DISSERTATIONES
INFORMATICAE
UNIVERSITATIS
TARTUENSIS
35

ELIZAVETA YANKOVSKAYA

Quality Estimation through Attention

DISSERTATIONES INFORMATICAE UNIVERSITATIS TARTUENSIS
35

DISSERTATIONES INFORMATICAE UNIVERSITATIS TARTUENSIS
35

ELIZAVETA YANKOVSKAYA

Quality Estimation through Attention

UNIVERSITY OF TARTU
Press
1632

Institute of Computer Science, Faculty of Science and Technology, University of Tartu, Estonia.

Dissertation has been accepted for the commencement of the degree of Doctor of Philosophy (PhD) in Computer Science on May 3, 2022 by the Council of the Institute of Computer Science, University of Tartu.

Supervisor

Prof. Mark Fishel
Institute of Computer Science
University of Tartu, Tartu, Estonia

Opponents

Dr. Chi-kiu (Jackie) Lo
Digital Technologies Research Centre
National Research Council Canada

Prof. Dr. Rico Sennrich
Department of Computational Linguistics
University of Zurich, Switzerland

The public defense will take place on June 17, 2022 at 16:15 in Narva rd 18, room 1021 and via Zoom.

The publication of this dissertation was financed by the Institute of Computer Science, University of Tartu.

ISSN 2613-5906
ISBN 978-9949-03-893-0 (print)
ISBN 978-9949-03-894-7 (PDF)

University of Tartu Press
http://www.tyk.ee/

To covid

ABSTRACT

In recent years, the use of machine translation (MT) systems has increased dramatically. Today machine translation is used not only by large corporations, government services and translation agencies, but also by people who want, for example, to know what their favourite song is about. With the development of machine translation systems, the quality of translation has also improved. However, translation quality still varies significantly not only across different machine translation systems, but also across translations produced by the same system. Modern MT systems usually generate fluent translations, but some of these translations may miss crucial details or completely misrepresent the original sentence. Thus, we need to evaluate each translation of each system to make sure that the translation does not distort the meaning of the original sentence.

In the case of translation agencies, professional translators edit the results of machine translation. However, in some scenarios, for example, online machine translation systems, it is not possible to assess translation quality with human editors. That is why automated systems for measuring translation quality are a crucial part of the machine translation pipeline.

There are two types of automated systems for estimating translation quality: with and without the use of reference translation(s). The former is often referred to as metrics or reference-based metrics; the second is called Quality Estimation (QE) metrics. We generally use reference-based metrics to assess the quality of MT output while training MT systems, whereas we can only apply QE metrics for measuring translation quality at run-time.

In this thesis, we focus on QE metrics and consider the distribution of attention—one of the internal parameters of modern neural machine translation (NMT) systems—as an indicator of translation quality. We first apply it to translations generated by NMT systems based on recurrent neural networks (RNNs). We examine how the proposed models work in both unsupervised and supervised way. The main drawback of supervised models is using human-annotated data, since labeling data by professional translators is a time-consuming and relatively costly task. That is why it is essential to have not only supervised models but also unsupervised ones. In addition, we show that our approach is applicable to translations produced by any unknown machine translation system.

Since transformers had replaced RNN-based MT systems, we adapted our approach to the transformers architecture. We examine how it performs in unsupervised, semi-supervised and supervised tasks. For supervised tasks, we study how much annotated data is needed to train a QE model. In addition, we demonstrate that supervised models achieve a sensible correlation with human judgments even with the use of synthetic labelled data.

CONTENTS

LIST OF PUBLICATIONS INCLUDED IN THE THESIS

I **Yankovskaya, E.**, Tättar, A., Fishel, M. (2018a). Quality Estimation with Force-Decoded Attention and Cross-lingual Embeddings. Proceedings of the Third Conference on Machine Translation Association for Computational Linguistics (ACL), 816–821.

Author's contributions: Set up all experiments, including computing input features and labels; had a major role in writing the paper.

II **Yankovskaya, E.**, Fishel, M. (2018b). Low-Resource Translation Quality Estimation for Estonian. In K. Muischnek, K. Müürisep (Eds.), Human Language Technologies – The Baltic Perspective (pp. 175–182), IOS Press.

Author's contributions: Implemented the analysis pipeline and computed all attention-based input features; had a major role in writing the paper.

III Fomicheva, M., Sun, S., **Yankovskaya, L.**, Blain, F., Guzmán, F., Fishel, M., Aletras, N., Chaudhary, V., Specia, L. (2020d). Unsupervised Quality Estimation for Neural Machine Translation. Transactions of the Association for Computational Linguistics, 8, 539–555.

Author's contributions: Modified `fairseq`[1] transformer MT models to extract attention weights for all heads; was responsible for conducting the attention mechanism research.

