Milla Koivuniemi

Attention-based Neural Machine Translation: A Systematic Mapping Study

Master's Thesis in Information Technology

May 13, 2020

University of Jyväskylä

Faculty of Information Technology

Author: Milla Koivuniemi

Contact information: m.koivuniemi@iki.fi

Supervisors: Paavo Nieminen and Antti-Juhani Kaijanaho

Title: Attention-based Neural Machine Translation: A Systematic Mapping Study

Työn nimi: Kiintopisteneuroverkkokääntäminen: systemaattinen kirjallisuuskartoitus

Project: Master's Thesis

Study line: Ohjelmointikielten periaatteet (Principles of Programming Languages)

Page count: 86+0

Abstract: Neural machine translation (NMT) is an emerging field of study in machine translation. The leading model for doing neural machine translation seems to be attention-based NMT, in which a part of the source sequence is selected and paid attention to in order to reduce the burden of the encoder. The present thesis is a literature mapping of attentional NMT. The study provides a crosscut of current research in attentional NMT, going over the most popular network features as well as translation quality. Special attention is given to a known problem area, translation of low-resource languages, i.e., languages with only small parallel corpora available. Judging by the papers reviewed, attentional NMT is efficient and produces fluent translation. As a whole, this mapping study produces new and valuable information about the state of research in NMT and provides foundation for different interesting topics for further research.

Keywords: Neural Machine Translation, NMT, Natural Language Processing, Attention-based Neural Machine Translation, Systematic Literature Mapping

Suomenkielinen tiivistelmä: Neuroverkkokonekääntäminen on kasvava konekääntämisen erityisala. Tällä hetkellä suosituin neuroverkkokääntämistekniikka lienee kiintopisteneuroverkkokääntäminen (engl. Attentional Neural Machine Translation, suomennos oma), jossa neuroverkko kiinnittää huomiota käännettävän lauseen tiettyihin osiin vähentäen näin verkon kuormitusta. Tämä pro gradu -tutkielma on kirjallisuuskartoitus kiintopisteneuroverk-

kokääntämisestä, jossa tehdään läpileikkaus käytetyimmistä neuroverkon ominaisuuksista sekä käännösten laadusta. Erityishuomion kohteena on tunnettu kehityskohde, pienen aineiston kielet (engl. low-resource languages), eli kielet, joille on tarjolla vain verrattain pienikokoisia rinnakkaiskorpuksia eli kieliaineistoja. Tutkielman tulosten perusteella kiintopisteneuroverkkokääntäminen on tehokasta ja tuottaa sujuvia käännöksiä. Kokonaisuutena tämä kirjallisuuskartoitus tuottaa uutta kiinnostavaa tietoa neuroverkkokonekääntämisen tutkimuksen nykytilasta sekä luo pohjan erilaisille mielenkiintoisille jatkotutkimusaiheille.

Avainsanat: neuroverkkokääntäminen, luonnollisen kielen prosessointi, kiintopisteneuroverkkokääntäminen, systemaattinen kirjallisuuskartoitus

Preface

The story behind the present study is interesting. Having worked as a translator and written my previous thesis[1] on translation, the topic obviously fascinates me. When the Finnish translators' trade union KAJ (current Kieliasiantuntijat) published an article titled "Neuroverkot valjastetaan kääntäjän apujuhdaksi" (translation: *Neural networks harnessed to aid translators*), I felt that I had found a way to combine my IT studies with my interest in translation. I pitched the idea to my study advisor and the rest is history.

The path to the finished thesis you are reading now was surprisingly straightforward. Despite having little previous experience in neural networks, it was always clear to me what to do next. I owe this mostly to my supervisors, Paavo Nieminen, who advised me with the theory of neural networks, and Antti-Juhani Kaijanaho, who advised me with the methodology. Thank you so much.

I would like to thank my dear study buddies – I loved our little interdisciplinary study sessions and lunches. I would also like to thank Kalle, who proof-read my thesis and did language checking. Kudos to my employer, Cinia, for flexibility and to my coworkers for their support during the writing process. Thanks to all the authors who gave me permission to use their figures. Finally, thanks to Antti for support and for helping me with chores like cooking.

Special thanks to the *Hommat Haltuun* project that arranges interdisciplinary weekend thesis undertakings in Jyväskylä. I participated in their weekend events while working on not only the present thesis but also my previous thesis. May your good work continue in the future.

Jyväskylä

May 13, 2020

Milla Koivuniemi

1. Koivuniemi, Milla. 2017. Translating software instructions: a case study on the translation process of instructions for a subscription software, with special attention to translation problems. Master's thesis, University of Jyväskylä. https://jyx.jyu.fi/handle/123456789/53011.

List of Figures

List of Tables

Contents

1 Introduction

Would you like to travel through space and time in a vehicle like Tardis from Doctor Who that translates everything into your native language? Or have a travel companion like the Babel Fish from The Hitchhiker's Guide to the Galaxy translate every language spoken around you in real time? Such devices rely on instantaneous and automated translation, and for some time, machine translation was not quite there when it came to translation fluency and accuracy. But thanks to recent advances in machine translation, we are now closer than ever.

Over the last few years, Neural Machine Translation (NMT) has gained popularity as it has been found to be more efficient in translation tasks than conventional machine translation methods, such as traditional statistical machine translation (SMT) or rule-based machine translation (RbMT) by, for example, Bentivogli et al. (2016).

This study was inspired not only by a fascination in machine translation but also by a number of recent findings in the field of neural machine translation. In 2015, Bahdanau, Cho, and Bengio (2015) introduced models of neural machine translation that use attention-based models in recurrent neural networks. Here, attention refers to the decoder deciding parts of the source sentence to pay attention to, which reduces the computational burden of the encoder and is therefore more efficient than other neural translators (Bahdanau, Cho, and Bengio 2015). In 2017, Vaswani et al. (2017) introduced the Transformer model, which replaces the widely used sequence-aligned RNN with a self-attention-based model (Vaswani et al. 2017). Currently, it seems that attentional models are the state-of-the-art in NMT, which makes them a relevant topic to study.

NMT is not perfect, of course. Koehn and Knowles (2017) presented a paper on six challenges for neural machine translation. These challenges are 1) quality differences between different domains, 2) small amount of training data, 3) rare words, 4) long sentences, 5) aligning (matching) source and target words, and 6) beam search quality decrease with large beams. Out of these challenges, small amount of training data, more commonly referred to as a *low-resource setting*, was selected as a specific area of interest in the current study. The motivation for selecting this challenge specifically was that the current study is conducted

in University of Jyväskylä, a Finnish university, and translating Finnish-to-English and vice versa is a low-resource setting.

The aim of this study is to answer the following research questions:

RQ1. How actively are papers on attention-based NMT published?
RQ2. What are the features of attention-based neural machine translation models?
RQ3. How well do attention-based NMT models perform in translation tasks?
RQ4. How well does attention-based NMT perform in translation tasks involving low-resource languages?

The method in which this study was conducted was systematic literature review, more specifically using a mapping study as the form of review. There does not seem to be any earlier, in-depth systematic review on this specific topic, which is why a systematic mapping study on the topic is justified to bring forth important information about current research and act as a basis for further research.

2 Theoretical background

This chapter introduces the concepts relevant to attention-based neural machine translation. First, some central concepts of neural machine translation are defined. The central concepts and terminology for this study are neural networks, machine translation, and neural machine translation. After these have been introduced, the concept of attention in the context of NMT is discussed and the general state of research in attention-based NMT is summarised.

2.1 Neural networks

A neural network, or more specifically, an *artificial neural network* is a network of artificial neurons that mimics the learning process of biological organisms (Aggarwal 2018, 1). Neural networks can be trained to complete tasks that are difficult for traditional computer algorithms, such as image recognition tasks (Aggarwal 2018, 3).

2.1.1 Structure of artificial neural networks

Textbooks on artificial neural networks, such as those by Aggarwal (2018, 1) and Bishop (2006, 226), describe artificial neural networks as simulations of biological neural networks that are found in the animal brain[1]. The artificial neurons are computational units (or in the context of network architecture, nodes in the network) that transmit signals between them, similarly to biological neurons that use synapses to pass signals to one another (Aggarwal 2018, 1–2). Figure 1 is a simplification of the network model of artificial neural networks.

The network in Figure 1 is a typical *multilayer feed-forward network*. In general, multilayer neural networks have additional computation layers, also known as *hidden layers*, in addition to the input and output layer (Aggarwal 2018, 17). The architecture is referred to as a feed-forward network because the previous layers feed their output forward to the next layer in vector form, starting from input layer and proceeding to the output layer (Aggarwal 2018, 17), as noted by the arrow connections in Figure 1.

1. However, Bishop (2006, 226) criticises the biological plausability of artificial neural networks and rather calls them "efficient models for statistical pattern recognition".

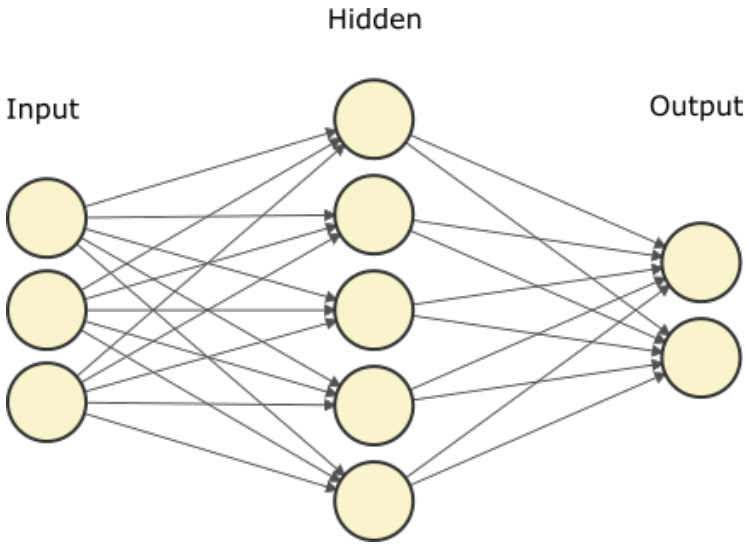

Figure 1. A model of an artificial neural network. This network contains three input neurons, five hidden neurons, and two output neurons. The arrows represent connections between neurons, also referred to as edges.

There are different types of artificial neurons. Two of the most common neuron types are the perceptron and the sigmoid neuron. A *perceptron* is a neuron that takes in several inputs and produces one binary output, effectively a "yes" or a "no" answer, like -1 or $+1$ (Aggarwal 2018, 5), or alternatively 0 or 1 (Nielsen 2015, 3). *Sigmoid neurons* are very similar to perceptrons, but their outputs are not binary. For example, with the definition given by (Nielsen 2015), they can produce any value between 0 and 1. Sigmoid neurons are useful when one is interested in the probability of a certain result (Aggarwal 2018, 11), or when wanting to observe how small changes in variables affect the output (Nielsen 2015, 10). For example, if a sigmoid neuron network is used for classifying if the animal in an image is a cat or a dog, the output of the network produces a certain probability for each case, and gives its answer based on which one has a higher probability.

The output of a neuron is determined by its *activation function*. More specifically, each neuron is given a weight vector w that contains a separate weight coefficient for each corresponding component of the input vector x. The output $y(x)$ of a neuron depends on whether the weighted sum of a neuron minus bias is less than or greater than zero (Nielsen 2015). The weight in the input can be thought of as the importance of the respective inputs to the

output (Nielsen 2015). The bias can be thought of as a negative threshold, so that instead of stating that the output depends on the weighted sum being less or greater than a certain threshold, it is stated that the output depends on the weighted sum plus bias being less or greater than 0.

In a perceptron, the output equation $y(x)$ is simple. Equation (2.1) (adapted from Nielsen 2015, 4) shows the output rule for a perceptron with an output that is either 0 or 1.

$$y(x) = \begin{cases} 0 & \text{if } \sum_j (w_j x_j) + b \leq 0 \\ 1 & \text{if } \sum_j (w_j x_j) + b > 0 \end{cases} \tag{2.1}$$

When the function in Equation (2.1) is plotted, the shape reveals that it is a unit step function. In other words, the activation function of a perceptron neuron is the unit step function, also known as Heaviside step function. Figure 2 is the plotted step function.

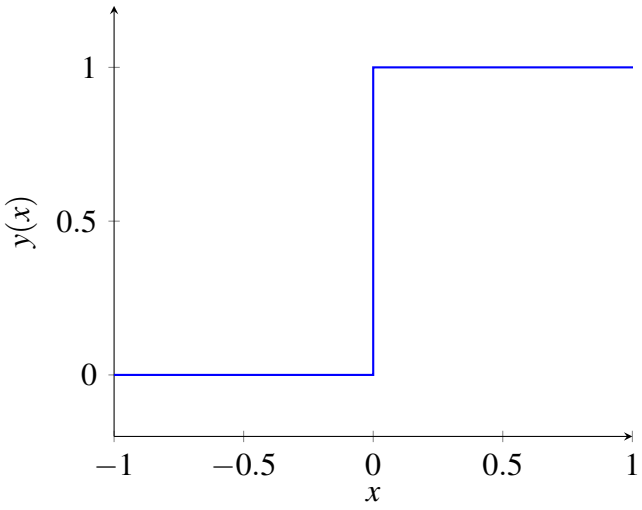

Figure 2. Plotted activation function of the perceptron neutron, the step function

Sigmoid neurons on the other hand have an output between 0 and 1, so a different type of activation function is needed. The output y(x) of the sigmoid neuron is determined by the sigmoid function $\sigma(x)$. The sigmoid function is defined by Nielsen (2015, 8) as Equation (2.2).

$$\sigma(x) = \frac{1}{1 + exp(-\sum_j (w_j x_j) - b)} \tag{2.2}$$

Plotted, the function takes the shape in Figure 3. As can be seen, its shape is like a smoothed out step function.

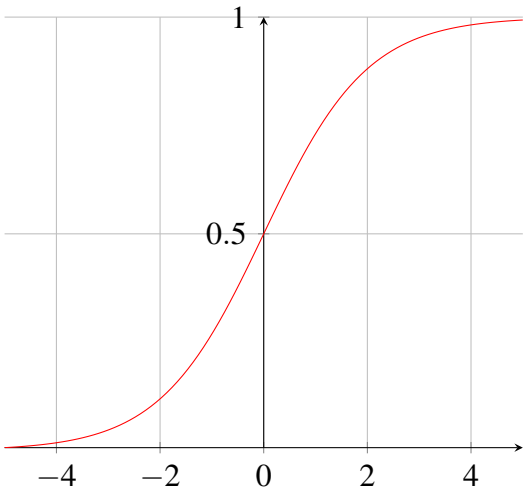

Figure 3. Sigmoid function

Other common activation functions include the rectifier (used in Rectified Linear Units, or ReLus), leaky ReLu (a variant of the rectifier), or hyperbolic tangent (tanh). Activation functions are often named by their associated neuron type, which is why the term 'neuron'/'cell'/'unit' and 'activation function' are often used in articles interchangeably. This was the case in some of the articles that were reviewed for this study.

Some neuron types relevant to networks used in neural machine translation, such as Long Short-term Memory (LSTM) and Gated Recurrent Unit (GRU) will be discussed in Section 2.3.2.

2.1.2 How neural networks learn

Neural networks are trained on a dataset known as the training set. The training set is a set of data with known outcomes, for example, if the network needs to recognise hand-written digits, each image in the training data is linked to the correct digit (Bishop 2006, 2). Then,

the network is presented with input that it has not processed before, known as the test set, and it will try to predict the correct output based on what it has learned from the training set (Bishop 2006, 2). The network's ability to make correct predictions of output based on new input is called *model generalisation* (Aggarwal 2018, 2), or just *generalisation* (Bishop 2006, 2).

The way how neural networks learn is a complex process. To follow the present mapping study, it is sufficient to provide a general, easy-to-understand description and leave investigation of details to the reader's interest (textbooks by Aggarwal 2018; Bishop 2006, are highly recommended). In the following, I will refer to Nielsen (2015), who describes the learning process in a manner that is suitable for the needs here.

