastically increased, often lasting for days even when using modern GPUs, such as a TitanX, and distributed training in TensorFlow [43]. To mitigate computational costs and save time, we adapted two pre-trained models from HuggingFace, an AI model hosting platform. The pre-trained models are made up of a standard transformer architecture consisting of an encoder and a decoder as well as an attention mechanism. This architecture is the same as in the model used in Approach I. The first pre-trained model used was the Helsinki-NLP [44] model. The exact name of the second pre-trained model, for commercial confidentiality reasons, will not be included and will be referred to as NLP-Model-III. For consistency, the Helsinki-NLP Model will be referred to as NLP-Model-II. Both models use the BLEU score for evaluation and have a baseline score of 15.2 and 32.7, respectively.

NLP-Model-II was developed for the Tatoeba translation challenge, which aims to serve as a catalyst for the development of open translation models [45]. The dataset used is an amalgamation of the Open Parallel Corpus (OPUS) [46], an open collection of parallel corpora [47], and test data extracted from [48]. Similarly, the dataset used to train NLP-Model-III was built using various datasets that consisted of a total of approximately 6.6 million bilingual pairs. Of these many datasets, the following were used: the Japanese-English Subtitle Corpus [49], the Kyoto Free Translation Task (KFTT) [50], the Tanaka Corpus [51], the Japanese SNLI dataset [52], and finally, WikiMatrix [53].

Each model's hyperparameters remained unchanged from the original baseline model's hyperparameters and Table 3 provides an overview of each of the model's hyperparameters. From a parameter's perspective, NLP-Model-II and NLP-Model-III are similar with some adjustments. For instance, the number of beams chosen for Model-II and Model-III are 6 and 12, respectively. This parameter represents the number of sequences at each time step that is used for the decoding process of the beam search algorithm. Increasing this value may be advantageous as this allows the model to explore the search space in a more diverse way in turn increasing the probability of finding a solution of higher quality.

Vocabulary size is a parameter used to represent the number of tokens in the dataset that are unique, and it influences the model's performance and translation quality. To choose an effective value for this parameter, it is important to find the right balance between computational efficiency and adequate coverage.

The boolean gradient check-pointing parameter may be used to save memory; however, if set to *True*, the parameter causes the backward pass to be slower and less effective. In newer versions, this parameter has been deprecated and thus is set to a default of *False*.

Table 3. Overview of Each Model.

Model	Parameters	Dataset	BLEU
NLP-Model-I	• Optimizer: Adam • Warmup steps: 4000 • Dropout: 0.1 • Label smoothing: 0.1	Japanese–English Bilingual Corpus [54] and Japanese–English sentence pairs [42]	<0.001
NLP-Model-II	• Weight decay: 0.01 • Batch size: 16 • Metric: sacrebleu • Vocabulary size: 32,001 • Num of beams: 12 • Gradient check-pointing: False	tatoeba mt [45]	<10
NLP-Model-III	• Weight decay: 0.01 • Batch size: 16 • Metric: sacrebleu • Learning Rate: 0.00002 • Vocabulary size: 65,001 • Num of beams: 6 • Gradient check-pointing: False *Deprecated parameter*	tailored patent dataset (Not publicly available)	46.8

6.4. Dataset Used for Fine-Tuning

Using both of the baseline models to translate technical patents from our database was not at a satisfactory level as many technical words and sometimes sentences would be incorrectly interpreted by the models. For example, the output would often contain incomplete sentences where words were cut off from the end of a sentence or incorrect grammar rules were applied. To improve the results outputted by the models, we fine-tuned the models by creating a dataset using approximately 80,000 patents that spanned various domains and used that dataset to train each model. The domains of the patents included human necessities; performing operations and transporting; chemistry and metallurgy; textiles and paper; fixed constructions; mechanical engineering, lighting, heating, weapons, and blasting; physics; and electricity. This was to ensure that each of the models can learn to accurately translate words according to their context. Additionally, from the Japanese Patent Office [55], we used data consisting of a Japanese–English machine translation dictionary to further enlarge our dataset. The dictionary was originally in unified power format and thus had to be extracted using the Accelerator library.