IV Fomicheva*, M., Sun*, S., **Yankovskaya***, **L.**, Blain, F., Chaudhary, V.,Fishel, M., Guzmán, F., Specia, L. (2020). BERGAMOT-LATTE Submissions for the WMT20 Quality Estimation Shared Task. Proceedings of the Fifth Conference on Machine Translation, 1010–1017.

Author's contributions: Ran all experiments with glass-box features; had a significant role in writing the paper.

V **Yankovskaya, L.**, Fishel, M. (2021). Direct exploitation of attention weights for translation quality estimation. Proceedings of the Sixth Conference on Machine Translation, 955–960.

Author's contributions: Designed and performed all tests; had a major role in writing the paper.

* - authors contributed equally

[1] https://github.com/pytorch/fairseq

1. INTRODUCTION

Machine Translation (MT) is a quickly growing field of Natural Language Processing (NLP). To see it, we compare two translations of the same sentence made three years apart. The first translation[1] generated in 2018 is "Like many calories per day than you should be on a regular basis, even depends on your physical activities.". The second one[2] is "The maximum number of calories you should eat per day also depends on your physical activities.". Even without knowing the original sentence[3], it is clear the second translation sounds more fluent and readable.

Although the transition to neural networks from statistical methods has significantly improved the quality of translations (Wu et al., 2016), translation quality varies across MT systems, language pairs and domains. While we have near-human quality for high-resource language pairs, like English-German, for languages with a limited amount of training data, such as Nepalese-English, the quality of translations is still relatively low. In addition, neural MT systems can generate fluent translations that completely distort the meaning of the original sentence. For example, Estonian sentence "Mina olen leevike" means "I am a bullfinch", but Google Translate translates it as "I am a freak"[4]. In this example, the incorrect translation makes us smile, but translation errors can be more serious and lead to tragic consequences. Thus, it is important to have an engine that measures translation quality as a part of an MT pipeline.

How can we measure the quality of translations? There are three main ways: (1) human annotations, (2) reference-based metrics and (3) Quality Estimation (QE) metrics. Human annotations provide the most reliable output; however, it is most expensive and time-consuming. Besides that, it is extremely expensive and impractical to use human annotations for evaluating translation at runtime. Reference-based and QE metrics are automated approaches that require little or no human intervention. The main difference between reference-based metrics and QE is the use of reference sentences. Reference-based metrics compare translation output and its corresponding reference(s), whereas QE methods assess the quality of translation without applying any gold-standard human translation. This fact makes them valuable for evaluating online MT systems.

QE approaches can be divided into three groups: (1) relying on MT system-independent features, so-called black-box, such as additional corpora, pre-trained representation of words and mixed models; (2) internal features of MT systems or glass-box features; and (3) their combination.

[1]The translation is taken from the German-English development set WMT18 (Specia et al., 2018)

[2]We translated it using Google Translate on November 3, 2021

[3]The original sentence is "Wie viele Kalorien Sie pro Tag höchstens zu sich nehmen sollten, hängt auch von Ihren körperlichen Aktivitäten ab."

[4]The translation is generated on November 3, 2021

QE methods based on black-box features generally yield better results than approaches using glass-box features. Their main disadvantage is that they require additional resources that may not be available or hard to collect (for example, if we want to translate Finnish into Lithuanian or if we need to evaluate translations of lesser-used languages such as Faroese or dialects). In addition, QE black-box models tend to be computationally intensive. In these cases, the benefits of glass-box QE systems become apparent.

There are QE systems based on glass-box features extracted from statistical MT systems, such as language model probabilities or the n-best list of translation hypotheses (Blatz et al., 2004; Specia et al., 2013). The transition to neural MT systems made the use of these features almost useless, but at the same time, it allowed us to explore the internal features of neural MT systems, such as uncertainty quantification, softmax distribution, and attention weights (Bahdanau et al., 2015).

In this thesis, we focus on attention weights (Bahdanau et al., 2015)—one of the internal features of MT systems. Before the advent of the attention mechanism, neural MT systems did not cope with the translation of long sentences well, "forgetting" the beginning of the original sentence. It happened because all information about the whole source sentence was captured into one vector, which led to the fact that the beginning of the source sentence carried less information than its end. The attention mechanism allowed an MT system to overcome this problem by assigning higher weights to a particular part of the source sentence in the overall representation.

1.1. Research Goals

In 2017, when we started our research, the only work that looked at the internal information extracted from neural MT systems as an indicator of the translation quality was a work of Rikters et al. (2017). Authors explore how the attention distribution can be used to evaluate the outputs of machine translation systems. Their analysis is done using a total of 200 sentences and only for translations produced by neural MT systems with the attention mechanism.