The network utilises a training algorithm in learning (Nielsen 2015). The goal is to find an algorithm that finds the right weights and biases so that the network can produce the correct answer in as many tasks as possible (Nielsen 2015). The key in this is to find an algorithm with which the network is accurate, but so that only small changes need to be made to the weights and biases (Nielsen 2015). In other words, *the cost function* needs to be minimised (Nielsen 2015).

However, with neural networks the cost function can be a very complicated multivariate function, which makes it time-consuming or even impossible to simply calculate the minimum analytically (Nielsen 2015). For this reason, we need to use something else to find the minimum. A commonly used method to find the minimum is *backpropagation*. The standard algorithm for doing backpropagation is *gradient descent*.

Gradient descent

In gradient descent, the computation starts at a random starting point, then a gradient vector, which is a vector of partial derivatives of cost function in relation to its components (the weights and biases), is calculated. Then, the weights and biases are adjusted so that we move to the next point with *the opposite of the gradient vector*, and then compute the next gradient and so on (Nielsen 2015). This way, some minimum is finally found, which is hopefully the global minimum, although finding it is nearly impossible to achieve in practice. A good way to visualise gradient descent is by imagining a ball rolling down hills until it reaches the

lowest point of the valley between the hills (Nielsen 2015). This is illustrated in Figure 4.

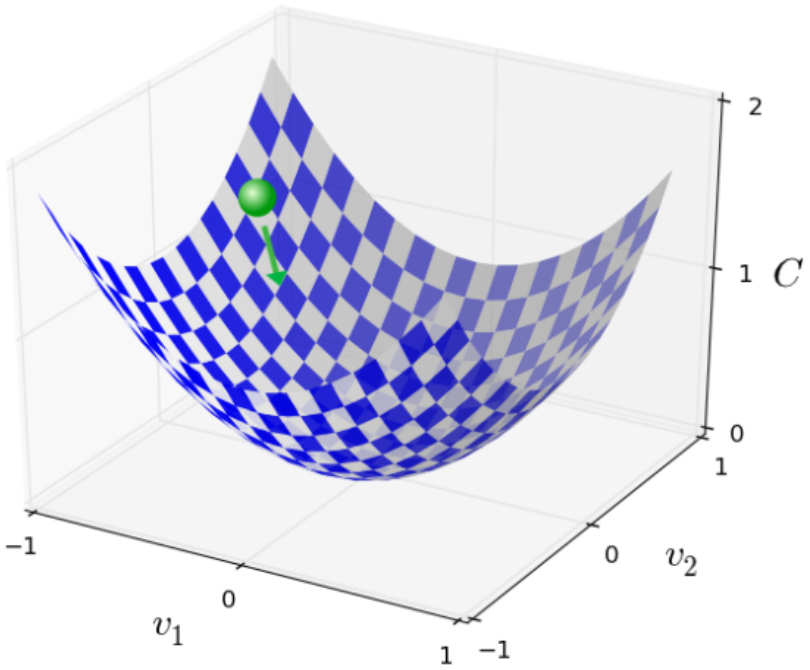

Figure 4. A simplified illustration of gradient descent by Nielsen (2015, 20, used with permission). In this illustration, the function $C(v)$ is minimised by its sole two variables, v_1 and v_2. The arrow represents the gradient descent.

In vanilla gradient descent, the gradient is the average of all the computed gradients of the entire set of training inputs. With a large set of training inputs, this is time-consuming and learning is slow. For this reason, some optimised algorithms have been developed.

Stochastic gradient descent and other learning algorithms

Stochastic gradient descent (SGD) is a popular learning algorithm. In SGD, the gradient is computed for a small sample of randomly chosen training inputs and their average is used to estimate the true gradient for the entire set of training inputs (Nielsen 2015). Since the entire set of inputs does not need to be taken into account, this learning method is faster than

regular gradient descent.

There are also optimisations of the stochastic gradient descent, in other words, SGD variants. Some popular optimisation algorithms include AdaGrad, Adadelta, RMSprop, and Adam. AdaGrad is an adaptive algorithm (Duchi, Hazan, and Singer 2011) that adapts the learning rate by caching the sum of squared gradients and using the inverse of its square root as a multiplier at each time step (Lipton 2015, 9). Adadelta was derived from AdaGrad and it uses a fixed number of past gradients instead of all past gradients as it accumulates the sum of squared gradients, aiming to prevent decay of learning rates through training (Zeiler 2012). RMSprop is an adaptive learning rate method, which adapts AdaGrad by introducing a decay factor in the cache (Lipton 2015, 9; Hinton 2020). Kingma and Ba (2015) describe Adam as a combination of the best properties of AdaGrad and RMSProp: the ability to deal with both sparse gradients as well as non-stationary objectives.

Deep learning

Deep neural networks are networks with multiple hidden layers (Nielsen 2015, 37). Learning in deep neural networks has been enabled since 2006 with the help of techniques that have made the learning process much faster than before (Nielsen 2015, 37, 204), such as the greedy learning algorithm in deep belief networks by Hinton, Osindero, and Teh (2006).

2.2 Machine translation

Machine translation (MT) is a sub-field of computational linguistics in which software is utilised to translate natural language text or speech from one language to another. The earliest experiments with machine translation started in the 1950s, closely following the advent of computers (Koehn 2009) and aided by the rise of structuralist linguism (Nord 2014).

As of late, the field of machine translation has also expanded from traditional natural language translation tasks to a broader concept of translation, that of translating information and/or meaning to another form, for example the “translation“ of photographs to paintings and vice versa (see e.g. Stein 2018). However, the focus of this study is machine translation of natural language in text form.

There are several models for doing machine translation. The two major models are rule-based machine translation and statistical machine translation. I will now briefly discuss rule-based and statistical machine translation.

2.2.1 Rule-based machine translation

Rule-based machine translation (RbMT) generates translations based on an analysis of the linguistic properties of the source and target language. RbMT utilises dictionaries and grammar to produce an analysis of the semantic, morphological, and syntactic construction of the source language input and then translate it to the target language equivalent output.

2.2.2 Statistical machine translation

Conventional statistical machine translation (SMT) generates translations based on statistical models. SMT makes use of parallel corpora in deriving the parameters for these models (Koehn 2009). Corpora are collections of texts, and in parallel corpora texts are paired with a translation of the text to another language. In other words, SMT is a data-driven approach to MT (Koehn 2009).

It is usual for SMT (as for other types of natural language processing) that raw text is broken into smaller, atomic units. There are different models of SMT based on what the unit of translation is. In *word-based translation*, the unit of translation is a single word. In *phrase-based machine translation*, the unit of translation is a phrase, or a phraseme, which is a statistical unit (not to be confused with linguistic phrases). Phrase-based machine translation was the most effective model for doing machine translation until about 2015 when another form of statistical MT, neural machine translation, started producing comparable results (Bentivogli et al. 2016). Neural machine translation will be discussed further in Section 2.3.

2.2.3 Quality evaluation for machine translation

The quality of translation produced by machine translation needs to be evaluated in some way to ensure the adequacy and fluency of translated text. Human evaluation is probably the most accurate metric, but it is also subjective, making it hard to compare results of one

MT model with another one. Furthermore, the set of translated sequences can be quite large, for example thousands of sentence pairs, making human evaluation also extremely time-consuming. For this reason, some automated metrics for measuring the quality of translation have been developed.

One of the most widely used translation quality metrics is BLEU, short for "bilingual evaluation understudy", which is a method of automatic machine translation evaluation (Papineni et al. 2002). Working on the sentence level, the basic idea of the score is to look at the set of target translation sentences (reference translations) for a given source sentence and compare them with the translated sentence produced by the MT system (candidate translation). The more there are matches between candidate translation and the reference translation, the better the score. Papineni et al. (2002) specifically tested the metric on a set of reference translation sentences, i.e., multiple adequate translations, although implicitly there can also be just one reference sentence. It is also notable that matches are position-independent, meaning that word-order is not considered. The fact that the score is based on comparison to reference translation inherently requires access to reference translations, e.g. by retrieving sentences to translate from parallel corpora. According to experiments by Papineni et al. (2002), the BLEU score correlates highly with human evaluation.

The BLEU score was originally a score between 0 and 1.0, with 0 being worst and 1.0 being best. However, it is more common in literature to present the score multiplied by 100, resulting in scores between 0 and 100. Rikters (2019) estimated that at the time of writing, state-of-the-art machine translation systems usually scored between 20 to 40 points on the BLEU metric. The data in this study is in line with this claim, however, the score is often lower than 20 in low-resource settings. It is uncommon for even a human translator to score close to 100, because it would require the use of exactly the same phrases as in the reference translation.

2.3 Neural machine translation

Neural machine translation (NMT) is an approach to doing statistical machine translation that utilises neural networks in machine translation. In the recent years, neural machine

translation has started to challenge the dominance of the phrase-based approach to SMT. For a long period of time, NMT was too computationally costly and resource-demanding to be useful, but this changed around 2015 when MT techniques utilising neural models proposed by e.g. Cho et al. (2014a) started to become lighter and performed comparably to phrase-based models with English-to-German translation tasks (Bentivogli et al. 2016). Neural machine translation has quickly caught the interest of the research community and the general public with its impressive results.

2.3.1 Typical network models in neural machine translation

It is necessary to introduce a few common network models in neural machine translation in order to understand the concepts and terminology used in the present study. These are *recurrent neural networks* and the *encoder-decoder architecture*.

Recurrent neural networks

Recurrent neural networks (RNNs) are feed-forward networks that are especially suited for processing sequential data like text sentences, where words depend on previous words (Aggarwal 2018, 38–39). In other words, RNNs introduce context to feed-forward networks. RNNs add temporal information to the network with the help of a time-stamp and a hidden state (Aggarwal 2018, 39; Lipton 2015, 10). A sequence is processed one word at a time, and at each time step, the hidden state is updated and used to process the next word (Aggarwal 2018, 39).

Encoder-decoder architecture

A common network model in neural machine translation is the encoder-decoder network. Cho et al. (2014b) introduced the network model called the *RNN encoder-decoder*, where both the encoder and the decoder were each an RNN individually. This model quickly became popular and has been adopted in many neural machine translation models ever since.

The encoder-decoder network in NMT consists of two networks: the encoder that reads the source sentence and encodes it into a fixed-length vector representation, and the decoder that reads the encoded vector and outputs the target translation. The system is connected by

a joint training process between the encoder and decoder that helps in achieving a correct translation (Bahdanau, Cho, and Bengio 2015).

2.3.2 Hidden layer computation units in neural machine translation

The hidden layers in neural networks can consist of different types of nodes. Traditional nodes are just like every other node in the network, in other words, a neuron that gets its value from applying a function to a weighted sum of its input values (Lipton 2015, 7, 17). However, the problem with standard hidden units is that the derivative of the error (gradient) can vanish or explode over time. For this reason, different hidden units have been developed. The most common hidden units in the neural machine translation context are the Long Short-Term Memory and the Gated Recurrent Unit.

Long Short-Term Memory (LSTM)

Long Short-Term Memory (LSTM) was developed by Hochreiter and Schmidhuber (1997) to prevent the error gradient from decaying over time, causing it to either vanish completely or grow exponentially. As the name implies, LSTM is a memory cell. It has an input gate unit and an output gate unit which control the error flow from and to the memory cell. This ensures that the error flow is constant and does not decay. The gates can relay information about the state of the network for decision-making, for example the input gate may use input from other cells to make decisions about what information to store in the cell. The LSTM model was later expanded with the introduction of a third gate, the forget gate (Gers, Schraudolph, and Schmidhuber 2002). The forget gate defines how long the information should be stored by resetting the memory cell's state when stored information becomes irrelevant. LSTM structure with all three gates and unit-to-unit connections are illustrated in Figure 5.

Gated Recurrent Unit (GRU)

Gated Recurrent Unit (GRU) is a hidden unit proposed concurrently with the introduction of the RNN encoder-decoder architecture (Cho et al., 2014b). This unit features a reset gate and an update gate that control how much information is remembered or forgotten. The update gate is similar to the memory cell in LSTM in that it controls how much information from the previous hidden state will be relayed to the current hidden state (remembering long-

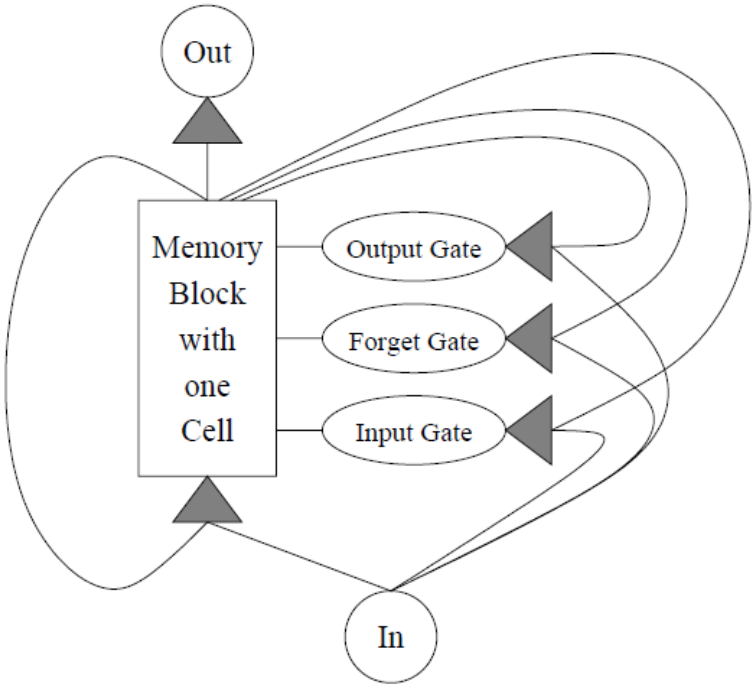

Figure 5. Illustration of LSTM topology from Gers, Schraudolph, and Schmidhuber (2002, 124, used with permission). Network has one input and one output unit, while the hidden layer consists of a single LSTM memory cell. Arrows represent unit-to-unit connections in the self-connected network. There are nine connections, making this a three-layer LSTM.

term information). The reset gate drops information, so it is similar to the forget gate in LSTM (Cho et al., 2014b). Choi (2019) describes GRU as a simplified version of LSTM and especially useful in completing language-related tasks. The activation function is illustrated in Figure 6.

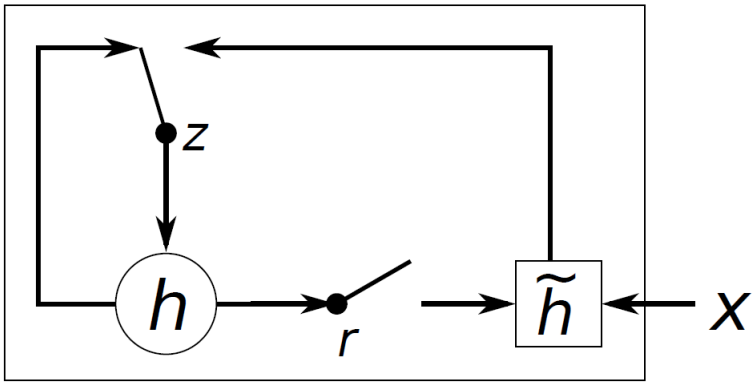

Figure 6. Illustration of the GRU activation function from Cho et al. (2014b, 1726, used with permission). x is the input. z is the update gate, which decides whether to replace the current state with the new hidden state $\tilde{h}$. r is the reset gate that decides if the previous hidden state is taken into account or ignored.

2.3.3 Research on neural machine translation

Neural machine translation is a popular area of research in general, and it also has a growing trend. Search for the search string 'Neural Machine Translation' on Google Scholar produced 631,000 hits all time, while the search for the literal search string "Neural Machine Translation" on Google Scholar produced ca. 14,200 hits. Observing results per year showed that there is a growing trend, in other words, the number of results increases over time.

The top results for neural machine translation are articles that utilise the attention-based approach. This shows that it is the most prominent model for doing NMT at the present. I will now present the results from two articles that compare NMT with other SMT approaches, and then move on to attention-based models in the next section.