Originally, the created dataset contained several spelling and grammatical errors and thus reduced the accuracy in the performance of the original model. To rectify this, an additional preprocessing step was performed. We used back translation, which is a technique that translates the target language to the source language and combines the original source sentences and back-translated sentences in an effort to train the model. The model was trained on the training dataset, which consisted of 124,828 rows of English and Japanese sentences, and training took approximately a total of three and a half hours. Then,

the model was tested using the validation dataset consisting of 31,207 rows of English and Japanese sentences. Moreover, the batch size chosen for fine-tuning the model was 16, and the metric used to measure accuracy was sacrebleu. The weight decay parameter was set to 0.01, and the number of train epochs was set to 3.

After implementing various models and completing preprocessing steps, the following step included implementing multi-GPU processing to improve model runtime. Multiple GPU implementation options were considered including Genesis Cloud, DIGITS2, NVIDIA, and Asynchronous Functions. Evaluation of those options and the chosen design is reported in the Results section (Section 7).

To fine-tune the model further, we have sent an application to ASPEC, a Japanese-based company that creates JP-EN data sets that include millions of parallel sentences extracted from scientific journals. This dataset is very promising as the main problem we faced was translating technical words that are hard for even a layman to grasp.

7. Results

7.1. Result I: NLP-Model-I

The results of NLP-Model-I were not what we were expecting considering the BLEU score achieved by Bharadwaj et al. [40]; the model did not perform as accurately as we had hoped. Many of the technical scientific words were not able to be translated correctly, and many sentences had severe grammatical errors. The results were dismissible as the BLEU score result was much smaller than 0%, meaning that the translation was not accurate at all. Moreover, fine-tuning this model would provide subpar results, and so we chose to explore different options in lieu of fine-tuning.

7.2. Result II: NLP-Model-II

Using test data consisting of technical terminology/jargon from the aforementioned domains, we tested both NLP-Model-II and NLP-Model-III. The results were very promising for NLP-Model-II when compared to NLP-Model-I since the BLEU score was much higher than it was for NLP-Model-I. However, it still could not perform acceptably when translating text from scientific patents. The BLEU score for small sentences was around 40–50%, and many sentences were cut off mid-sentence after the translation. We decided to use post-editing to confirm the results. Post-editing is the procedure of human intervention to amend machine-translated text so that translation is acceptable and is used to polish a final product. In our case, it is to check and evaluate whether a specific model is performing below par or if the model is accurate in its translation of the scientific text.

It was especially important as some words such as a 'technological arm' were being translated into the Japanese word for a human arm, which changed the meaning and context of the paragraph that was being translated. This example is a clear demonstration of the limitations of machine translation and the importance of post-editing. It was confirmed with post-editing that the accuracy was much lower than expected. Using post-editing, the accuracy was only 10%. This demonstrates one of the disadvantages when it comes to using the BLEU score as an evaluation metric.

7.3. Result III: NLP-Model-III

After testing and evaluating NLP-Model-III, we determined that our proposed technique for machine translation of Japanese patents, which encompasses NLP-Model-III, performed the best for this particular problem. Post-fine-tuned results also confirmed the improvement. The average BLEU score of the three epochs was determined to be 46.18. This is a 41.22% increase in model performance from the original model BLEU score of 32.7. Table 4 provides sample results of each model and depicts their level of accuracy in translation.

Table 4. Sample English Sentences and Their Corresponding Japanese Sentences Translated using models.