Inspired by this work, we have focused our research on attention weights to further explore how they can be used for quality estimation purposes. Our **first research goal** is examination of attention weights extracted from MT systems based on recurrent neural networks (RNNs) as a QE indicator. To examine it, we discussed several aspects. First of all, we studied how well attention-based QE models perform for supervised as well as unsupervised QE tasks (Research Question 1, RG1-Q1). It is worth considering unsupervised models, since obtaining gold labels needed to train supervised models is a time-consuming and an expensive task. As statistical MT systems and neural MT systems without the attention mechanism were still common in 2017, it was important to investigate whether the attention-based approach can be used for translations produced by any unknown

MT system (RG1-Q2).

Since transformers (Vaswani et al., 2017) had supplanted RNN-based MT systems and had become state-of-the-art, we explored the attention distribution extracted from transformer-based MT systems in terms of translation quality. This is **the second research goal** of this thesis. Although the encoder-decoder attention mechanism of transformer-based MT models is similar to the attention mechanism used in RNN-based MT systems, unlike the latter, it consists of several attention matrices. Thus, the previous attention-based approach has to be adapted to the new architecture (RG2-Q1).

Similar to the RNN-based systems, we explored performance of the unsupervised and supervised QE models for transformer-based MT models; in addition, we extended them to supervised models using synthetically generated labels (RG2-Q2). Apart from that, we examined how much annotated data is needed to train attention-based supervised models (RG2-Q3). Like any approach, the attention-based approach has its weaknesses; we discussed its limitations and possible ways to overcome them (RG2-Q4).

2. BACKGROUND

As we mentioned in Introduction, most modern machine translation systems are trained using deep neural network algorithms. In our experiments, we used MT systems based on recurrent neural networks and transformer.

In the first part of this chapter, we briefly discuss the main modifications of recurrent neural networks that have improved translation quality and we talk about state-of-the-art architecture—transformer. In the second part, we focus on quality estimation of MT outputs. Most QE systems today also use machine learning algorithms, including neural networks. We talk about a framework used in the most efficient QE systems, but at the the same time these QE systems tend to be computationally intensive, so we have a trade-off between QE models performance and the amount of resources required. In addition, we discuss why there are different reference-less types of MT evaluation and how they are correlated.

2.1. Recurrent Neural Networks

Recurrent neural networks (Rumelhart et al., 1985, RNNs) are a class of neural networks that work with time series data or sequential data. One of the essential features of RNNs is that the output of each time step depends not only on the input but also on the prior information (see Figure 1). In addition, RNNs can process inputs and outputs of any length and share the learnable parameters across different positions of data. All of these features together help to handle sequence data.

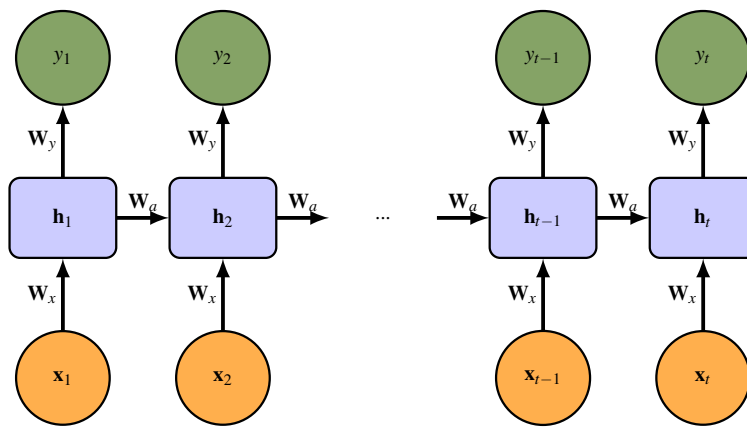


Figure 1: Architecture of a many-to-many recurrent neural network. "Many-to-many" reflects that the number of input and output tokens is more than one.

The exact formula to compute the hidden state $\mathbf{h}_t$ and the output $\mathbf{y}_t$ at each time step t are expressed below:

$$\begin{aligned} \mathbf{h}_t &= g_1(\mathbf{W}_a\mathbf{h}_{t-1} + \mathbf{W}_x\mathbf{x}_t + \mathbf{bias}) \\ y_t &= g_2(\mathbf{W}_y\mathbf{h}_t + \mathbf{bias}_y) \end{aligned} \tag{2.1}$$

where $\mathbf{W}_a$, $\mathbf{W}_x$ and $\mathbf{W}_y$ are weight matrices, g_1 and g_2 are activation functions.

One of the limitations of RNNs is that they only see the previous time steps, $\mathbf{x}_{(1)}, \ldots, \mathbf{x}_{(t-1)}$, and the present input $\mathbf{x}_t$, but not future ones. However, in many applications, it is crucial to observe the whole sequential input. For example, in machine translation, the correct translation of the current word may depend on the next words. Bidirectional-RNNs (Schuster et al., 1997) address this problem as they can be trained simultaneously in both time directions: from the first time step to the last and from the last to the first time step.