Cho et al. (2014a) used two NMT models, RNN encoder-decoder (RNNenc; Cho et al 2014)

and gated recursive convolutional neural network (grConv) for doing machine translation, and compared the results with the performance of a phrase-based machine translator. Both models were trained with a minibatch stochastic gradient descent using AdaDelta. They tested the models for English-to-French translation. Cho et al. (2014a) found that while their models did not perform as well as the phrase-based machine translator they compared them to, both of these models performed well in translation tasks and that there is a future for purely neural machine translation. They also found that both of the models perform poorly with long sentences, which was, in their opinion, something that future research could address (Cho et al., 2014a).

Bentivogli et al. (2016) compared the translation results of three phrase-based MT systems and one NMT system in translating English to German. They found that the NMT system performed better than the PbMT systems (Bentivogli et al. 2016). The NMT system had greatest difficulties with long sentences and with reordering linguistic constituents that require a deeper understanding of meaning in the text, which is why Bentivogli et al. (2016) conclude that there is still work to do in perfecting machine translation.

When we compare the findings of Cho et al. (2014a) and Bentivogli et al. (2016), we notice that even while the sentence length problem still persisted, NMT clearly improved and managed to get past PbMT in performance in the short time of just two years. The reason for this improvement may lie in the introduction of the attention-based NMT model.

2.4 Attention in neural machine translation

In the context of neural networks, attention refers to the decoder part of the neural network deciding parts of the source sentence to pay attention to. The attention mechanism was proposed by Bahdanau, Cho, and Bengio (2015). The attention mechanism makes NMT more computationally affordable (Bentivogli et al. 2016).

The attention mechanism was proposed to alleviate the encoder's difficulty with encoding long source sentences in the encoder-decoder model. As was described in Section 2.3.1, the encoder usually encodes the entire source sentence into a fixed-length vector. In the model with attention mechanism, the translated word is predicted based on most relevant

information in the source sentence and the previous generated target words (Bahdanau, Cho, and Bengio 2015). Searching for most relevant parts of the source sentence reduces the burden of the encoder, because it does not have to encode all information in the source sentence into a fixed-length vector (Bahdanau, Cho, and Bengio 2015).

Interestingly enough, Bahdanau, Cho, and Bengio (2015) describe the model not as attentional, but as an *alignment model*. The term *attention mechanism* became common later on. The alignment model itself is a feedforward neural network that is trained jointly with the rest of the system (Bahdanau, Cho, and Bengio 2015). The alignment model computes a soft alignment that allows to use the gradient to train the whole translation model jointly (Bahdanau, Cho, and Bengio 2015).

2.4.1 Implementation of the attention mechanism

The implementation of the attention mechanism is quite simple. The following is an explanation of the attention mechanism using the terminology that Bahdanau, Cho, and Bengio (2015) used. Whereas a traditional encoder-decoder network has a decoder that is trained to predict translation of a word $y_{t'}$ on basis of the context vector c and all the previous translated words $\{y_1, ..., y_{t'-1}\}$, the attentional decoder has a different type of context vector c_i that is a weighted sum of annotations. Each annotation h_j contains information about the input with focus on parts surrounding the j:th word. In short, the i-th context vector c_i is the expected annotation over all the annotations $h_1 ... h_{T_x}$. Annotations also have associated probabilities a_{ij}. The following equation (Bahdanau, Cho, and Bengio 2015, 3) represents context vector c_i:

$$c_i = \sum_{j=1}^{T_x} a_{ij} h_j \tag{2.3}$$

a_{ij} is a weight (or probability) that the target word y_i is a translation of a source word x_j (i.e., that the words are aligned). Then, the probability a_{ij} reflects the importance of the annotation h_j to the previous hidden state s_{i-1} in deciding what the next state s_i and what the generated translation y_i is. The network model is illustrated in Figure 7.

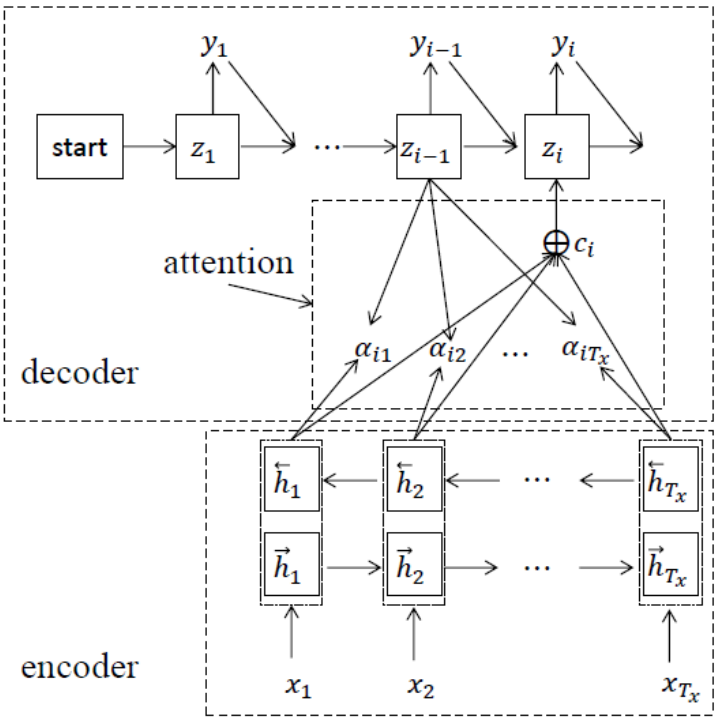

Figure 7. NMT encoder-decoder with attention, as illustrated by Zhang and Zong (2016, 1536, used with permission). All symbols are the same as in the description above, but here, the decoder hidden state is marked with z_i instead of s_i.

Alignment between source sentence words and target sentence words can be visualised by observing annotation weights a_{ij} on a matrix. Figure 8 shows an example of such alignment matrix from Bahdanau, Cho, and Bengio (2015).

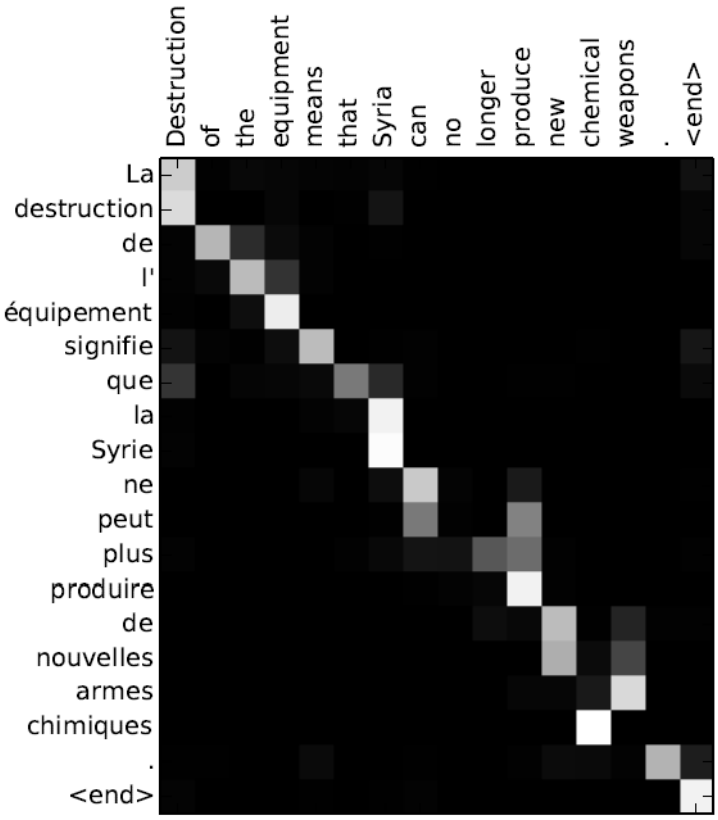

Figure 8. An example of an alignment matrix from Bahdanau, Cho, and Bengio (2015, 6, used with permission). The y-axis represents the English source sentence words and the x-axis represents the generated French translation. Grayscale pixels show the weight a_{ij} of the annotation of j-th source word to the i-th target word. Black indicates that weight = 0 (not likely equivalent), while white indicates that weight = 1 (most likely equivalent).

Bahdanau, Cho, and Bengio (2015) had promising results with the proposed attentional model. They compared two models, the RNN encoder-decoder (RNNencdec), proposed by Cho et al. (2014a) and Sutskever, Vinyals, and Le (2014), and their own proposed model, referred to as RNNsearch. Both models were trained with a minibatch stochastic gradient descent using AdaDelta. They tested the models for English-to-French translation. They found that their RNNsearch model outperforms the RNNencdec model significantly. Their model

also performed well regardless of source sentence length. The most significant finding was that the performance of their model is comparable to phrase-based machine translation models. Bahdanau, Cho, and Bengio (2015) found that there is still work to do with improving the translation of unknown or rare words.

2.4.2 Research on attention-based neural machine translation

Search for the search string "neural machine translation, attention OR attention-based OR attentional" on Google Scholar produced 199,000 hits all time. Once again, the number of articles per year increases over time, so a growing trend can be detected. Next, I will sum up the findings of some most cited articles, however, the papers that were reviewed as part of the mapping study are presented in Chapter 5.

Luong, Pham, and Manning (2015) used RNN with a Long Short-Term Memory (LSTM) hidden unit for encoder and decoder in their NMT model. The model was trained with a minibatch using plain stochastic gradient descent. They also used different attentional models (global, local-m, and local-p). The model was tested for English-German-English translation. Luong, Pham, and Manning (2015) found that attention-based NMT models are superior to non-attentional ones in many cases, for example in translating names and handling long sentences.

Ha, Niehues, and Waibel (2016) presented their first attempts in building a multilingual Neural Machine Translation framework using a unified approach. The goal was "to employ attention-based NMT for many-to-many multilingual translation tasks" (Ha, Niehues, and Waibel 2016). Ha, Niehues, and Waibel (2016) found that their approach is especially effective in an under-resourced translation scenario, achieving a higher translation score. Ha, Niehues, and Waibel (2016) also state that their approach achieved promising results in translation in cases where there is no direct parallel corpus present for the language pair.

2.5 Previous reviews on neural machine translation

Based on database search results, there have been multiple reviews on neural machine translation, but few on attentional neural machine translation. The search string "neural machine

translation review, OR survey“ on Google Scholar produced 80,900 hits all time and circa 17,400 hits since the year 2015. Judging from the top relevant results for this search, there have been surveys and reviews on neural machine translation, but none have focused on attention-based NMT especially. During the research process, one survey on attention-based NMT emerged (Basmatkar, Holani, and Kaushal 2019), presented at conference in March 2019 and added to IEEE Xplore in August 2019. I will now go over some interesting and relevant reviews on NMT.

Concerning existing literature reviews, some of the top results have a quite superficial look at research. Lipton (2015) present the technological aspects behind using RNN for sequence learning in general, not only for translation. Chaudhary and Patel (2018) study the use of deep neural networks in machine translation by comparing research papers. However, they only conclude that deep learning is better than other methods in machine translation, without providing an in-depth analysis of the articles or the performance of the models.

Going into a more practical direction, Basmatkar, Holani, and Kaushal (2019) conducted a survey in which they studied the efficiency of different attentional NMT models on translation between six Indian language pairs and also English-to-Tamil translation. The survey was not a literature review, however, but rather a comparison of performance of different models. For English-Tamil translation, Basmatkar, Holani, and Kaushal (2019) compared two attentional models: Luong, Pham, and Manning (2015) and Bahdanau, Cho, and Bengio (2015). They achieved the best score (on the BLEU evaluation metric) for English-Tamil translation with a birectional LSTM with word embedding and Bahdanau's attention model, using Adam optimiser, with a byte-pair encoding of 25,000. They also compared their results with Google Translator and found that all their models achieved significantly better BLEU scores than Google Translator.

Britz et al. (2017) experimented with different NMT system parameters in their extensive exploration of NMT architectures. The features that they explored were RNN cell variant, network depth, unidirectional vs. bidirectional[1], attention mechanism, embedding dimensionality, and beam search strategy. They found that the best performing model was an

1. Unidirectional encoders take only past inputs into account, while bidirectional ones take both past and future inputs into account (Britz et al. 2017, 1446).

LSTM network with a bidirectional encoder with depth 4 and a decoder of depth 4 with the Bahdanau, Cho, and Bengio (2015) attention model. Both the attention dimension and the embedding dimension were found to be optimal at 512. Beam size, i.e., how many most probable predictions for the translated word are retrieved in the translation model, was found to affect results significantly and the best beam size was found to be 10. Somewhat surprisingly, Britz et al. (2017) also found that deep models are not always better than shallow ones. They tested the performance of the best model with the newstest2014 and newstest2015 English-to-German task, using SGD and Adam as learning method as well as word embedding and batch size of 128. They compared their system to nine different NMT models, and their model was only outperformed by the model by Wu et al. (2016), but as the authors note, the model by Wu et al. (2016) lacks public implementation and is more complex. The exploration by Britz et al. (2017) is practical and is not directly comparable to the results of a literature review, but it can be beneficial to view the results of the present study in light of findings by Britz et al. (2017).

Based on the search results, there is a gap in systematic literature reviews on attentional neural machine translation. According to Kitchenham, Budgen, and Brereton (2016), a review can be seen as necessary if no good quality review exists already. This seems to be the case for this topic, so conducting a review is justified. Furthermore, it seems that there is not enough reviews on this topic to perform a tertiary study, i.e., a review of existing reviews.

3 Research design

In this chapter, I will go over the details of the research design for this study. First, I will go over the research questions and how they will be answered. Second, I will present and justify the used research method. Then, stages of the review are briefly discussed, followed by a description of the search process. Next, I will list the inclusion and exclusion criteria, as is usual for systematic reviews. This is followed by a description of data synthesis and aggregation. Finally, I will briefly discuss the time frame and limitations for the present study.

3.1 Research questions

The goal of this study is to describe the state of research on the topic of attention-based neural machine translation in the recent years. The study looks into the research settings as well as the results of the studies on attentional NMT. An additional aim is to review how well attention-based models perform in one known problematic translation context, translation tasks involving low-resource settings.

The aims of the study have been formulated as the following research questions:

RQ1. How actively are papers on attention-based NMT published?
RQ2. What are the features of attention-based neural machine translation models?
RQ3. How well do attention-based NMT models perform in translation tasks?
RQ4. How well does attention-based NMT perform in translation tasks involving low-resource languages?

The first question is a general question answered simply by providing statistics of search engine hits for keywords. This is not by any means an exhaustive answer, but it provides a glimpse of the body of literature that exists. The second question is answered with statistics and analysis of what structural neural network features were present in the models that the authors have developed. Structural features include, for example, the neural network architecture type, learning methods, activation functions, optimisations, and computational

units.

The third question is answered based on comparing the authors' own analysis on performance, which was usually provided with BLEU scores, but sometimes also word alignment results and qualitative analysis. Comparison of performance also takes different language pairs into account. The fourth question will be answered with roughly the same means as the third question: comparing BLEU scores and other types of quality measures. My initial hypothesis was that, as with other types of MT, attention-based NMT also performs poorly on low-resource language translation tasks. This hypothesis was based on the notion that since NMT is data-driven, naturally the lack of data affects the performance, regardless of the technique in which MT is done.

At the beginning of this study process, the aim was to form an all-encompassing overview of the current state of research on the topic of attention-based neural machine translation, to provide a summary of the current research in attention-based neural machine translation, and to identify the current limitations of research on attention-based models. However, the search process revealed that the body of literature was far too extensive (over 44,000 articles), which makes is extremely challenging to form an exhaustive overview of attentional NMT research overall. For this reason, the aims and research questions were reformulated to fit a smaller cross-section study. Kitchenham, Budgen, and Brereton (2016) state that the amount of work in doing the review task should be feasible considering the resources of the one doing it. This has also been considered in that the topic and the amount of literature for the review have been narrowed down to suit the requirements and amount of work suitable for a master's thesis.