Model	English (Original)	Japanese (After Translation)
NLP-Model-I	The car scheduling server receives a ride request sent by the client.	The no break Izu NAGATA suffering? Six a ? military shop She Please—not because The have n't OK I I I am—such good I I I are I I I good I—Bus—Bus restaurant I I—while home sha—while home Tokyo—while home studying delicious the I I—not He could not He Monday delicious not He could wearing " mother and not He could
NLP-Model-II	The present invention provides a carbon dioxide absorbent during combustion of fossil fuels comprising a pressed dry powder of plant fibers, and a fossil fuel characterized by comprising of such a pressed dry powder of plant fibers.	The present invention is characterized by absorbing carbon dioxide during the combustion of fossil fuels comprising a pressed dry powder of plant fibers , drying the plant fibers in such a way as to produce fossil fuels . We supply.
NLP-Model-III	The present invention provides a carbon dioxide absorbent during combustion of fossil fuels comprising a pressed dry powder of plant fibers, and a fossil fuel characterized by comprising such a pressed dry powder of plant fibers.	The present invention provides a carbon dioxide absorbent at the time of combustion of fossil fuels containing a compressed dry powder of plant fibers, and such a compressed dry powder of plant fibers.

As seen in Table 4, NLP-Model-I outputs an incomprehensible translation that is completely unrelated to the input. The output of NLP-Model-II was a lot more accurate; however, it was a recurring problem to see sentences cut prematurely or irrelevant words added to the sentences. The importance of accurate translation, especially when working with legal documents, was one of the main reasons we moved on from NLP-Model-II and investigated other models. NLP-Model-III, as seen above, provided the closest and most accurate translation without compromising on technical words and preserving their meaning according to their context. To strive to achieve further improvement, we tested three different values for three different hyperparameters of the model using a manual search based on our judgment. Although there was some improvement from the baseline model, ultimately, the best-performing model was the one fine-tuned using our data set and the default parameters. The hyperparameters tuned were the learning rate, batch size, and weight decay. Table 5 provides an overview of the hyperparameters and their resulting BLEU score.

Table 5. Fine-Tuning Hyperparameters of NLP-Model-III.

Hyperparameters	Experiment 1	Experiment 2	Experiment 3
Learning rate	0.1	0.01	0.0001
Weight decay	0.1	0.001	0.0001
Batch size	8	16	32
BLEU	33.32	31.22	34.74

Given these outcomes, it is clear that NLP-Model-III is the best model to focus on as it translated scientific patent abstracts the most accurately and showed promising results. One problem that remained was the performance speed of the model. For large paragraphs, it would take the model approximately 19 s to deliver an output of translated text, and

sentences took approximately 3 s to translate. The aim for us was to achieve a translation in milliseconds so that the translated content would be able to load faster on a webpage.

8. Conclusions

In this paper, we proposed a comprehensive multi-step technique to solve the problem of Japanese patent translation. The unique challenges of patent translation stem from the legal nature of the patent document in comparison to general Japanese-to-English translation and from existing challenges of the Japanese language, which raise the complexity of the models that could be successful. Our technique included prepossessing steps, data preparation, enlisting human expert feedback, and linguistic analysis to refine the machine learning model performance. The Results section (Section 7) includes evaluation results of three major alternatives for the transformer model to depend on for the last step. The aim was to achieve an output from the models that would fall in the range of 0.5–0.7 of the BLEU score, which is the current state of the art. Our technique, which encompassed a variation of NLP-Model-III, achieved the best performance for the problem at hand, reaching a BLEU score of 46.8. It is also noticed that fine-tuning hyperparameters yield up to a three-point improvement on the BLEU score. This work included developing a novel dataset that consisted of data collected from patent documents.

Moreover, the interest in commercial use motivates the further study of MT as the initial findings of this work were demoed in Japan at the PIFC commercial conference where companies tested the performance of the model and were satisfied with the level of translation provided. Along with the quality of translation, the speed of translation is also of great importance. Moving forward, it is important to study how to implement data parallelism to use multiple GPUs at once to increase the speed at which translation occurs.