Another common weakness of RNNs is the vanishing gradient. It is happening because we use gradient descent with the back-propagation algorithm to train a neural network, and the gradient can get very small with respect to the number of layers/time steps. As a result, it is quite hard to capture long-term dependencies well. To tackle this problem, Long Short-Term Memory (Hochreiter et al., 1997, LSTM) cells were used.

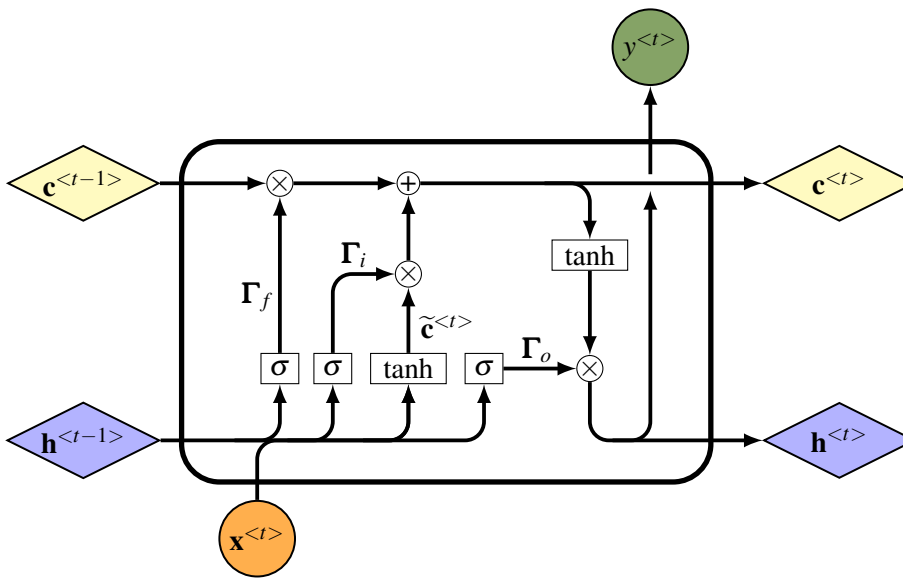


Figure 2: The LSTM cell.

LSTM is a modification of the hidden layer of RNNs that makes it much better to handle long-term dependencies. Instead of neurons, LSTM has memory units (see Figure 2), each containing a gate that has a well-defined specific purpose:

- Input gate $\mathbf{\Gamma}_i$ controls how much we want to update a new cell;
- Forget gate $\mathbf{\Gamma}_f$ determines what information to keep or discard from the unit;
- Output gate $\mathbf{\Gamma}_o$ decides how much to reveal of a current state cell.

The equations below mathematically describe LSTM architecture:

$$
\begin{aligned}
\mathbf{\Gamma}_f &= \sigma(\mathbf{U}_f \mathbf{h}_{t-1} + \mathbf{W}_f \mathbf{x}_t + \mathbf{bias}_f) \\
\mathbf{\Gamma}_i &= \sigma(\mathbf{U}_i \mathbf{h}_{t-1} + \mathbf{W}_i \mathbf{x}_t + \mathbf{bias}_i) \\
\mathbf{\Gamma}_o &= \sigma(\mathbf{U}_o \mathbf{h}_{t-1} + \mathbf{W}_o \mathbf{x}_t + \mathbf{bias}_o) \\
\widetilde{\mathbf{c}}_t &= tanh(\mathbf{U}_c \mathbf{h}_{t-1} + \mathbf{W}_c \mathbf{x}_t + \mathbf{bias}_c) \\
\mathbf{c}_t &= \mathbf{\Gamma}_i \cdot \widetilde{\mathbf{c}}_t + \mathbf{\Gamma}_f \cdot \mathbf{c}_{t-1} \\
\mathbf{h}_t &= \mathbf{\Gamma}_o \cdot tanh(\mathbf{c}_t)
\end{aligned}
\tag{2.2}
$$

where $\mathbf{U}$ and $\mathbf{W}$ are weight matrices specifically designed for each gate, `tanh` and σ are hyperbolic tangent and sigmoid functions, $\widetilde{\mathbf{c}}$ is a candidate value for the internal cell state $\mathbf{c}$, $\cdot$ denotes the element-wise product.

2.2. Encoder-decoder architecture

Machine translation is an example of converting one sequence to another which is not necessarily of the same length. A typical architecture for solving this kind of problem is shown in Figure 3. It consists of two parts: an encoder and a decoder, hence its name - the encoder-decoder architecture (Cho et al., 2014; Sutskever et al., 2014). Both Cho et al. (2014) and Sutskever et al. (2014) used recurrent neural networks as an encoder and a decoder.