The original aims were formulated as the following original research questions: 1. How actively are papers on attention-based NMT published? 2. How do purely attention-based neural machine translators perform in relation to other NMTs? 3. What are the limitations of attention-based neural machine translation? and 4. How well does attention-based NMT perform in translation tasks involving a low-resource language? Questions 1 and 4 were kept, but question 2 and 3 were discarded altogether. The reason for discarding question 2 is that authors more often compared their results with other attentional NMT tools rather than with non-attentional models, which is why the data could not provide a good answer to this

question. Question 3 was discarded in order to focus on one known limitation, low-resource setting, instead of trying to include all possible limitations.

3.2 Method

The current study is an exploratory research in the form of a literature survey. The method used in this study is systematic literature review. The review was done with the conventions proposed by Kitchenham, Budgen, and Brereton (2016). Kitchenham, Budgen, and Brereton (2016) outline a method for conducting systematic literature reviews in the field of software engineering. Because of its focus on this specific field, this method is suitable for the purpose of this study.

3.2.1 Justification for used method

According to Kitchenham, Budgen, and Brereton (2016), the motivation for doing systematic reviews is usually:

- gathering knowledge about the field of study in question,
- identifying the needs for future research,
- establishing the context of a research topic, and
- identifying the main methodologies and research techniques for the field of study in question.

These motivation criteria are in line with the aims and research questions of the current study, which is why it is justifiable to use this research method for this study.

The present study was motivated by the small number of systematic reviews on the topic. Kitchenham, Budgen, and Brereton (2016) emphasise that before doing a systematic review or a mapping study it is important to think about whether the review will provide new knowledge in the field of study. The motivation for this study was the prominence of attention-based models in current NMT research, and judging from the search process conducted for this thesis, there is not only a small amount of reviews on the topic of NMT in general, but also very few seem to concentrate on attention-based models especially. For these reasons,

the current study is justified in hopefully providing new valuable knowledge.

3.2.2 Mapping study as a form of review

There are different forms of doing evidence-based literature review, i.e., methods for the process of combining research data to form new knowledge. The book by Kitchenham, Budgen, and Brereton (2016) covers doing systematic reviews with three review types: quantitative, qualitative, and mapping study, which are all suitable for this study. Mapping study is the review form selected for this study.

According to Kitchenham, Budgen, and Brereton (2016), a mapping study is usually a general classification of the analysed data. It is usually used for clustering data for more in-depth systematic reviews and for identifying gaps in existing literature. Due to its simplicity and general nature, it is suitable for the scope of a master's thesis.

3.3 Stages of the review

According to Kitchenham, Budgen, and Brereton (2016), the review project first needs to be justified and the research questions need to be specified. Both of these were presented earlier in this study. Then, the review protocol, which is a documented plan of how the review will be conducted needs to be developed (Kitchenham, Budgen, and Brereton 2016). The research plan I wrote was the protocol for this review. The research plan included all the review protocol parts outlined by Kitchenham, Budgen, and Brereton (2016), namely 1) background, 2) research questions, 3) search strategy, 4) study selection, 5) quality assessment of the primary studies, 6) data extraction, 7) data synthesis, i.e., the plan for analysing the data, 8) limitations, 9) reporting, and 10) review management, i.e., making sure that the review project is sensible, manageable and done properly (Kitchenham, Budgen, and Brereton 2016).

3.4 Search process

In this section, I will describe the search process.

3.4.1 Search method

Kitchenham, Budgen, and Brereton (2016) introduce a variety of search methods, which were applied to some measure in this study. From these, I chose to conduct an automated search from electronic resources. Kitchenham, Budgen, and Brereton (2016) outline that this method requires 1) deciding which resources to use and 2) specifying the search strings that drive the search. The following sections describe the resources and search strings used.

3.4.2 Search engines and databases

I searched for articles and books on the topic of neural machine translation and attention-based NMT from Google Scholar and Web of Science. For Web of Science, the available databases via University of Jyväskylä were Science Citation Index Expanded (1945–present), Social Sciences Citation Index (1956–present), Arts & Humanities Citation Index (1975–present), and Emerging Sources Citation Index (2015–present).

3.4.3 Electronic resources

Search was conducted with the Google Scholar and Web of Science search engines. The results from these were filtered according to the selection criteria, which will be introduced later in this section.

Primary electronic resources for this study include Arxiv, IEEE Digital Library, and the ACM Digital library (the latter two suggested by Kitchenham, Budgen, and Brereton 2016). Arxiv contains many of the most relevant articles related to the topic, including the pioneering article by Bahdanau, Cho, and Bengio (2015), as well as most conference proceedings for relevant conferences. IEEE Digital Library and the ACM Digital library are available via University of Jyväskylä. Also Web of Science finds articles from both of these (Kitchenham, Budgen, and Brereton 2016).

3.4.4 Search strings

To find articles to review, the following search strings and their variants were used:

- Neural Machine Translation AND Attention
- Neural Machine Translation AND Attention-Based
- Neural Machine Translation AND Attentional
- Neural Machine Translation AND (Survey OR Review)

3.5 Inclusion and exclusion criteria

To narrow down the set of articles considered for review, some selection criteria were applied. First, the topic of the article needed to be neural machine translation and clearly stated as so. For example, the title or the abstract needed to refer to neural machine translation, or, if there were keywords given in the articles reviewed, they included "neural machine translation", either as one keyword or a combination of keywords (a combination can for example be "neural networks" and "machine translation"). Additionally, the approach used in the study had be in some way attention-based and clearly stated as so. One exclusion criterion was that the title and/or abstract of the article indicates that its focus is on something other than translation of written texts from one language to another, for example, on speech recognition or multimodal translation.

Since the language used in this study is English, only articles that were written in English were chosen. Furthermore, one of the languages in the language pairs (or sets) studied in the study that the article reports had to be English, so that comparisons between findings could be made.

One criterion was that papers needed to be peer-reviewed to be considered for inclusion. Kitchenham, Budgen, and Brereton (2016, 68) mention that reviews typically include only peer-reviewed articles, leaving out, for example, technical reports and PhD theses. All journal articles included in the present study were published in peer-reviewed journals. Articles published as part of conference proceedings were also all peer-reviewed.

Only papers that had the full text available via university credentials were reviewed. This directly resulted in the exclusion criterion that any studies reported as abstracts or only available as abstracts or other types of texts, such as presentations or blog posts, were excluded.

Since the origin of the attention-based approach to NMT was established in year 2014, it made sense to exclude all studies that were published before 2014.

Finally, since the present study is a master's thesis, it was necessary to limit how many search results were considered as the set from which reviewed articles were selected, in other words, the candidate papers (a term used by Kitchenham, Budgen, and Brereton 2016). Furthermore, since the initial searches returned over 200,000 results, it would have been unnecessarily time-consuming to go through all search results. For this reason, whenever the number of search results was very high, the candidate papers were selected from among the first 50–60 search results, while the rest of the results were discarded altogether. The sorting method for the results was "according to relevance" as determined by the search engine. A similar limiting method was used by Pozdniakova and Mazeika (2017).

In summary, the inclusion criteria were:

- The topic is neural machine translation
- The model studied in the article is attentional
- Peer-reviewed

The exclusion criteria were:

- Language of the article is not English
- Domain is not text translation, e.g. domain is spoken language
- One of the languages in the studied language pair is not English
- Full text is not available via University of Jyväskylä student credentials
- Text is a PhD thesis, technical report, or presentation
- Published before 2014
- Ranks after 60 first hits in results for searches with a large number of hits

3.6 Data synthesis and aggregation

The method used in this study was mapping study. A mapping study is a useful method for categorising papers to form general summaries of research on the topic, like the current study does. Mapping studies usually utilise gathering data into clusters and analysing them in light

of the research questions. Clustering data is especially good for identifying areas for more detailed study and gaps in current research (Kitchenham, Budgen, and Brereton 2016, 315). The generalist nature of the mapping study entails the use of certain means of data synthesis and aggregation.

The data synthesis method in this study was formed in two ways: 1) determined by the research questions and 2) inductively. This is in line with Kitchenham, Budgen, and Brereton (2016, 351–353) who emphasises that while there is no standard way for doing synthesis, there should at least be a clear link from the research questions to data and syntheses. The research questions determined which large categories data synthesis concerned, while the details of these categories were determined inductively, i.e., by aspects that emerged in the review process. Clustered data can roughly be divided into four categories: publication details, research setting, neural network model, and translation evaluation method.

Clustering data of publication details partially answers research question RQ1. *How actively are papers on attention-based NMT published?* (the main data for answering the question is the search hits in general). Data clusters per publication details are:

1. Author name
2. Publication year
3. Publication type

Documenting research setting details is relevant to answer research questions RQ3. *How well do attention-based NMT models perform in translation tasks?* and RQ4. *How well does attention-based NMT perform in translation tasks involving low-resource languages?*, because this background information is necessary to be able to compare performance results properly. Data clusters per research setting include:

1. Language-pair
2. Dataset (corpora or task used in analysis)

Information about network models was collected to answer question RQ2. *What are the features of attention-based neural machine translation models?*. Ways to cluster data per neural network model:

1. Network model
2. Learning method(s)
3. Activation function(s)
4. Computational unit(s)

Finally, clustering data about translation evaluation results answers two questions: RQ3. *How well do attention-based NMT models perform in translation tasks?* and RQ4. *How well does attention-based NMT perform in translation tasks involving low-resource languages?*. Ways to cluster data per translation evaluation method:

1. BLEU score (linear)
2. Other linear scores
3. Human evaluation

The review results were aggregated into tables. According to Kitchenham, Budgen, and Brereton (2016), it is common in mapping studies to aggregate primary study features into tables.

3.7 Time frame

The data was gathered within the timeframe that was planned for the thesis, which was between October 2018 and May 2020. While the preliminary search was conducted already in October 2018, the actual search process took place in the latter half of 2019. The papers reviewed in the present study were published between 2014 and 2019.

3.8 Limitations

The current review is necessarily nonexhaustive. This is due to the limited scope of the work and the exclusion criteria listed above. Especially the number of papers reviewed as well as the criterion that only the first 60 results were considered for candidate papers derives from the usual scope of a systematic review, which can have a set of candidate papers consisting of hundreds or thousands of papers. This was justifiable because of the scope of the work and working alone (usually reviews are done in researcher teams).

4 Search and data extraction results

This chapter is a summary of search results and data extraction. Here, I will describe search results, go through all the selection rounds, and present the final set of candidate papers in numbers. The results of the analysis will be presented in Chapter 5.

4.1 Search plan

The search process was planned as consisting of preliminary search, first search, first selection round, followed by other search and selection rounds, if necessary, and finally the final selection round. During the research process, the first selection round was found sufficient and thus it was immediately followed by the final selection round.

Kitchenham, Budgen, and Brereton (2016) outline different selection criteria to apply to selecting the articles for review, which were utilised in the search process. Kitchenham, Budgen, and Brereton (2016) point out that initial criteria can include selecting relevant articles based on title, keywords, and abstract of the paper.

The preliminary search rounds were conducted when drafting the research plan. The first selection round was based on titles, abstracts, and keywords of the papers. The papers for the first selection round were the first 50–60 results of each search, since it was an exclusion criterion to discard all search results after the first results. The final selection round focused on the contents of the paper in more depth, as well as taking the quality and inclusion/exclusion criteria into account. This is in line with findings of the usual review process by Kitchenham, Budgen, and Brereton (2016).

4.2 Summary of papers found at different stages of the process

In this section, I will present the search strings used, the results for each string, and how many papers were selected for review.

4.2.1 Search string results in numbers

The following search strings were used in Google Scholar:

Sch1. neural machine translation, attention OR attention-based OR attentional

   A. All time
   B. Since 2014

Sch2. "neural machine translation", attention OR attention-based OR attentional

   A. All time
   B. Since 2014

Sch3. "neural machine translation attention" (All time)

Sch4. "neural machine translation"

   A. All time
   B. Since 2014

Search Sch1A retrieved 206,000 results for all time and search Sch1B (since 2014) 17,400 results. The first 50 results were the same for both searches, resulting in 50 duplicates alone, which is why search Sch1A was discarded completely. Sch2 and Sch3 were attempts to further narrow down the search results.

Sch2A (all time) had 15,500 results and Sch2B (since 2014) had 11,800 results. Since the top 51 results were the same for Sch2A and B, Sch2A was discarded. In Sch2B, most of the top 51 results were the same as Sch1A and Sch1B, but not all, so Sch2B was kept.

Sch3 on the other hand was very narrow, resulting in only 47 results, of which only 19 remained after discarding based on title and abstract. It is also notable that Sch3 did not intersect that much with the results from Sch1 or Sch2.

Sch4 had many of the same hits as previous searches, but also a few that were related to the topic and which previous searches did not discover, at least not among top results. Sch4A returned 23,000 hits, while Sch4B returned 15,000 hits. The top results for Sch4A and B were the same, so Sch4A was discarded.

In some previous literature review theses within the same major study subject in University of Jyväskylä, such as that of Peuron (2017), Mononen (2018), and Haapanen (2018), the search strings were refined until the number of results was reasonable. The range was 145–884 results in the aforementioned theses. In the present study, despite the efforts to narrow down the searches on Google Scholar, most searches still returned over 10,000 hits, whereas the narrowest returned only 47 and found very few relevant articles. Therefore, for each Google Scholar search included, results after the first 50–60 hits were discarded. This was necessary to keep the scope of the work reasonable.

The following search string were used in Web of Science:

WoS1. neural machine translation attention
WoS2. neural machine translation attention-based
WoS3. neural machine translation attentional

All in all, the Web of Science searches returned few articles. WoS1 returned 42 hits, WoS2 returned eight hits, and WoS3 returned only two hits. All of the articles discovered by WoS2 were also discovered by WoS1, which is why WoS2 was discarded. WoS1 found only one of the two articles that WoS3 discovered.

Table 1 sums up search string results in numbers.

4.2.2 Narrowing down to top results and removing duplicates

The selected search strings returned over 44,000 results in total. As stated earlier, only the first 50–60 results for searches that return a large number of hits were considered for analysis. For Sch1B, this meant narrowing over 17,400 hits down to 51, for Sch2B 11,800 hits to 51, and for Sch4B 15,000 hits to 50. Other selected searches were narrow enough to process as is. After this process, there were 199 candidate articles found through Google Scholar and 44 articles through Web of Science, in other words, 243 candidate articles in total.

Next, duplicates were removed. There were 81 duplicates between the six included searches. After filtering the duplicates from amongst the 244 candidate articles, there were 162 articles left for scrutiny based on preconditions and content.

Table 1. Search results in numbers

Search	Database	Search string	Hits	Included
Sch1A	Google Scholar	neural machine translation, attention OR attention-based OR attentional	ca. 206,000	No
Sch1B	Google Scholar	neural machine translation, attention OR attention-based OR attentional	ca. 17,400	Yes
Sch2A	Google Scholar	“neural machine translation”, attention OR attention-based OR attentional	ca. 15,500	No
Sch2B	Google Scholar	“neural machine translation”, attention OR attention-based OR attentional	ca. 11,800	Yes
Sch3	Google Scholar	“neural machine translation attention”	47	Yes
Sch4A	Google Scholar	“neural machine translation”	ca. 23,000	No
Sch4B	Google Scholar	“neural machine translation”	ca. 15,000	Yes
WoS1	Web of Science	neural machine translation attention	42	Yes
WoS2	Web of Science	neural machine translation attention-based	8	No
WoS3	Web of Science	neural machine translation attentional	2	Yes
Total			ca. 288,800	ca. 44,292

4.2.3 Scrutiny based on preconditions

In the present study, there were some inclusion and exclusion criteria that qualify as preconditions before considering the article for review based on topical information (abstract and keywords). The most common were that the text was not an article (it was e.g. a Powerpoint presentation), the article was not in English, and that the full text was not available, at least not with JYU student credentials.