References

1. Somers, H. Machine Translation: History, Development, and Limitations. In *The Oxford Handbook of Translation Studies*; Malmkjær, K., Windle, K., Eds.; Oxford Academic: Oxford, UK, 2012. [CrossRef]
2. Weaver, W. Translation. In Proceedings of the Conference on Mechanical Translation, 1952. Available online: https://aclanthology.org/volumes/1952.earlymt-1/ (accessed on 10 January 2023).
3. Newsmantraa. Machine Translation Market to Observe Exponential Growth by 2022 to 2030: Microsoft Corporation, IBM. *Digital Journal*, 14 June 2022. Available online: https://www.digitaljournal.com/pr/machine-translation-market-to-observe-exponential-growth-by-2022-to-2030-microsoft-corporation-ibm (accessed on 10 January 2023).
4. Bianchi, C. Everything You Should Know about Patent Translation, the Professional Translation Blog. *Language Buró*, 20 October 2020. Available online: https://languageburo.com/blog/everything-to-know-about-patent-translation (accessed on 8 February 2023).
5. Galvani, T.W. Accuracy and Precision in Patent Writing. 30 August 2019. Available online: https://galvanilegal.com/accuracy-and-precision-in-patent-writing/ (accessed on 11 February 2023).
6. Okpor, M.D. Machine Translation Approaches: Issues and Challenges. *Int. J. Comput. Sci. Issues (IJCSI)* **2014**, *11*, 159–165.
7. Poibeau, T. *Machine Translation*; MIT Press: Cambridge, MA, USA, 2017.
8. Garg, A.; Agarwal, M. Machine translation: A literature review. *arxiv* **2018**, arXiv:1901.01122.
9. Tsarfaty, R.; Seddah, D.; Goldberg, Y.; Kuebler, S.; Versley, Y.; Candito, M.; Foster, J.; Rehbein, I.; Tounsi, L. Statistical parsing of morphologically rich languages (spmrl) what, how and whither. In Proceedings of the NAACL HLT 2010 First Workshop on Statistical Parsing of Morphologically-Rich Languages, Los Angeles, CA, USA, 5 June 2010; pp. 1–12.
10. Stahlberg, F. Neural machine translation: A review. *J. Artif. Intell. Res.* **2020**, *69*, 343–418. [CrossRef]
11. Tan, Z.; Wang, S.; Yang, Z.; Chen, G.; Huang, X.; Sun, M.; Liu, Y. Neural machine translation: A review of methods, resources, and tools. *AI Open* **2020**, *1*, 5–21. [CrossRef]
12. Cifka, O. *Continuous Sentence Representations in Neural Machine Translation*; Institute of Formal and Applied Linguistics: Prague, Czech Republic, 2018
13. Collobert, R.; Weston, J. A unified architecture for natural language processing: Deep neural networks with multitask learning. In Proceedings of the 25th international conference on Machine learning, Helsinki, Finland, 5–9 July 2008; pp. 160–167.
14. Yousuf, H.; Lahzi, M.; Salloum, S.A.; Shaalan, K. A systematic review on sequence-to-sequence learning with neural network and its models. *Int. J. Electr. Comput. Eng. (IJECE)* **2021**, *11*, 2315. [CrossRef]
15. Brownlee, J. How Does Attention Work in Encoder-Decoder Recurrent Neural Networks. MachineLearningMastery.com. 2019. Available online: https://machinelearningmastery.com/how-does-attention-work-in-encoder-decoder-recurrent-neural-networks/ (accessed on 9 January 2023).
16. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention Is All You Need. Advances in Neural Information Processing Systems. 2017. Available online: https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf (accessed on 9 January 2023).
17. Compton, E. Analyzing Japanese with Natural Language Processing and Go. *Medium*, 19 June 2018. Available online: https://medium.com/@enocom/analyzing-japanese-with-natural-language-processing-and-go-64b26cb2436 (accessed on 9 January 2023).
18. Nagao, M. A Framework of a Mechanical Translation between Japanese and English by Analogy Principle. Available online: https://aclanthology.org/www.mt-archive.info/70/Nagao-1984.pdf (accessed on 9 January 2023).
19. Barchas-Lichtenstein, J. Low and High Context Communication. *CultureReady*, 1 March 2016. Available online: https://www.cultureready.org/blog/low-and-high-context-communication (accessed on 9 January 2023).
20. Shih, J. Linguistic differences between Japanese and English. *JBI Localization*, 13 October 2020. Available online: https://jbilocalization.com/linguistic-differences-japanese-english/ (accessed on 9 January 2023).
21. Lauffer, S.; Harold, S. (Eds.) *Computers and Translation: A Translator's Guide*; John Benjamins Publishing Company: Philadelphia, PA, USA, 2003; 349p.
22. WIPO. *World Intellectual Property Indicators 2021*; World Intellectual Property Organization: Geneva, Switzerland, 2021.
23. Nitoń-Greczuk, E. Things to Know about Patent Translation: Textology. *WeAreTextoloxy*, 16 December 2022. Available online: https://wearetextology.com (accessed on 9 January 2023).
24. Madhavan, R. Machine Translation—14 Current Applications and Services. *Emerj Artificial Intelligence Research*, 22 November 2019. Available online: https://emerj.com/ai-sector-overviews/machine-translation-14-current-applications-and-services/ (accessed on 10 January 2023).
25. Pricing Translator: Microsoft Azure. Available online: https://azure.microsoft.com/en-us/pricing/details/cognitive-services/translator/#pricing (accessed on 10 January 2023).
26. 2021 Top 50 US Patent Assignees. IFI CLAIMS® Patent Services. Available online: https://www.ificlaims.com/rankings-top-50-2021.htm (accessed on 10 January 2023).
27. Chatzikoumi. How to evaluate machine translation: A review of automated and human metrics. *Nat. Lang. Eng.* **2020**, *26*, 137–161. [CrossRef]
28. Panić, M. Automated Mt Evaluation Metrics. *TAUS*, 22 July 20200. Available online: https://www.taus.net/resources/blog/automated-mt-evaluation-metrics (accessed on 11 January 2023).
29. Evaluating Models. Google. Available online: https://cloud.google.com/translate (accessed on 11 January 2023).