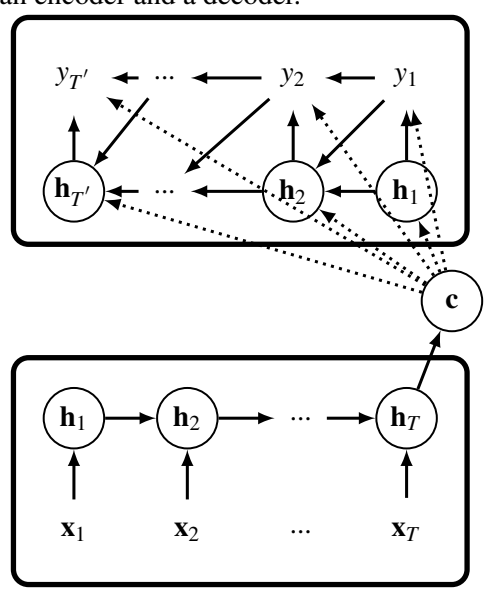


Figure 3: The graphical illustration of the RNN Encoder-Decoder Cho et al., 2014

The encoder reads each token of the input sequence $\mathbf{x} = [\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_T]$ and calculates hidden states at each step t:

$$\mathbf{h}_t = f(\mathbf{h}_{t-1}, \mathbf{x}_t) \tag{2.3}$$

After reading the whole sentence, we get the context vector $\mathbf{c}$ that is actually the last hidden state $\mathbf{h}_T$ of the encoder. This is expected to be a summary of the entire source sentence.

This vector $\mathbf{c}$ is passed to the decoder as an input; and the decoder computes its hidden state at step t taking into account the vector $\mathbf{c}$:

$$\mathbf{h}_t = f(\mathbf{h}_{t-1}, y_{t-1}, \mathbf{c}) \tag{2.4}$$

and the conditional distribution of y_t token is

$$P(y_t|y_{t-1}, y_{t-2}, ..., y_1, \mathbf{c}) = g(\mathbf{h}_t, y_{t-1}, \mathbf{c}), \tag{2.5}$$

where f and g are activation functions.

2.3. Attention mechanism

In the encoder-decoder architecture described above, the encoder reads and transforms the entire input sentence into a fixed-length vector $\mathbf{c}$, regardless of the input's length. Thus, all information about the whole sentence is compressed into one vector and the last input tokens are given more importance, as a result, the model may not cope with long sentences. To address this problem, Bahdanau et al. (2015) proposed the attention mechanism.

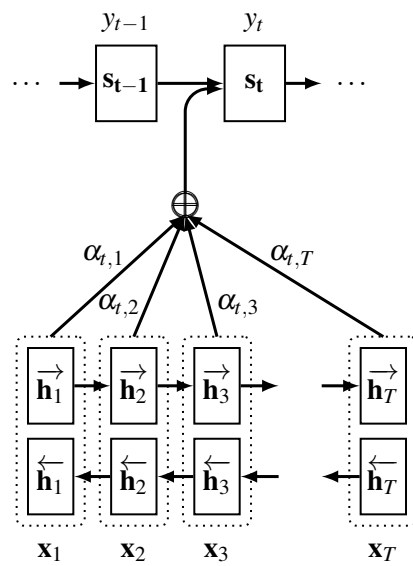


Figure 4: The graphical illustration of the attention mechanism proposed by Bahdanau et al. (2015).

The core idea of the mechanism is to take into account all hidden states of the encoder and focus on a particular part of the source sentence on each step

of the decoder (Figure 4). In other words, instead of using a fixed-length vector, Bahdanau et al. (2015) suggest using a variable-length vector. To some extent, this is similar to the work of a translator, who usually focuses not only on the entire original sentence but also on its various parts, especially if the given sentence is long.

Here, the encoder is a bidirectional RNN, the forward RNN reads the input sequence $\mathbf{x} = [\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_T]$ from the beginning and calculates a sequence of hidden states $(\overrightarrow{\mathbf{h}_1}, \overrightarrow{\mathbf{h}_2}, ..., \overrightarrow{\mathbf{h}_T})$, whereas the backward RNN reads the input in the reverse order, computing backward hidden states $(\overleftarrow{\mathbf{h}_1}, \overleftarrow{\mathbf{h}_2}, ..., \overleftarrow{\mathbf{h}_T})$. To get the summary of both the following and preceding words for a word x_i, we concatenate hidden states into one $\mathbf{h}_i = [\overrightarrow{\mathbf{h}_i}, \overleftarrow{\mathbf{h}_i}]$.

The context vector $\mathbf{c}_t$ is computed as a weighted sum of these hidden states:

$$\mathbf{c}_t = \sum_{i=1}^{T} \alpha_{ti} \mathbf{h}_i \tag{2.6}$$

where α_{ti} is the attention weight of each $\mathbf{h}_i$:

$$\alpha_{ti} = \frac{exp(e_{ti})}{\sum_{k=1}^{T} exp(e_{tk})} \tag{2.7}$$

where e_{ti} is defined by the previous hidden state of the decoder $\mathbf{s}_{t-1}$ and encoder's hidden state $\mathbf{h}_i$ as $score(\mathbf{s}_{t-1}, \mathbf{h}_i)$.