The most common preconditions that resulted in discarding the article was that the text was not a scientific paper (five texts) or that the full text was not available for free (five texts). Other reasons were that the article was not in English (three articles). One article was discarded at this stage because it had already been read and it was known that its focus was on neural machine translation challenges (Koehn and Knowles 2017). There were altogether 12 articles discarded from Google Scholar and two from Web of Science based on preconditions.

4.2.4 Scrutiny based on title and abstract

Finally, the article titles and abstracts were investigated based on exclusion and inclusion criteria. Naturally, the most common reason for exclusion was that the article was not related to the topic. This included, for example, not being related to translation or describing a non-attentional model. Other reasons include that English was not one of the languages that was studied in the article or that the articles turned out to be technical reports of some specific translation tool. Altogether 40 articles were discarded based on title and 16 based on abstract. In effect, the final number of articles considered for review was 92. Table 2 sums up the entire candidate article search process in numbers.

After the initial screening process, there were 92 candidate papers left. For a sole master's thesis researcher, this was still a large number. Kitchenham, Budgen, and Brereton (2016) offer some solutions for dealing with a large number of papers. One is having the work divided to more people, which is not possible in the present study. The other two are revising the research questions and basing selection on a random sample of studies. Seeing how the study is quantitative and the number of articles was already narrowed down, it is justifiable

Table 2. Papers found at different stages (adapted from Kitchenham, Budgen, and Brereton 2016)

	Google Scholar	Web of Science
Search strings	ca. 44,220	44
After filtering out results after first 50–60 results	199	44
After discarding duplicates	122	40
After discarding based on some precondition	110	38
After discarding on basis of title	90	18
After discarding on basis of abstract and/or keywords	78	14

to make the analysis on a random sample.

A random permutation of all 92 articles was made to determine which articles will be selected for analysis. The random permutation was made with Python's random library using the *shuffle* function. The function is based on the Mersenne Twister random number generator (Python Software Foundation 2020). The function was used on a list of article names (a list of strings) with no user-provided seed, meaning that the seed was the default seed, the current time.

The initial sample was a third of all articles, meaning that 31 articles were selected for analysis, i.e., for the final selection round. Some of these selected articles turned out to fill out exclusion criteria, so they were excluded from analysis. The final set of articles and the analysis of selected articles is presented in the following chapter.

5 Literature mapping on attention-based neural machine translation

In this chapter, results of the systematic literature review will be presented. The data was collected on topics relevant to the research questions.

First, I will list and describe the papers that were randomly selected for review, dividing them into tables according to whether they were included in the final review or not. Then, I will go through the most important features of network architectures present in the articles to answer research question RQ2. *What are the features of attention-based neural machine translation models*. In Section 5.4, I will go through the language directions and training and test data used in the papers. In Section 5.5, I will present how well the models performed in translation tasks, according to the results reported by authors themselves. This data is essential to answer my third and fourth question, RQ3. *How well do attention-based NMT models perform in translation tasks?* and RQ4. *How well does attention-based NMT perform in translation tasks involving low-resource languages?*.

5.1 Articles selected for analysis

This section sums up the articles selected for analysis. The papers included in analysis are listed in Table 3. Papers that were included in the random sample, but upon closer inspection filled some exclusion criteria and were discarded are listed in Table 4. Table also includes reason for discarding. Table 5 sums up the article types.

Table 3 sums up the selected papers. There were a total of 24 papers selected for the review. As can be seen from the table, the publication dates range from late 2015 to late 2019. There are altogether two papers from 2015, 12 papers from 2016, five from 2017, two from 2018, and three from 2019. The abundance of older articles is most likely caused by the search engines' tendency to rank most cited work high in the results: the older the publication, the more likely it has numerous references to it.

Table 3: Papers included in the analysis

ID	Title	Author(s)	Published
P1	Coverage embedding models for neural machine translation	Mi et al.	Nov 2016
P2	Incorporating Source-Side Phrase Structures into Neural Machine Translation	Eriguchi, Hashimoto, and Tsuruoka	Jun 2019
P3	Multi-way, multilingual neural machine translation	Firat et al.	Sep 2017
P4	Achieving open vocabulary neural machine translation with hybrid word-character models	Luong and Manning	Aug 2016
P5	Montreal neural machine translation systems for WMT'15	Jean et al.	Sep 2015
P6	Attention is all you need	Vaswani et al.	Jun 2017
P8	Promoting the knowledge of source syntax in Transformer NMT is not needed	Pham, Machácek, and Bojar	Oct 2019
P9	A hierarchy-to-Sequence Attentional Neural Machine Translation Model	Su et al.	Mar 2018
P11	Multi-source Neural Translation	Zoph and Knight	Jun 2016
P13	Improved Neural Machine Translation with SMT Features	He et al.	Feb 2016
P16	Incorporating Discrete Translation Lexicons into Neural Machine Translation	Arthur, Neubig, and Nakamura	Nov 2016
P18	Massive Exploration of Neural Machine Translation Architectures	Britz et al.	Sep 2017

Table 3: Papers included in the analysis

ID	Title	Author(s)	Published
P19	Agreement-based Joint Training for Bidirectional Attention-based Neural Machine Translation	Cheng et al.	Jul 2016
P20	Neural machine translation by jointly learning to align and translate	Bahdanau, Cho, and Bengio	May 2015
P21	Character-based Neural Machine Translation	Costa-jussà and Fonollosa	Aug 2016
P22	Google's Multilingual Neural Machine Translation System: Enabling Zero-Shot Translation	Johnson et al. (Google)	Oct 2017
P23	Fine-grained attention mechanism for neural machine translation	Choi, Cho, and Bengio	Jan 2018
P24	Transfer Learning for Low-Resource Neural Machine Translation	Zoph et al.	Nov 2016
P25	Exploiting Source-side Monolingual Data in Neural Machine Translation	Zhang and Zong	Nov 2016
P26	Context Gates for Neural Machine Translation	Tu et al.	Mar 2017
P27	Neural Machine Translation with Supervised Attention	Liu et al.	Dec 2016
P28	Controlling politeness in Neural Machine Translation via Side Constraints	Sennrich, Haddow, and Birch	Jun 2016

Table 3: Papers included in the analysis

ID	Title	Author(s)	Published
P29	Persistent hidden states and nonlinear transformation for long short-term memory	Choi	Feb 2019
P30	Supervised Attentions for Neural Machine Translation	Mi, Wang, and Ittycheriah	Nov 2016

Table 4 reveals that two papers (P17 and P31) were discarded based on not being published as part of any publication (journal, conference proceedings or workshop proceedings). P31 also had multimodal research data, not only text data, which qualified for discarding in this study. P14 was found not to discuss translation at all, while P15 was not focused on translation especially. In P7, the language pairs discussed did not involve English (this was not evident from the abstract). P10 was discarded because the proceedings of the conference it was part of had not yet been published at the time of writing. P12 was a compilation of previously published articles, and none of them was found to be exactly on the topic of the present study.

Table 4: Papers discarded from the analysis

ID	Title	Reason for discarding
P7	Coverage for Character Based Neural Machine Translation	One of the language pairs was not English
P10	Dynamic Fusion: Attentional Language Model for Neural Machine Translation	Submitted to PACLING2019, but proceedings not published yet
P12	Hybrid Machine Translation by Combining Output from Multiple Machine Translation Systems	Article is a compilation of previously published papers

Table 4: Papers discarded from the analysis

ID	Title	Reason for discarding
P14	Effective Attention-based Neural Architectures for Sentence Compression with Bidirectional Long Short-Term Memory	Not related to translation
P15	What do Neural Machine Translation Models Learn about Morphology?	Focus of the article was not translation
P17	Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation	Not published as part of conference or in a journal
P31	Doubly Attentive Transformer Machine Translation	Not published as part of conference or in a journal + multimodal data

Table 5 shows the papers by publication type. The most common type of publication was a conference paper published as part of conference proceedings, with a total of 15 out of 24 papers. The second most common type was an article in a journal. One paper, P5, was published in workshop proceedings.

Table 5. Papers included in analysis by publication type

Publication type	Articles (id)	Total
Conference proceedings	P1, P4, P6, P11, P13, P16, P18, P19, P20, P21, P24, P25, P27, P28, P30	15
Article in a journal	P2, P3, P8, P9, P22, P23, P26, P29	8
Workshop proceedings	P5	1

5.2 Quality assessment

One part of the process outlined by Kitchenham, Budgen, and Brereton (2016) is quality assessment for the reviewed papers. Given the expected scope of a master's thesis, an ex-

tensive quality assessment was not done for this study. However, a short quality checklist was applied to the reviewed papers. The checklist is as follows, adapted from (Kitchenham, Budgen, and Brereton 2016, 83):

1. Is a clear chain of evidence established from observations to conclusions?
2. Is raw data available, e.g. in the form of source code?

All reviewed papers were found to fulfill the first criterion. All papers had a clear structure, and all arguments were supported with transparent and clearly presented results. Providing links to source code on the other hand was fairly rare. Altogether eight authors provided links to their source code while the rest did not. One of the provided links had expired by the time of writing. Availability of source code per paper is summed up in Table 6. In this study, the location of the source code needed to be explicitly mentioned in the paper to count as available.

Table 6. Availability of source code.

Source code availability	Present in articles (by id)	Total
Provided	P3, P4, P6, P11, P18, P20, P26	7
Provided, but link expired	P16	1
Not provided	P1, P2, P5, P8, P9, P13, P19, P21, P22, P23, P24, P25, P27, P28, P29, P30	16

5.3 Neural network architectures

There were multiple neural network architectures used. The most common by a clear margin was recurrent neural networks (RNN). The reason for this, as was indicated by some authors, was that the original attentional network was an RNN. An impressive competitor to RNN was the purely attentional architecture called Transformer, which was also present in two papers.

Table 7 sums up the used architectures. 22 out of 24 papers featured a model with an RNN architecture. Two papers, P6 and P8, discussed the Transformer architecture. The Transformer, in short, is a non-recurrent attentional model (see e.g. Vaswani et al. 2017, the paper marked P6, for more information).

Table 7. Neural network architectures

Architecture	Present in articles (by id)	Total
Recurrent	P1, P2, P3, P4, P5, P9, P11, P13, P16, P18, P19, P20, P21, P22, P23, P24, P25, P26, P27, P28, P29, P30	22
Transformer	P6, P8	2

5.3.1 Learning methods

Learning method is a central feature of a neural network as it determines how fast and how well the model learns. Across the reviewed literature, the traditional SGD was a popular learning method for NMT, but alongside there were optimisations such as Adam and Adadelta. The learning methods used are summed up in Table 8.

Sometimes the learning method was not explicitly mentioned, but the authors would state that they based their network on some specific network, meaning that the learning method was probably the same. This was the case in P21, P22, P24, and P26. However, if the author did not explicitly name the learning method, it appears in Table 8 as "not mentioned", some with a footnote on what it might be.

Table 8. Learning methods. Note: some articles appear multiple times in the table because authors used more than one learning method.

Learning method	Present in articles (by id)	Total
SGD	P2, P3, P4, P13, P18, P19, P20, P28	8
Adam	P3, P6, P16, P18, P23, P29	6
Adadelta	P1, P5, P13, P19, P20, P27, P30	7
RMSprop	P9	1
AdaGrad	P25	1
Not mentioned	P8, P11, P21[1], P22[2], P24[3], P26[1]	6

1. Based on GroundHog/P20, so SGD and Adadelta were probably used.
2. Based on P17, so SGD and Adam were probably used.
3. Mentions being based on Luong, Pham, and Manning (2015), so most probably uses SGD.

Table 8 shows that SGD and Adadelta were used in the models in eight papers and Adam was used in six papers. The model in P9 used RMSprop and the model in P25 used AdaGrad. For two papers, the learning method was not mentioned and for four the used method was implicit.

5.3.2 Activation functions

Table 9 sums up the activation functions used throughout the body of literature. As can be seen from the table, softmax was the most popular activation function by a large margin: it was used in 14 models. Hyperbolic tangent (tanh) and maxout were both used in four models. As the sole example, P6 used ReLu. In four papers, the used activation function was not mentioned, although P28 was based on a variant of GroundHog (P20), meaning that it may have used softmax and maxout.

Table 9. Activation functions. Note: some articles appear multiple times in the table because authors used more than one activation function.

Activation function	Present in articles (by id)	Total
softmax	P1, P2, P4, P5, P11, P16, P18, P20, P21, P22, P24, P26, P27, P30	14
tanh	P3, P13, P26, P29	4
maxout	P5, P9, P19, P20	4
ReLu	P6	1
Not mentioned	P8, P23, P25, P28[1]	3

5.3.3 Computational units

The authors often mention what types of computation unit their network model employ on the hidden layers. The hidden units used were mainly Long Short-Term Memory (LSTM) and gated recurrent unit (GRU), as was expected in the natural language processing context. A few variants of these appeared as well. Table 10 sums up the types of hidden units used across models.

1. Based on GroundHog/P20, so softmax and maxout were probably used.

Table 10 reveals that GRU was the most common hidden unit, used in 13 models. LSTM was close behind with eight usages. Two variants of the aforementioned hidden units appeared as well: P2 used Tree-LSTM and P29 used PRU, both of which are variants of LSTM.

Table 10. Use of hidden units

Type of hidden unit	Present in articles (by id)	Total
GRU	P1, P3, P5, P9, P13[1], P19[1], P20, P21[1], P25[1], P26 P27, P28, P30	13
LSTM	P2, P4, P11, P16, P18, P22, P23, P24	8
Tree-LSTM	P2	1
PRU[2]	P29	1
Not mentioned / custom	P6, P8	2

5.4 Languages, translation directions and text data

Authors used various language pairs and translation directions for testing their translation models. One criteriion for my article selection was that one of the languages must be English, hence English is always one of the languages in a given translation direction. Table 11 sums up the translation directions present in the reviewed articles. The language pair total in the table exceeds the number of articles reviewed, because many articles included more than one language pair.

The most common language pair and translation direction was English-to-German (marked in Table 11 simply as English – German), followed closely by Chinese-to-English. Third most common were both English-to-French and German-to-English, which were both featured in five papers (counting P11 for the latter direction). P3 and P22 focused especially on multilingual translation, which is why they included so many language pairs. P3 included all the language pairs in the WMT15 translation task as well as two low-resource language pairs: Uzbek-to-English and Turkish-to-English. P3 also specifically had a multi-way model: there were multiple source and target languages simultaneously. P22 also included English

1. Implicit; work based on GroundHog (P20), which uses GRU.
2. PRU = persistent recurrent unit, a variant of LSTM (see Choi 2019, for more details)

↔ Korean (notation ↔ means both directions, English-to-Korean and Korean-to-English), English ↔ Spanish, English ↔ Portuguese, English ↔ Ukranian, and English ↔ Belarusian. These pairs were not included in this review, because there was no other paper with which to compare results for these languages. In addition to the languages listed here, P24 also contained Hansa-to-English and Urdu-to-English direction, but these were not included in review for the same reason as the aforementioned languages in P22. P11 had a similar multisource approach as P3: it used two sources at a time (French+English, marked as 'Fr' and 'En', and German + French, marked as 'De' and 'Fr') to translate to one target language.

Table 11. Languages and translation directions

Translation direction	Present in articles (by ID)	Total
English – German	P3, P5, P6, P9, P18, P21, P22, P23, P28, P29	10
Chinese – English	P1, P9, P13, P19, P25, P26, P27, P30	8
English – French	P3, P6, P19, P20, P22	5
German – English	P3, P5, P21, P22	4
French – English	P3, P19, P22	3
English – Czech	P3, P4, P5	3
Czech – English	P3, P5, P8	3
English – Finnish	P3, P23, P29	3
Finnish – English	P3, P5, P29	3
English – Japanese	P2, P16, P22	3
English – Russian	P3, P22	2
Russian – English	P3, P22	2
Turkish – English	P3, P24	2
Uzbek – English	P3, P24	2
English – Chinese	P19	1
multisource(Fr+En) – German	P11	1
multisource(De+Fr) – English	P11	1

The authors also used multiple parallel corpora sources in their experiments. Table 12 sums up the text datasets used for training the network and Table 13 sums up the datasets used for testing in the reviewed articles. For the sake of brevity, different translation directions have been merged into one row (for example, English – German – English, English-to-German, and German-to-English are all marked as English $\leftrightarrow$ German).