30. Dorr, B.; Snover, M.; Madnani, N. Part 5: Machine translation evaluation. *Handb. Nat. Lang. Process. Mach. Transl. DARPA Glob. Auton. Lang. Exploit* **2011**, *801–887*.

31. Doshi, K. Foundations of NLP explained. *Medium*, 11 May 2021. Available online: https://towardsdatascience.com (accessed on 11 January 2023).

32. Vashee, K. Understanding Machine Translation Quality: BLEU Scores. 12 April 2019. Available online: https://www.rws.com/blog/understanding-mt-quality-bleu-scores/ (accessed on 11 January 2023).

33. Chauhan, S.; Daniel, P. A comprehensive survey on various fully automatic machine translation evaluation metrics. *Neural Process. Lett.* **2022**, 1–55. [CrossRef]

34. Zhang, Y.; Vogel, S.; Waibel, A. Interpreting bleu/nist scores: How much improvement do we need to have a better system? In Proceedings of the LREC. 2004. Available online: http://www.lrec-conf.org/proceedings/lrec2004/pdf/755.pdf (accessed on 12 January 2023).

35. The Trustees of Princeton University. WordNet. Princeton University. Available online: https://wordnet.princeton.edu/ (accessed on 12 January 2023).

36. Poornima, C.; Dhanalakshmi, V.; Anand, K.M.; Soman, K.P. Rule based sentence simplification for english to tamil machine translation system. *Int. J. Comput. Appl.* **2011**, *25*, 38–42.