In (Bahdanau et al., 2015), the alignment score α is parametrized by a feed-forward neural network which is jointly trained with the whole system. The score $score(\mathbf{s}_{t-1}, \mathbf{h}_i)$ is calculated by:

$$score(\mathbf{s}_{t-1}, \mathbf{h}_i) = \mathbf{v}_a^T tanh(\mathbf{W}_a[\mathbf{s}_{t-1}; \mathbf{h}_i]) \tag{2.8}$$

where $\mathbf{v}_a$ and $\mathbf{W}_a$ are learnable weight matrices.

In contrast to the usual encoder-decoder architecture with a single context vector $\mathbf{c}$, the proposed decoder generates its hidden state $\mathbf{s}_t$ with the unique context vector $\mathbf{c}_t$ for each target token t:

$$\mathbf{s}_t = f(\mathbf{s}_{t-1}, y_{t-1}, \mathbf{c}_t). \tag{2.9}$$

The attention mechanism solves the problem of poor translation of long sentences, and also mitigates the vanishing gradient problem. Thus, the use of neural MT systems with the attention mechanism significantly improves neural MT performance. Besides that, attention distribution provides some interpretability — it allows us to see what part of the source sentence the decoder was focusing on when generating the output.

2.4. Transformer

While RNN-based MT systems achieved great performance around 2016 (Wu et al., 2016), they suffered from a lack of parallelizability. RNNs do not allow us to compute future hidden states before previous hidden states have been computed, resulting in very extensive training time. The use of LSTMs only partially helped to solve the vanishing gradient problem related with length of the sequences. Thus, RNN-based MT systems still cannot handle long-distant dependencies well.

To remedy these issues, Vaswani et al. (2017) proposed the transformer architecture. It is based on the encoder-decoder architecture, but compared to the previous encoder-decoder MT models, it no longer uses sequence-aligned RNNs, but instead relies entirely on the attention mechanism.

The encoding part (the left part of Figure 5) is a stack of N identical encoders (the setting described in the paper uses six encoders) on top of each other. Each encoder has two layers: a multi-head self-attention mechanism and a fully connected feed-forward neural network. The encoder's input is fed to the multi-head self-attention layer — which helps to encode a specific token as well as taking a look at other tokens. Its output then goes through the feed-forward neural network. There is a residual connection (He et al., 2016) around of each of the two sub-layers, followed by layer normalization (Ba et al., 2016); using residual connections and layer normalization helps models to train better and faster.

The decoding part (the right part of Figure 5) is also composed of a stack of N identical decoders. Like the encoder, the decoder has the multi-head self-attention and the fully connected feed-forward neural network layers and in addition to them, it has an encoder-decoder attention layer between them. The encoder-decoder attention layer is similar to the encoder-decoder attention mechanism described in Section 2.3 and it allows the decoder to focus on the relevant parts of the input sentence. Residual connections with a layer-normalization step are also wrapped around each sub-layer. In contrast to the encoder's multi-head self-attention layer, a masked layer is applied. It ensures that during the prediction of the token i, the system does not see the future tokens, thus making its decision is based only on the known outputs at positions less than i.

Attentions are the crucial part of the transformer architecture. To describe how attentions work, we use three vectors named as query $\mathbf{q}$, value $\mathbf{v}$ and key $\mathbf{k}$ which are created for each embedded token. One of the powerful ideas is to use matrices that allows to compute attention on a set of queries $\mathbf{Q}$, keys $\mathbf{K}$ and values $\mathbf{V}$ at the same time.

$$Attention(\mathbf{Q},\mathbf{K},\mathbf{V}) = softmax\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}}\right)\mathbf{V}, \tag{2.10}$$

where d_k is the dimensions of k, $\sqrt{d_k}$ prevents from the extremely small gradients.