Table 12 reveals that overall, the WMT dataset was most popular for training, with eight authors using WMT15 and seven using WMT14. The LDC (Linguistic Data Consortium) dataset was also popular. There were instances of use of LDC dataset from the years 2000–2005 and 2014, however, in Table 12 different LDC datasets are marked generically as 'LDC' for brevity. Numerous other datasets were also used in the reviewed papers, but the rest of the datasets were used in only a single paper, apart from DARPA BOLT which was used in two. There were two custom training datasets, one a concatenation of multiple datasets and the other automatically crawled from web.

Table 13 shows which datasets were used for testing the networks. There were multiple NIST MT datasets used, NIST MT02–MT06 and NIST MT08 (both authors using MT08 used the news and web subsets). In Table 13, all different NIST datasets are marked simply as 'NIST MT'. It should be noted that while the newstest datasets are a part of WMT collection, sometimes authors only mention WMT in general without specifying which individual set they use. In these cases, the given dataset in the table is WMT<year>. WMT stands for Workshop on Statistical Machine Translation (see e.g. EMNLP 2015), while WAT stands for Workshop on Asian Translation (see e.g. WAT 2015).

As can be seen from Table 13, all eight authors who experimented on the English-to-Chinese pair used the NIST MT dataset for testing. For testing other language pairs, most popular were different WMT datasets, with seven authors using newstest2015 and one using WMT2015 generically, and six authors using newstest2014 and three using WMT2014 generically. There were also single uses of newstest2013, WAT2015, LDC2014, KFTT, BTEC, OpenSubtitles2013, and Google's production datasets. Peculiar enough, P24 had no mention of which dataset was used for testing, but implicitly it might mean that a sample of the training dataset, WMT2015, was used.

Table 12. Training datasets

Training data	Translation direction	Present in articles (by ID)	Total by direction	Data total
WMT2015	English ↔ German	P3, P5, P18, P19, P23, P29	6	8
	English ↔ Czech	P3, P4, P5	3	
	English ↔ Finnish	P3, P23, P29	3	
	English ↔ French	P3	1	
	English ↔ Russian	P3	1	
	Turkish – English	P24	1	
	Uzbek – English	P24	1	
WMT2014	English – French	P17, P19, P20, P22	4	7
	English – German	P5, P9, P22	3	
	English – Japanese	P22	1	
	English – Russian	P22	1	
	multisource(Fr+En) – German	P11	1	
	multisource(Fr+De) – English	P11	1	
LDC	English ↔ Chinese	P9, P19, P25, P26	4	5
	Uzbek – English	P3	1	
DARPA BOLT	Chinese – English	P1, P30	2	2
WAT2015	English – Japanese	P2[1]	1	1
CzEng 1.7	Czech – English	P8	1	1
NIST2008	Chinese – English	P27	1	1
KFTT	English – Japanese	P16	1	1
BTEC	English – Japanese	P16	1	1
OpenSubtitles-2012	English – German	P28	1	1
Custom[2]	Turkish – English	P3	1	1
Custom[3]	Chinese – English	P13	1	1

1. The authors solely mentioned this dataset, so it is unsure if it was used both for training and testing
2. Custom concatenation of LDC2014E115, WIT TED Talks, SETimes2, OpenSubtitles, Tatoeba Corpus
3. Automatically crawled from the web

Table 13. Test datasets

Test data	Translation direction	Present in articles (by ID)	Total by direction	Data total
NIST MT	English ↔ Chinese	P1, P9, P13, P19, P25, P26, P27, P30	8	8
newstest2015	English ↔ German	P3, P5, P18, P22, P23, P29	6	7
	English ↔ Finnish	P3, P5, P23, P29	4	
	English ↔ Czech	P3, P4, P5	3	
	English ↔ French	P3	1	
	English ↔ Russian	P3	1	
newstest2014	English ↔ German	P5, P9, P18, P22	4	6
	English ↔ French	P19, P20, P22	3	
WMT2014	English – French	P6, P20	2	3
	English – German	P6	1	
	multisource(Fr+En) – German	P11	1	
	multisource(Fr+De) – English	P11	1	
WMT2015	English ↔ German	P21	1	1
newstest2013	English – German	P23	1	1
WAT2015	English – Japanese	P2	1	1
LDC2014	Turkish – English	P3	1	1
	Uzbek - English	P3	1	
KFTT	English – Japanese	P16	1	1
BTEC	English – Japanese	P16	1	1
OpenSubtitles-2013	English – German	P28	1	1
Google's production datasets	multiple	P22	1	1
Not mentioned	Turkish – English	P24	1	
	Uzbek – English	P24	1	

Comparing Table 12 and Table 13, one can notice that sometimes authors used the same dataset for both training and testing. However, authors always clarified how the data was

split to ensure that the same data was not used for both training and testing. This is important with neural networks to ensure the reliability and validity of the results.

5.5 Translation quality

In this section, I will go through the translation quality results from the reviewed papers. All authors used BLEU scores to measure how well their models performed in translation tasks. Some authors also included human evaluation of translation quality. Other metrics apart from BLEU were also used, some notable examples including TER-BLEU, PPL, NIST, and chrF. Authors of P1 and P30 also reported BP (brevity penalty), which is a constituent of BLEU. A summary of use of metrics overall is presented in Table 14.

Table 14. Use of different metrics as a measure of translation quality

Use of BLEU score	Article (id)	Total
BLEU	ALL	24
TER-BLEU (TB)	P1, P3, P30	3
PPL	P11, P24	2
NIST	P16, P26	2
chrF	P4, P8	2
RIBES	P2	1
LL	P3	1
CharacTER and BEER	P8	1

Comparing BLEU scores is a handy way to assess translation quality, however, the use of other metrics was so marginal that no sensible comparison could be made for them in the present study. Hence, other metrics will not be discussed. I will first go through the BLEU scores and then briefly discuss human evaluation results.

5.5.1 BLEU scores for high-resource languages

All papers that were reviewed had authors use BLEU scores to measure how well their models performed in translation tasks. This goes to show how definitive this metric is considered

in NMT research.

Authors also put a lot of emphasis on the BLEU scores they achieved. The scores were often mentioned already in the abstract, especially in relation to previous studies, e.g. Vaswani et al. (2017) stated in the abstract of P6 that "our model achieves 28.4 BLEU – improving over the existing best results – by over 2 BLEU". Authors would also use BLEU scores as to enforce the arguments they made. Choi, Cho, and Bengio (2018) (P23) state that an improvement of +1.4 BLEU "clearly confirms the importance of treating each dimension of the context vector separately". Zoph and Knight (2016) (P11) used a model that utilised the same source sentence from two different languages to disambiguate between synonyms in the translation, and they state that less gains in BLEU with English and French as source languages rather than French and German act as evidence that more distinct source languages are better at disambiguating each other.

It is hard to compare BLEU scores between papers that had not only different language pairs but also different data used. For this review, the results are tabulated per language pair and also translation direction, for example, English-to-German and German-to-English have their own tables. The tables also include a column for the used test dataset. Some papers appear in multiple tables, because the authors tested their model with more than one language pair. Papers in the tables are also always presented in chronological order, the oldest one at the top and the most recent one at the bottom, to see if score development happens over time. The best score for the translation direction is marked in bold. For each paper and translation direction, only one BLEU score, the best result, was considered.

First, I will go through high-resource European languages, then high-resource Asian languages. In Section 5.5.2, I will go through BLEU scores for all low-resource languages present in the data.

European languages

European high-resource languages and translation directions present in this study were English <-> German, English <-> French, English <-> Czech, and English <-> Russian.

Table 15 reveals that the best score for English-to-German was 28.4 BLEU in P6. It is an

interesting result for many reasons. First, the network model in P6 was the self-attentional, non-recurrent Transformer, as opposed to the traditional attentional recurrent neural network model. Second, it was published chronologically mid-way for this translation direction, dating back to June 2017. P6 also used Adam as learning method and ReLu for activation. As a Transformer model, P6 did not have an LSTM or a GRU hidden unit. The second best score was 26.43 in P22, published in October 2017. P22 had an attentional recurrent neural network, with LSTM hidden unit, SGD and Adam as learning methods, and softmax for activation. The specialty of the model in P22 was that the translator was multilingual and uses multiple source languages at a time. In this case, however, the single language pair model performed better. Third best score, 25.23, was in P18, which was also published in 2017. P18 was an exploration of different network models, so its model was carefully optimised. P18 featured a bidirectional, four-layer RNN encoder and a four-layer attentional RNN decoder. The average score for all papers on English-to-German was 23.42 BLEU.

Table 15. BLEU scores of English–German translation

Language-pair	Article (id)	BLEU (top result)	Test data	Published
English – German	P5	24.8	WMT2015[1]	2015
	P11[2]	18.6	WMT2014	6/2016
	P28	24.0	OpenSubtitles2013	6/2016
	P21	20.22	WMT2015	8/2016
	P6	**28.4**	WMT2014	6/2017
	P3	21.75	newstest2015	9/2017
	P18	25.23	newstest2015	9/2017
	P22	26.43 (single)/ 24.01 (multi)	newstest2014	10/2017
	P23	23.74	newstest2013	1/2018
	P9	20.93	newstest2014 and 2015	3/2018
	P29	22.98	newstest2015	2019

1. It is unsure if the authors used a subset of WMT2015; they mentioned several subsets.
2. Multisource, source languages were English and French.

Table 16 shows that the direction German-to-English retrieved similar results to the opposite direction. P22, which had the second best model for the other direction, achieved best BLEU score for German-to-English translation, 28.4, with its multilingual model, but also the second best score, 31.77, with its single model. It is notable that P6, the best model for English-to-German, did not feature this other direction. The best model is once again from the year 2017, however, it is notable that the body of literature did not feature any model past the year 2017 for this translation direction. The third best model for this translation direction was in P11, with a BLEU score of 30.0. P11 also had a multisource model, with German and French as source languages in this case. The average score for all papers on German-to-English was 28.0 BLEU.

Table 16. BLEU scores of German–English translation

Language-pair	Article (id)	BLEU (top result)	Test data	Published
German – English	P5	27.6	newstest2014	2015
	P11[1]	30.0	WMT2014	6/2016
	P21	22.10	WMT2015	8/2016
	P3	24.20	newstest2015	9/2017
	P22	31.77 (single) / **32.32** (multi)[2]	newstest2015	10/2017

The best BLEU score for English-to-French translation was 41.0 in P6, as can be seen from Table 17. This means that P6 achieved the best result for both translation directions it featured (En-De and En-Fr). The second best score, 38.16, was achieved with the multilingual model from P22. The original paper presenting the attention mechanism, P20, had the third best BLEU score, 36.15, for English-to-French. The average for English-to-French translation results was 35.64 BLEU.

Table 18 shows the BLEU scores for French-to-English translation direction. For French-to-English, the best BLEU score was 36.47 with the single model P22, and second best

1. Multisource, source languages were German and French.
2. “Single” stands for single language pair, “Multi” stands for multilingual model.

Table 17. BLEU scores of English–French translation

Language-pair	Article (id)	BLEU (top result)	Test data	Published
English – French	P20	36.15	WMT 2014	2014
	P19	33.45	newstest2014	2016
	P6	**41.0**	WMT 2014	6/2017
	P3	29.7	newstest2015	9/2017
	P22	35.37 (single) / 38.16 (multi)	newstest2014	10/2017

score, 35.93, was achieved with the multilingual model from P22. Third best score for this translation direction was 31.51 in P19. The model in P19 featured a bidirectional RNN encoder and an attentional RNN decoder. The specialty of the model in P19 is that source-to-target and target-to-source models are trained to prefer agreeing on alignment matrices. The average BLEU score for French-to-English was 32.39 BLEU.

Table 18. BLEU scores of French–English translation

Language-pair	Article (id)	BLEU (top result)	Test data	Published
French – English	P11[1]	30.0	WMT 2014	6/2016
	P19	31.51	newstest2014	7/2016
	P3	28.06	newstest2015	9/2017
	P22	**36.47** (single) / 35.93 (multi)	newstest2014	10/2017

Table 19 shows the scores for English-to-Czech translation. The best BLEU score, 20.7, was achieved in P4. The model in P4 is deep RNN encoder-decoder. Its specialty is that its model was a hybrid of two models: translation was usually made at word level, while rare words were inspected at character level. The second best model in P5 scored 18.3 BLEU and the third one in P3 scored only 13.84. Both P5 and P3 used an RNN encoder-decoder architecture, however, P3 utilised decoders for each target language individually. The average for this translation direction was 17.61 BLEU.

1. Multisource, source languages were French and German.

Table 19. BLEU scores of English–Czech translation

Language-pair	Article (id)	BLEU (top result)	Test data	Published
English – Czech	P5	18.3	newstest2015/WMT2015	2015
	P4	**20.7**	newstest2015	2016
	P3	13.84	newstest2015	2017

For the Czech-to-English pair, the best score is 38.01 in P8, by a considerable margin to the second best of over 14 BLEU points, as can be seen from Table 20. The second best score was 23.3 in P5 and third was 20.57 in P3, similarly to the English-to-Czech direction. The model in P8 was Transformer, while P3 and P5 were RNNs. It is also notable that P8 was published in 2019 and was significantly more recent than the other papers. Furthermore, P8 uses a different dataset than P3 and P5. Judging by the BLEU score, the models in P3 and P5 performed better with this translation direction than with the opposite direction. The average for this translation direction was 27.39 BLEU, almost 10 points more than the opposite direction, however, it was raised significantly by the score in P8 alone.

Table 20. BLEU scores of Czech–English translation

Language-pair	Article (id)	BLEU (top result)	Test data	Published
Czech – English	P5	23.3	newstest2015/WMT 2015	2015
	P3	20.57	newstest2015	2017
	P8	**38.01**	CzEng	2019

The results for Czech-to-English make this the third case in this study where the Transformer model performs best. In fact, both papers with a Transformer model, P6 and P8, achieved best results in the translation directions they included (Czech-to-English in P8 and English-to-German and English-to-French in P6), which seems to suggest that Transformer models might perform better than recurrent models. However, the sample in this study is too small to confirm such a claim, but it provides grounds for future research.

Tables 21 and 22 present the BLEU score results for the English - Russian language pair, which was present in only two papers, P3 and P22. For both directions, P22 performed better

than P3. P22 scored 22.21 BLEU on the English-to-Russian direction and 28.46 BLEU on the opposite direction, while P3 scored 19.54 on the English-to-Russian direction and 23.44 BLEU on the opposite direction. Both papers were published at the end of 2017, so time does not explain the difference. Both P3 and P22 had an RNN encoder-decoder with a multilingual model. They both utilised SGD and Adam as learning method, however, P22 used LSTM hidden unit while P3 utilised GRU.

Table 21. BLEU scores of English–Russian translation

Language-pair	Article (id)	BLEU (top result)	Data	Published
English – Russian	P3	19.54	newstest2015	9/2017
	P22	**22.21**	Google's production data	10/2017

Table 22. BLEU scores of Russian – English translation

Language-pair	Article (id)	BLEU (top result)	Data	Published
Russian – English	P3	23.44	newstest2015	9/2017
	P22	**28.46**	Google's production data	10/2017

Asian languages

Asian high-resource languages and translation directions present in the reviewed papers were Chinese <-> English and English-to-Japanese. However, the English-to-Chinese direction was only present in one paper, P19, so no comparison can be made for this direction in the present study. P19 achieved a BLEU score of 21.70 for English-to-Chinese.