37. Terumasa, E. Rule Based Machine Translation Combined with Statistical Post Editor for Japanese to English Patent Translation. In Proceedings of the Workshop on Patent translation. 2007. Available online: https://aclanthology.org/2007.mtsummit-wpt.4.pdf (accessed on 12 January 2023).

38. Şatır, E.; Bulut, H. A Novel Hybrid Approach to Improve Neural Machine Translation Decoding using Phrase-Based Statistical Machine Translation. In Proceedings of the 2021 International Conference on INnovations in Intelligent SysTems and Applications (INISTA), Kocaeli, Turkey, 25–27 August 2021; IEEE: Piscataway Township, NJ, USA, 2021; pp. 1–5.

39. Artetxe, M.; Labaka, G.; Agirre, E.; Cho, K. Unsupervised neural machine translation. *arXiv* **2017**, arXiv:1710.11041.

40. Bharadwaj, S.; Janardhanan, M.T.; Minkuri, P. Translation of Japanese to English language. GitHub. 2020. Available online: https://github.com/bharadwaj1098/Machine_Translation (accessed on 12 January 2023).

41. Team AI. Japanese-English Bilingual Corpus. Kaggle. 2017. Available online: https://www.kaggle.com/datasets/team-ai/japaneseenglish-bilingual-corpus (accessed on 12 January 2023).

42. Tab-Delimited Bilingual Sentence Pairs from the Tatoeba Project. Available online: https://www.manythings.org/anki/ (accessed on 12 January 2023).

43. Neural Machine Translation Background. Tutorial: Neural Machine Translation—seq2seq. Available online: https://google.github.io/seq2seq/nmt/ (accessed on 12 January 2023).

44. Helsinki-NLP/Opus-Tatoeba-en-ja. Helsinki-NLP/opus-tatoeba-en-ja · Hugging Face. Available online: https://huggingface.co/Helsinki-NLP/opus-tatoeba-en-ja (accessed on 12 January 2023).

45. Tiedemann, J . The Tatoeba Translation Challenge–Realistic Data Sets for Low Resource and Multilingual MT. *arXiv* **2020**, arXiv:2010.06354.

46. The Open Parallel Corpus. Available online: https://opus.nlpl.eu/ (accessed on 12 January 2023).

47. Tiedemann, J. Parallel data, tools, and interfaces in OPUS. In Proceedings of Eighth International Conference on Language Resources and Evaluation (LREC'12), Istanbul, Turkey, 21–27 May 2012; Volume 2012, pp. 2214–2218.

48. Collection of Sentences and Translations, Tatoeba. Available online: https://tatoeba.org/en/ (accessed on 12 January 2023).

49. Japanese-English Subtitle Corpus. *JESC*, 12 May 2019. Available online: https://nlp.stanford.edu/projects/jesc/ (accessed on 12 January 2023).

50. Neubig, G. The Kyoto Free Translation Task (KFTT). The Kyoto Free Translation Task. Available online: http://www.phontron.com/kftt/ (accessed on 12 January 2023).

51. Tanaka Corpus—EDRDG Wiki. Available online: http://www.edrdg.org/wiki/index.php/TanakaCorpus (accessed on 12 January 2023).

52. Japanese SNLI Dataset. Available online: https://nlp.ist.i.kyoto-u.ac.jp (accessed on 12 January 2023).

53. Facebookresearch. GitHub. 2019. Available online: https://github.com/facebookresearch/LASER/tree/main/tasks/WikiMatrix (accessed on 12 January 2023).

54. Team, A.I. Japanese-English Bilingual Corpus of Wikipedia's Kyoto Articles [Data Set]. Kaggle. 2018. Available online: https://www.kaggle.com/datasets/team-ai/japaneseenglish-bilingual-corpus (accessed on 12 January 2023).

55. Novak, M. (2012, May 29). The Cold War Origins of Google Translate. BBC Future. Available online: https://www.bbc.com/future/article/20120529-a-cold-war-google-translate (accessed on 9 January 2023).