There is a minor adjustment to computing attention in the transformer architecture, called "multi-head attention". Instead of one set of $(\mathbf{Q},\mathbf{K},\mathbf{V})$ weight ma-

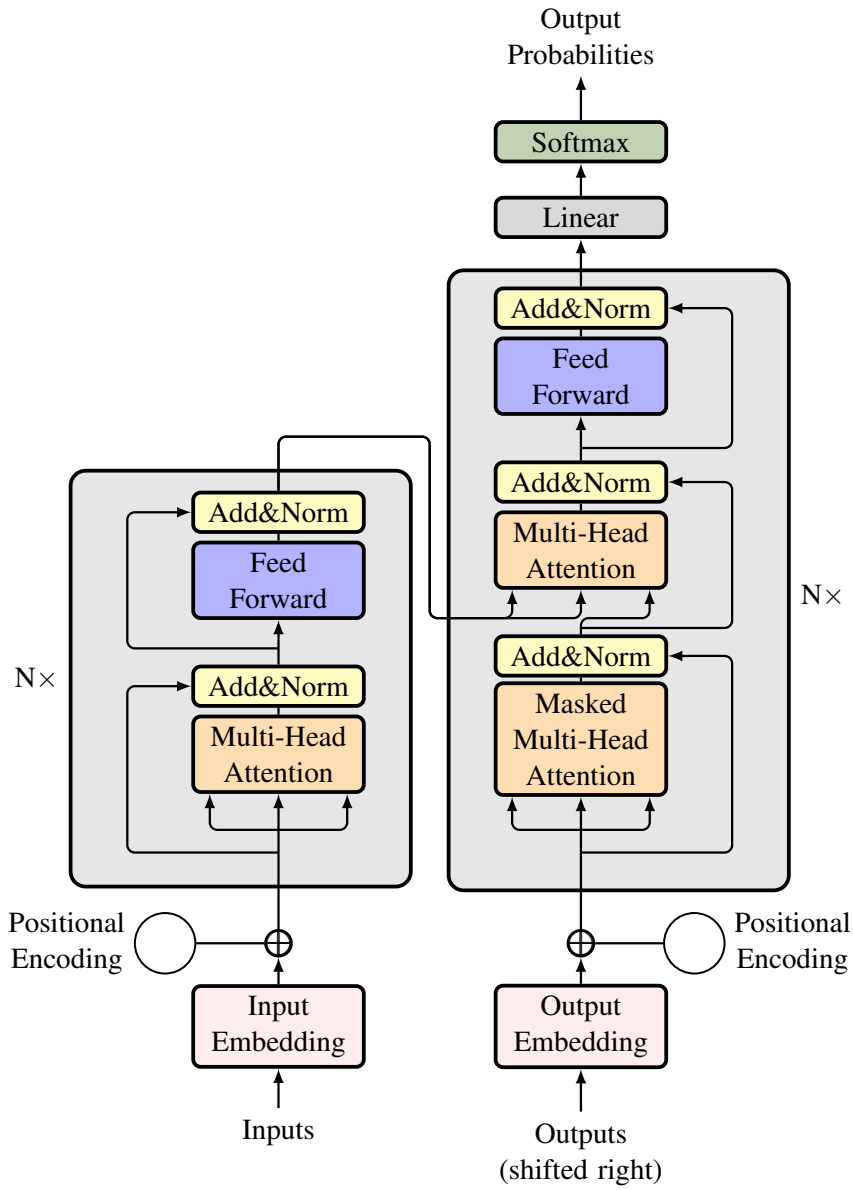


Figure 5: The Transformer - model architecture (Vaswani et al., 2017).

trices, the transformer has n sets of $(\mathbf{Q}, \mathbf{K}, \mathbf{V})$ matrices, where n is the number of heads. It lets create multiple representation sub-spaces.

$$\begin{aligned} MultiHead(\mathbf{Q}, \mathbf{K}, \mathbf{V}) &= Concat(head_1, ..., head_n)\mathbf{W}^o, \\ \text{where } head_i &= Attention(\mathbf{Q}\mathbf{W}_i^Q, \mathbf{K}\mathbf{W}_i^K, \mathbf{V}\mathbf{W}_i^V), \end{aligned} \tag{2.11}$$

where $\mathbf{W}$ are weighted matrices that were trained jointly with the model.

As we mentioned above, there are two different types of attention used in the

transformer architecture: self-attention and encoder-decoder attention. The key difference between them is that all queries **Q**, keys **K** and values **V** come from the output of the previous layer of the encoder or the decoder in self-attention layers. In contrast, in encoder-decoder attention layers, the queries **Q** come from the decoder part and the keys **K** and the values **V** from the encoder.

2.5. Quality Estimation of Machine Translation

There are generally three types of QE systems depending on the granularity of the predictions: at the word, sentence or document level. The goal of word-level QE is to label a correctly translated word as "OK" and an incorrectly translated word as "BAD". Sentence-level and document-level QE models aim at predicting the quality of a whole translated sentence or a document.

In this thesis, we focus on sentence-level QE systems, that is, when we talk about QE systems, we mean sentence-level QE, unless other details are mentioned.

In this section, we will discuss two aspects of quality estimation of machine translation. First, we will talk about a framework that is used in the best performing QE systems and its limitations. After that, we will examine what kind of labels QE systems predict and why it does not make sense to use the same QE model to predict translation quality from the perspective of a human and a translator.