Table 23 shows results for Chinese-to-English translation, which was the second most popular translation direction in the reviewed body of literature. The best score for this translation direction was 41.68 BLEU in P9, which was also the most recently published paper for this direction. In P9, the network model is a bidirectional RNN encoder-decoder. Its specialty is a hierarchical encoder, which segments long sentences into smaller clauses to translate, thus alleviating the usual difficulties with long sentences common in NMT. The second best score

was 37.8 BLEU in P27. P27 employs an usual RNN encoder-decoder model, but its specialty is a supervised attention mechanism that affiliates non-aligned source words to their closest aligned words. Very close to the second best, the third best score was 37.41 BLEU in P25. P25 has a traditional RNN encoder-decoder, but as a specialty utilises source-side monolingual data, multi-task learning, and sentence reordering to improve the encoder-side of the network. The average BLEU score for Chinese-to-English translation was 36.77.

Table 23. BLEU scores of Chinese–English translation.

Language-pair	Article (id)	BLEU (top result)	Data	Published
Chinese – English	P13	32.94	NIST MT08	2/2016
	P19	35.72	NIST MT04	7/2016
	P1	36.80	NIST MT06	11/2016
	P25	37.41	NIST MT04	11/2016
	P30	36.95	NIST MT06	11/2016
	P27	37.8	NIST MT05	12/2016
	P26	34.83	NIST MT06	3/2017
	P9	**41.68**	NIST MT04	2018

Table 24 shows the BLEU scores for English-to-Japanese translation. The best score by a clear margin was 51.04 in P16. The model in P16 incorporated lexicons to prevent mistranslation of rare content words. P16 also has the best individual BLEU score for any language pair present in this study. However, it is worth noting that the authors in P16 have the system by Bahdanau, Cho, and Bengio (2015) as their baseline, and the baseline system scores 48.31 BLEU on the same dataset as well. Furthermore, the same model in P16 achieves 23.30 BLEU on the other test dataset, KFTT. The authors address this difference by stating that BTEC is easier than KFTT, has a narrower domain, less rare words, and shorter sentences. This is very clearly a case in which the test dataset matters and a better BLEU score does not correlate with better performance.

The second best model was in P2 with a BLEU score of 38.0. The model has a usual bidirectional RNN encoder and attentional RNN decoder, but its specialty is a tree-based rather than sequential encoder. The mission of the tree-based encoder is to convey more syntactical

Table 24. BLEU scores of English–Japanese translation

Language-pair	Article (id)	BLEU (top result)	Data	Published
English – Japanese	P16	**51.04**	BTEC	2016
	P22	23.66 (single) / 21.72 (multi)	Google's production data	2017
	P2	38.0	WAT2015	2019

structure information about the source. The third best model for this translation direction was the single-source model of P22 with a BLEU score of 23.66. The average for English-to-Japanese translation was 33.61 BLEU, raised by the unusually high BLEU score in P16.

5.5.2 BLEU scores for low-resource languages

As low-resource languages are a specific topic of interest in the current study, it is justified to review the BLEU scores for them separately from high-resource languages. In the end, the included body of literature had five papers that studied low-resource languages: P3, P5, P23, P24, and P29. The low-resource languages present in the reviewed papers were Finnish, Turkish, and Uzbek. There does not seem to be a specific defined line between low-resource and high-resource languages, but the categorisation in this study was simply whether one or more authors addressed the language as low-resource or not. The lack of definition also meant that datasets were of different sizes, which can also be seen from results.

There are some special factors to concern with low-resource languages. One major issue is that with small datasets the model can accidentally be overfit by including every unique target word in the vocabulary. In the reviewed articles, the authors usually compensated for the small size of corpora by having a smaller vocabulary for low-resource language pairs than for other languages. For example, in P5, the vocabulary for English-to-Finnish was 40K tokens while for other languages it was 200K–500K tokens. Similarly, in P29, the English-to-Finnish vocabulary was 10K tokens, while the English-to-German vocabulary was 30K tokens. However, in P29, the hidden unit proposed by authors had promising results for alleviating overfitting problems that are common with small data, meaning that there are

other ways to avoid overfitting.

Table 25 shows the results for English-to-Finnish. All models were tested on the WMT15, making them fairly comparable. Overall, the scores for this language pair are significantly lower than for high-resource languages. The best BLEU score was 10.20 in P23, however, it is not significantly higher than the other scores since the overall average for English-to-Finnish translation was 9.69 BLEU. The model in P23 is a usual RNN encoder-decoder with an LSTM hidden unit, with the specialty that an attention score is assigned to each individual dimension of the context vector instead of the entire vector at a time. P29, the second best model by a small margin has an RNN with a custom variant of LSTM, PRU, as its hidden unit. In P3, which scored 9.23 BLEU, there is multilingual model, with a shared RNN encoder and target-language-specific decoders and attention mechanisms.

Table 25. BLEU scores of English–Finnish translation

Language-pair	Article (id)	BLEU (top result)	Test data	Published
English – Finnish	P3	9.23	newstest2015	2017
	P23	**10.20**	newsdev2015 and newstest2015	2018
	P29	9.64	newstest2015	2019

Finnish-to-English translation retrieved slightly different results than the opposite direction, as can be seen from Table 26. P5, which did not include the opposite direction, scored best for Finnish-to-English translation with 13.6 BLEU. It featured a traditional attentional RNN encoder-decoder with GRU hidden unit, but also included a monolingual corpus for Finnish-to-English translation. Experiments in P3 achieved a score 12.61 BLEU for this direction, which is significantly better than its score for the opposite direction, 9.23. P29 also scored better with this direction (12.26 as opposed to 9.64 for opposite direction), but had the lowest overall score for this direction, if only by a small margin to the second best. Overall, the average for Finnish-to-English translation was 12.82 BLEU.

Two papers, P3 and P24, featured Turkish-to-English translation, as can be seen from Table 27. P3 scored better with a BLEU of 20.9 with its multilingual model, but P24 scored an

Table 26. BLEU scores of Finnish–English translation

Language-pair	Article (id)	BLEU (top result)	Test data	Published
Finnish – English	P5	**13.6**	newstest2015	2015
	P3	12.61	newstest2015	2017
	P29	12.26	newstest2015	2019

equally impressive 18.7 BLEU. The model in P24 utilised transfer learning, i.e., training a model with a high-resource parent model first and then transfering learned parameters in training the low-resource language pair model.

Table 27. BLEU scores of Turkish–English translation

Language-pair	Article (id)	BLEU (top result)	Test data	Published
Turkish – English	P24	18.7	-	2016
	P3	**20.9**	LDC2014E115	2017

The average for Turkish-to-English translation was 19.8, which is significantly higher than Finnish-to-English with its average of 12.82. Although different language pairs are not directly comparable, this raises some questions. It is hard to pinpoint why this difference is so large, especially when all models present were based on an attentional RNN encoder-decoder architecture. From a linguistic point of view, the difference can be explained simply by the differences between the languages. On another note, in P24, the largest factor is the specialty of the model: the score for the presented model is significantly improved from the baseline system (no parent model) that scored only 11.4 for Turkish-English and 10.7 for Uzbek-English. The reason may also be different training and test datasets, as this was established as a cause for significant score differences with the English-to-Japanese pair.

Same two papers that included Turkish-to-English translation, P3 and P24, featured Uzbek-to-English translation as well. The BLEU scores for this language direction are in Table 28. For this direction, P24 scored better with a BLEU of 16.8. The model in P3 scored over 4 points lower with 12.33 BLEU. The contrast between the score for Turkish-to-English and Uzbek-to-English in P3 may in part be explained by the fact that the model for the former

was trained with 10 times larger corpora than the latter (784.65K for En-Tr as opposed to 73.66K for En-Uz). The average for Uzbek-to-English pair was 14.57 BLEU.

Table 28. BLEU scores of Uzbek–English translation

Language-pair	Article (id)	BLEU (top result)	Test data	Published
Uzbek – English	P24	**16.8**	-	2016
	P3	12.33	LDC2014E115	2017

5.5.3 Qualitative evaluation of translation quality

Some authors also included some human-perceived insights on translation quality. Table 29 summarises which methods of qualitative human evaluation were reported in articles. It is worth noting that even if authors did not explicitly mention that human evaluation was involved, they might have had human evaluators as part of the process.

Table 29. Involvement of human evaluation

Human evaluation	Articles (by ID)	Total
Translation samples given and analysed by authors	P2, P3, P4, P9, P13, P16, P19, P20, P21, P22, P26, P27, P28	13
Alignment matrix analysis	P1, P3, P16, P19, P20, P23, P26, P27	8
Qualitative numeric metric used	P5, P26	2
Translation samples given, but not analysed	P11	1
Not mentioned	P6, P8, P18, P24, P25, P29, P30	7

In some cases, qualitative evaluation was given significant attention. For example, in P2 the authors provided and discussed word alignments for some example sentences from their data. Some authors presented alignment matrices of example sentences and analysed them, for example, in P3 authors note that alignments were language-pair-independent, meaning a single attention mechanism can be shared across multiple language pairs. In P5 and P26, a human ranking was given for produced translations. In both cases, it was a custom ranking,

for example, in P26 two evaluators compared the baseline system and the proposed systems and ranked whether the proposed system was worse, equal, or better in adequacy and fluency. This result was presented as percentages. In P11, some translation samples were provided for the reader's interest but not analysed in the text.

6 Discussion

I will now sum up the contents of the reviewed papers and the most relevant findings in light of the research questions. First, there is an overview of current research, then a summary of the features of network models in the reviewed papers, and finally a look at how the reviewed models performed in translation tasks.

6.1 Overview

I will now provide an overview of current research as well as the reviewed articles. This overview also provides the answer to research question RQ1. *How actively are papers on attention-based NMT published?*.

Attention-based neural machine translation is currently a very popular topic, judging by the database searches conducted in this study. There were circa 288,800 results with all the candidate search strings, with the broadest single search string retrieving over 200,000 hits on Google Scholar. The number of hits grew year by year, meaning that there is a growing trend in research on this field.

Due to the overwhelming number of search results, the first challenge in the present study was finding a suitable set of papers to review. Refining search strings did not narrow down the search results sufficiently to produce a set of candidate papers from results of a single search. The narrowest searches did produce 2 to 47 hits, but the papers were mostly unrelated to topic. For these reasons, the results of multiple search strings were considered to provide a specific enough yet also diverse body of literature. After applying the exclusion criteria, the set of candidate papers was 92 papers, which in the end was narrowed down to a random sample of 31 papers. The final set of candidate papers was small considering the number of papers published on the topic, but the strategies to narrow down aimed at providing a comprehensive crosscut of literature.

The final set of papers was 24 papers, consisting of papers published as part of conference proceedings, articles in journals, and one workshop proceedings paper. Among the reviewed

papers there were 15 conference papers and eight articles in peer-reviewed journals. All papers reviewed were found to be of good quality, with clear results and consistent arguments supported by data. Selected papers were peer-reviewed. Some authors also provided a link to publicly available source code, but it was more common to not provide a link.

Comparing the reviewed articles was significantly easier and more reasonable due to the use of same or similar datasets. The most popular training and test datasets were WMT2014 and WMT2015 and their subsets. newstest2015, a subset of WMT2015 was especially popular for testing European languages, while for the second most popular language pair, English-to-Chinese, the most popular test dataset was NIST. The authors had also made it explicit that whenever they used the same dataset for both training and testing, it was split so that the training data was not the same as test data, which is an important factor for validity.

6.2 Details of attention-based NMT architectures

I will now answer my second research question, RQ2. *What are the features of attention-based neural machine translation models?* by going through the features of the NMT models present in the body of literature reviewed.

In the reviewed papers, there were two types of architectures present: recurrent neural networks (RNNs) and Transformer networks. RNNs were the most common type of architecture. This result would suggest that RNNs have become the baseline architecture for attentional NMT. This is not a surprising result, since the original attentional system had an RNN architecture as well. The Transformer, on the other hand, is a newer architecture and has had such promising results that it is to be expected that it will gain more popularity in the near future.

Learning methods in attentional NMT models were very diverse. There were altogether five different learning methods present in the reviewed papers: SGD, Adam, Adadelta, RMSprop, and AdaGrad. Many authors used more than one in their model. The three most common ones were stochastic gradient descent (SGD) and optimisations Adam and Adadelta. The popularity of SGD was to be expected, but it is an interesting result that optimisations were equally popular. Furthermore, despite the popularity of RMSprop and AdaGrad in other

domains, each was used in only one of the reviewed papers, suggesting that these methods are not popular in this domain.

In regard of activation functions, there were altogether four different that were used in the reviewed models: softmax, tanh, maxout, and ReLu. The most common by a clear margin was softmax.

Finally, the types of computational units used in the hidden layers of the networks were scattered quite evenly between Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU). The even distribution of the use of GRU and LSTM is easily explained: LSTM as a unit is older and more customary to use, whereas GRU is more domain-specific. In the present data, GRU was the more common one, which is not surpising because of its alleged suitability for language-related tasks. Two variants of LSTM also appeared in two papers, one being Tree-LSTM and the other being Persistent Recurrent Unit (PRU).

6.3 Performance of attention-based NMT

My third research question was RQ3. *How well do attention-based NMT models perform in translation tasks?*. The main data to answer this research question is the BLEU scores presented in Section 5.5.1, which I will now sum up.

Table 30 sums up the best BLEU scores, BLEU averages, and best performing systems for each high-resource language pair. Results for low-resource language translation will be discussed in Section 6.4.

Table 30 shows that BLEU scores varied a lot between languages. The average scores for high-resource languages were between 17.61 BLEU for English-to-Czech and 36.77 BLEU for Chinese-to-English. The average is an especially powerful measure in cases where there are multiple papers per translation direction and the papers used the same corpora. For example, Chinese-to-English translation was featured in eight papers, which all used NIST MT translation tasks. However, in some cases, the average is clearly skewed by one unusually high score and small number of papers. One notable example is English-to-Japanese that featured one simpler translation task with an unusually high BLEU score.

Table 30. Best performing models, best BLEU scores, and BLEU averages

Language	Best paper	Best score	No of papers	Average
En - De	P6	28.40	11	23.42
De - En	P22	32.32	5	28.00
En - Fr	P6	41.00	5	35.64
Fr - En	P22	36.47	4	32.29
En - Cz	P4	20.70	3	17.61
Cz - En	P8	38.01	3	27.39
En - Ru	P22	22.21	2	20.88
Ru - En	P22	28.46	2	25.95
Ch - En	P9	41.68	8	36.77
En - Ch	P19	21.70	1	-
En - Jp	P16	51.04	3	33.61

There were two papers in which the proposed systems scored best for more than one translation direction: P6 and P22. The system that had most best scores across language pairs was P22 with four language directions, German-to-English, French-to-English, and English <-> Russian. P22 was also the second best system for English-to-German and English-to-French, while P6 was the best for these two. Of course, the total number of best performance per language direction cannot be used to define the best system because some authors only tested one or two pairs, while some, like in P22 and P3, included over 10 translation directions.

The best overall score was 51.04 BLEU in P16. However, as was already discussed in Section 5.5.1, this was highly affected by the easy translation task, BTEC corpus. This is a prime example how comparing BLEU scores between different corpora can be misleading. For best comparison, the same corpus should be used for all compared systems. For further research, quality assessment should include checking that authors compare all systems on the same corpora. In the present data, all authors did this. Furthermore, comparing translation results is more robust when all reviewed papers use the same corpus.

Overall, it seems that translating into English retrieves better BLEU scores than translating from English. However, there was also one exception to this in the present data: the average and best score for English-to-French is higher than the opposite direction. The significantly higher score and average is explained for the most part by the best performing system for the direction, P6, possibly also affected by its absence from the opposite direction. It is justifiable to rule out the effect of used dataset on the score, since the same dataset (WMT 2014) was also used in all but one paper in the same translation direction.

Table 31 puts together the BLEU scores and network features by summing up the features of the best performing models. The table includes all best and second best models for high-resource languages. P19 is not included because it was the only model for English-to-Chinese translation and only third best for French-to-English.

Table 31. Features of best performing models for high-resource languages

Paper	Architecture	Learning method	Activation function	Computational unit
P3	RNN	SGD, Adam	tanh	GRU
P4	RNN	SGD	softmax	LSTM
P5	RNN	Adadelta	softmax	GRU
P6	Transformer	Adam	ReLu	-
P8	Transformer	-	-	-
P9	RNN	RMSprop	maxout	GRU
P11	RNN	-	softmax	LSTM
P16	RNN	Adam	softmax	LSTM
P22	RNN	-	softmax	LSTM
P27	RNN	Adadelta	softmax	GRU

Table 31 shows that models were distributed very evenly and no single learning method or computational unit was clearly dominant in best performing models. Adam was used in three models, Adadelta and SGD in two separate models, and RMSprop in one. Learning methods for three models (P8, P11, and P22) were not disclosed. The use of computational unit was

very even, with five uses of LSTM and four uses of GRU. Transformer models P6 and P8 use neither computational unit. Britz et al. (2017) reported that LSTM cells perform better than GRU, but this claim cannot be enforced by the results of the current study. However, these results do not disprove it either. Regarding activation units, softmax was clearly the most used, but this is most likely directly connected to the prominent use of softmax in the data in general, as 14 models used softmax.

The excellent performance of Transformer models was perhaps the most interesting result regarding network architecture. The Transformer model was present in only two papers, P6 and P8, but they both had outstanding translation results. P6 received best score for both translation directions it included and P8 scored best in the sole language direction it featured. As was stated in Section 5.5.1, both papers featuring the Transformer model scored best for each translation direction they included, with a clear margin to the second best models, which featured RNN. Indeed, the Transformer model with its non-recurrent attentional model is an interesting topic for research in the future.

6.4 Low-resource languages and attention-based NMT

One of the aims of this study was to review how well attentional neural machine translation models handle low-resource language translation. I will now answer research question RQ4. *How well does attention-based NMT perform in translation tasks involving low-resource languages?* in light of the data gathered.

In the end, the reviewed papers included five papers with a total of four low-resource translation directions, which is very little to give a thorough answer to the research question. However, some interesting remarks and suggestions for more in-depth further research can be made. Table 32 sums up the best performing models, best BLEU score, and BLEU averages for low-resource languages.

On average, the BLEU scores for low-resource language translation were significantly lower than for high-resource languages. The lowest average was 9.69 BLEU for English-to-Finnish translation and the highest average was 19.8 BLEU for Turkish-to-English translation. Low scores for low-resource languages was an expected result and supports my initial hypothesis

Table 32. Best performing models, best BLEU scores, and BLEU averages for low-resource languages

Language	Best paper	Best score	No of papers	Average
En - Fi	P23	10.2	3	9.69
Fi - En	P5	13.6	3	12.82
Tr - En	P3	20.9	2	19.8
Uz - En	P24	16.8	2	14.57

that lack of data translates to lower performance.

However, some of the highest scoring models for low-resource languages were equal with high scores for some high-resource languages. The model in P24 scored 18.7 BLEU and model in P3 scored 20.9 BLEU for Turkish-to-English direction, which is comparable to for example 19.54 BLEU in the same paper (P3) for English-to-Russian translation, or 20.7 BLEU in P4 for English-to-Czech translation. On the one hand, one also needs to take into account that different datasets cannot be compared in a straightforward way: P3 used the LDC dataset for Turkish-to-English and newstest2015 for English-to-Russian. On the other hand, the LDC dataset was also used for Uzbek-to-English translation but with a significantly lower BLEU score. The size of the dataset for each individual pair might explain the difference. The size of the dataset was significantly higher for Turkish-to-English pair (over 800K sentence pairs) than for Uzbek-to-English pair (74K sentence pairs). This goes to show how flexible the concept of 'low-resource' can be. For further research, a study concentrating on low-resource languages especially would benefit from taking into account different sizes of datasets.

Table 33 sums up the features of best performing models for low-resource language translation. Here, only the best performing model for each pair were taken into account.

Similarly to high-resource languages, there was no outstanding network feature that would have been present in most of the best performing models. SGD and Adam were the most prominent learning methods, with two explicit uses and one implicit use (in P24) of Adam

Table 33. Features of best performing models for low-resource languages

Paper	Architecture	Learning method	Activation function	Computational unit
P3	RNN	SGD, Adam	tanh	GRU
P5	RNN	Adadelta	softmax	GRU
P23	RNN	Adam	-	LSTM
P24	RNN	-	softmax	LSTM

and one explicit use and one implicit use (in P24) of SGD. Adadelta was also used in one model, in P5. Softmax was used as an activation function in two systems, in P5 and P24, and hyperbolic tangent was used in one, P3. P23 did not specify any activation function. The use of computational unit was once again even, with P3 and P5 using GRU and P23 and P24 using LSTM. The observation by Britz et al. (2017) that LSTM performs better than GRU is neither supported nor refuted by this study in low-resource settings.

The Transformer model was not present for any of the reviewed articles for low-resource language translation. It would be beneficial to study the performance of Transformer in the low-resource domain in the future.

7 Conclusion

In the past decade, neural machine translation has become more computationally affordable and its translation quality has become comparable to that of traditional machine translation models and even human translators. The aim in this study was to present an overview of current research in one of the most popular types of neural machine translation, attentional neural machine translation.

Overall, the translation quality of attentional models in translation tasks was found to be good. The average BLEU score for many high-resource languages was over 20, with the best average being 36.77 for Chinese-to-English translation. The single best overall BLEU score was 51.04 BLEU for English-to-Japanese translation, however, the ease of the particular translation task affected this score. This highlights the importance of both taking the dataset into account as well as critical inspection of BLEU results. One valuable outcome to take from the present study is that even though BLEU scores are known to correlate with human evaluation and are the norm in reporting results, they are not an absolute mark of superiority of a model.

The attention mechanism was first presented in a recurrent neural network, and RNNs have maintained their dominance in attentional network models. However, the more recent non-recurrent Transformer architecture has achieved very promising results as well. Further research in using the Transformer model for translation is needed.

The network models in the reviewed papers covered a wide variety of network features. However, no single learning method, activation function, or type of computational unit significantly excelled above others in the present study. softmax was an especially popular activation function among the articles reviewed. More research in the relationship between network features and translation performance is needed. In addition to network features reviewed in this study, further research could also explore data preprocessing and postprocessing techniques, such as omission of long sentences, as well as different hyperparameters like beam search.

Reviewing translation quality of attentional NMT for low-resource languages was one of

the focal points in this study. The set of articles reviewed in the current study was very small, which is why the results should be interpreted as approximate. Low-resource language translation with all types of attentional network models, including the Transformer model, would be an interesting topic for further study.

All the models reviewed in this study were trained and tested with formal texts, mostly news texts. It would be an interesting topic for further research to look into translation in other domains, like prose, poetry, or humorous texts. Once machine translation is able to master translation in multiple language domains, we are a step closer to creating the incredibly useful simultaneous universal translator devices we see in science fiction.

Bibliography

Aggarwal, Charu C. 2018. *Neural Networks and Deep Learning: A Textbook.* Springer International Publishing. ISBN: 9783319944630. doi:10.1007/978-3-319-94463-0. https://link.springer.com/book/10.1007/978-3-319-94463-0.

Bahdanau, Dzmitry, Kyunghyun Cho, and Yoshua Bengio. 2015. "Neural Machine Translation by Jointly Learning to Align and Translate". In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings,* edited by Yoshua Bengio and Yann LeCun. http://arxiv.org/abs/1409.0473.

Basmatkar, P., H. Holani, and S. Kaushal. 2019. "Survey on Neural Machine Translation for multilingual translation system". In *2019 3rd International Conference on Computing Methodologies and Communication (ICCMC),* 443–448. doi:10.1109/ICCMC.2019.8819788.

Bentivogli, Luisa, Arianna Bisazza, Mauro Cettolo, and Marcello Federico. 2016. "Neural versus Phrase-Based Machine Translation Quality: a Case Study". In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing,* 257–267. Austin, Texas: Association for Computational Linguistics. doi:10.18653/v1/D16-1025. https://www.aclweb.org/anthology/D16-1025.

Bishop, Christopher M. 2006. *Pattern Recognition and Machine Learning (Information Science and Statistics).* Berlin, Heidelberg: Springer-Verlag. ISBN: 0387310738.

Britz, Denny, Anna Goldie, Minh-Thang Luong, and Quoc Le. 2017. "Massive Exploration of Neural Machine Translation Architectures". In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing,* 1442–1451. Copenhagen, Denmark: Association for Computational Linguistics. doi:10.18653/v1/D17-1151. https://www.aclweb.org/anthology/D17-1151.

Chaudhary, Janhavi R, and Ankit C Patel. 2018. “Machine Translation Using Deep Learning: A Survey”. *Journal of Scientific Research in Science, Engineering and Technology* 4 (2): 145–150.

Cho, Kyunghyun, Bart van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. 2014a. “On the Properties of Neural Machine Translation: Encoder–Decoder Approaches”. In *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation,* 103–111. Doha, Qatar: Association for Computational Linguistics. doi:10.3115/v1/W14-4012. https://www.aclweb.org/anthology/W14-4012.

Cho, Kyunghyun, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014b. “Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation”. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP),* 1724–1734. Doha, Qatar: Association for Computational Linguistics. doi:10.3115/v1/D14-1179. https://www.aclweb.org/anthology/D14-1179.

Choi, Heeyoul. 2019. “Persistent hidden states and nonlinear transformation for long short-term memory”. *Neurocomputing* 331:458–464. ISSN: 0925-2312. doi:https://doi.org/10.1016/j.neucom.2018.11.069. http://www.sciencedirect.com/science/article/pii/S0925231218314152.

Choi, Heeyoul, Kyunghyun Cho, and Yoshua Bengio. 2018. “Fine-grained attention mechanism for neural machine translation”. *Neurocomputing* 284:171–176. ISSN: 0925-2312. doi:https://doi.org/10.1016/j.neucom.2018.01.007. http://www.sciencedirect.com/science/article/pii/S0925231218300225.

Duchi, John, Elad Hazan, and Yoram Singer. 2011. “Adaptive Subgradient Methods for Online Learning and Stochastic Optimization”. *Journal of Machine Learning Research* 12 (2011): 2121–2159.

EMNLP. 2015. “EMNLP 2015 Tenth Workshop on Statistical Machine Translation”. Accessed: 12-04-2020. http://www.statmt.org/wmt15/.

Gers, Felix A., Nicol N. Schraudolph, and Jürgen Schmidhuber. 2002. “Learning Precise Timing with LSTM Recurrent Networks”. *Journal of Machine Learning Research* 3 (2002): 115–143. ISSN: 1532-4435. doi:10.1162/153244303768966139.

Ha, Thanh-Le, Jan Niehues, and Alexander Waibel. 2016. “Toward multilingual neural machine translation with universal encoder and decoder”. *arXiv preprint arXiv:1611.04798.*

Haapanen, Ville. 2018. “Pelimusiikin adaptiivisuus : systemaattinen kirjallisuuskartoitus”. Master’s thesis, University of Jyväskylä.

Hinton, Geoffrey. 2020. *Neural Networks for Machine Learning - Lecture 6a - Overview of mini-batch gradient descent.* Accessed: 09-05-2020. https://www.cs.toronto.edu/~tijmen/csc321/slides/lecture_slides_lec6.pdf.

Hinton, Geoffrey E., Simon Osindero, and Yee-Whye Teh. 2006. “A Fast Learning Algorithm for Deep Belief Nets”. *Neural Computation* (Cambridge, MA, USA) 18, number 7 (): 1527–1554. ISSN: 0899-7667. doi:10.1162/neco.2006.18.7.1527. https://doi.org/10.1162/neco.2006.18.7.1527.

Hochreiter, Sepp, and Jürgen Schmidhuber. 1997. “Long Short-term Memory”. *Neural computation* 9, number 8 (): 1735–1780. doi:10.1162/neco.1997.9.8.1735.

Kingma, Diederik P., and Jimmy Ba. 2015. “Adam: A Method for Stochastic Optimization”. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings,* edited by Yoshua Bengio and Yann LeCun. http://arxiv.org/abs/1412.6980.

Kitchenham, Barbara Ann, David Budgen, and Pearl Brereton. 2016. *Evidence-based software engineering and systematic reviews.* 399. Chapman & Hall/CRC innovations in software engineering and software development. Boca Raton: CRC Press.

Koehn, Philipp. 2009. *Statistical machine translation.* Cambridge University Press.

Koehn, Philipp, and Rebecca Knowles. 2017. “Six Challenges for Neural Machine Translation”. In *Proceedings of the First Workshop on Neural Machine Translation,* 28–39. Vancouver: Association for Computational Linguistics. doi:10.18653/v1/W17-3204. https://www.aclweb.org/anthology/W17-3204.

Lipton, Zachary Chase. 2015. “A Critical Review of Recurrent Neural Networks for Sequence Learning”. *ArXiv* abs/1506.00019.

Luong, Thang, Hieu Pham, and Christopher D. Manning. 2015. “Effective Approaches to Attention-based Neural Machine Translation”. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing,* 1412–1421. Lisbon, Portugal: Association for Computational Linguistics. doi:10.18653/v1/D15-1166. https://www.aclweb.org/anthology/D15-1166.

Mononen, Niko. 2018. “Systemaattinen kirjallisuuskartoitus luovasta ohjelmoinnista opetuskontekstissa”. Master’s thesis, University of Jyväskylä.

Nielsen, Michael A. 2015. *Neural networks and deep learning.* Volume 25. Determination press USA.

Nord, Christiane. 2014. *Translating as a purposeful activity: Functionalist approaches explained.* Routledge.

Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. “BLEU: a method for automatic evaluation of machine translation”. In *Proceedings of the 40th annual meeting on association for computational linguistics,* 311–318. Association for Computational Linguistics.

Peuron, Ilkka. 2017. “Ohjelmistoarkkitehtuurit peleissä : systemaattinen kirjallisuuskatsaus”. Master’s thesis, University of Jyväskylä.

Pozdniakova, Olesia, and Dalius Mazeika. 2017. “Systematic Literature Review of the Cloud-Ready Software Architecture”. *Baltic Journal of Modern Computing* 5 (). doi:10.22364/bjmc.2017.5.1.08.

Python Software Foundation. 2020. “random — Generate pseudo-random numbers”. Visited on January 29, 2020. https://docs.python.org/3/library/random.html.

Rikters, Matīss. 2019. “Hybrid Machine Translation by Combining Output from Multiple Machine Translation Systems”. *Baltic Journal of Modern Computing* 7 (3): 301–341.

Stein, Gregory J. 2018. “For AI, Translation is about more than language”. Visited on October 7, 2018. http://cachestocaches.com/2018/9/ai-translation-more-language/.

Sutskever, Ilya, Oriol Vinyals, and Quoc V Le. 2014. “Sequence to sequence learning with neural networks”. In *Advances in neural information processing systems,* 3104–3112.

Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. “Attention is All you Need”. In *Advances in Neural Information Processing Systems 30,* edited by I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, 5998–6008. Curran Associates, Inc. http://papers.nips.cc/paper/7181-attention-is-all-you-need.pdf.

WAT. 2015. “WAT 2015 The 2nd Workshop on Asian Translation”. Accessed: 12-04-2020. https://lotus.kuee.kyoto-u.ac.jp/WAT/WAT2015.

Wu, Yonghui, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, et al. 2016. “Google’s Neural Machine Translation System: Bridging the Gap between Human and Machine Translation”. *CoRR* abs/1609.08144. arXiv: 1609.08144. http://arxiv.org/abs/1609.08144.

Zeiler, Matthew. 2012. “ADADELTA: An adaptive learning rate method”. 1212 (). arXiv: 1212.5701. https://arxiv.org/abs/1212.5701.

Zhang, Jiajun, and Chengqing Zong. 2016. “Exploiting Source-side Monolingual Data in Neural Machine Translation”. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing,* 1535–1545. Austin, Texas: Association for Computational Linguistics. doi:10.18653/v1/D16-1160. https://www.aclweb.org/anthology/D16-1160.

Zoph, Barret, and Kevin Knight. 2016. “Multi-Source Neural Translation”. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies,* 30–34. San Diego, California: Association for Computational Linguistics. doi:10.18653/v1/N16-1004. https://www.aclweb.org/anthology/N16-1004.