2.5.1. Trade-off between performance and "lightweightness"

As mentioned in Introduction, there are two types of features applied in Quality Estimation models: *glass-box* features extracted from the machine translation systems and *black-box* obtained from the external resources.

Fomicheva et al. (2020a, which is our Publication III) study glass-box features of transformer MT systems in terms of unsupervised QE. All proposed features can be grouped into three categories: (1) output probability distribution from deterministic neural MT systems; (2) uncertainty quantification based on the Monte Carlo dropout (MC dropout) method (Gal et al., 2016) and (3) attention distribution. The uncertainty-related features computed using MC dropout demonstrate higher performance than the other two groups of features. In supervised settings, the combination of MC dropout and attention-based features show the best results (Fomicheva et al., 2020b, which is our Publication IV). Apart from "pure" glass-box QE models, there are several QE supervised models that use the glass-box features described above or their modifications in combination with black-box features (Wang et al., 2021; Moura et al., 2020).

Today, the best performing QE systems are based on black-box features or their combination with glass-box features. They typically use the predictor-estimator framework as well as model ensembling (Specia et al., 2021). The Predictor-Estimator (Kim et al., 2017) consists of an encoder-decoder RNN as a predictor and a unidirectional RNN as an estimator. During the first step, the predictor

trained on a large amount of parallel corpora predicts a word in the target sentence. Then, the context representations generated by the predictor are used as inputs of a regression layer of the neural quality estimation model within the second step.

Modern predictor-estimator models typically use multilingual pre-trained transformers such as XLM-RoBERTa (Conneau et al., 2020) instead of training a predictor and apply a feed-forward neural network as a estimator rather than a RNN (Wang et al., 2021; Zerva et al., 2021).

Using multilingual pre-trained representation allows researchers to train well-performing models without additional resources. However, these models require a large amount of computational resources, for example, the most efficient QE systems, presented in WMT21[1], have more than 500M parameters (Specia et al., 2021), that may limit their practical use.

Another limitation of QE models built on top of multilingual pre-trained transformers is that these pre-trained transformers do not cover all languages in the world, for example XLM-RoBERTa model produces representations for 100 languages, whereas there are more than 7000 languages in the world.

While black-box models generally outperform glass-box models in predicting translation quality, glass-box models usually require less computational resources and do not require additional data (Specia et al., 2021). This is why they can be more appropriate for lesser-used languages and for quick quality assessment.

2.5.2. Is QE for translators or humans?

The goal of Quality Estimation is to assess how good a translation is without comparing it to a reference sentence. There are two common ways to consider the translation quality. The first way is to measure post-editing (PE) effort and it is more suitable for professional translators. The latter is human Direct Assessment (DA) (Graham et al., 2017) and it is used mainly in everyday situations, for example, to assess the translation quality of online machine translation systems.

Post-editing effort (Snover et al., 2006, HTER) depends on the number of edits that need to be made to get the correct translation. A post-editor has to find the minimum number of shifts, substitutions, insertions and deletions in order to get an adequate translation. As a rule, all edits have equal "weight"; and the ratio between the number of edits and the length of the reference sentence is used. Thus, the higher the HTER value, the more changes need to be made and the lower the value, the better the translation.

Table 1 shows an example of MT output and its post-edited version (PE). There are six edits to be made: five substitutions and one deletion — to get the correct output. The number of tokens in the reference, including punctuation marks, is 24. So, the final HTER score is $6/24 = 0.25$.

[1] Sixth conference on Machine translation: https://www.statmt.org/wmt21/

MT:	McDonald 's , the international **high-speed** food chain operating in Estonia , **has** changed its business **people** in connection with **the exchange** of **owners** .
PE:	McDonald 's , the international fast food chain operating in Estonia , changed its business name in connection with a change of ownership .

Table 1: Machine translation output (MT) and its post-edited version (PE). The example is taken from MLQE-PE Et-En dataset (Fomicheva et al., 2020c). Highlighted words should be replaced or deleted.

Another way to measure the effectiveness of QE systems is using Direct Assessment (DA) (Graham et al., 2017). For example, professional annotators of WMT20 and WMT21 datasets (Fomicheva et al., 2020a; Fomicheva et al., 2020c) evaluated the quality of translation on a continuous 0-100 scale following the FLORES setup (Guzmán et al., 2019):

0-10 an incorrect translation;

11-29 a translation with few correct keywords, but the overall meaning does not preserve;

30-50 a translation with major mistakes;

51-69 a translation is understandable, the overall meaning of the source is translated correctly but with typos or grammatical errors;

70-90 a translation that closely preserves the semantics of the source;

91-100 a perfect translation.

While both metrics measure the quality of the translation, they assess different things; therefore, the linear correlation between them is not always moderate. For example, Pearson correlation coefficients between DA and HTER are -0.17 and -0.55 for En-De and Et-En WMT20 training sets[2], respectively.

(i)	Muutused uurimistöö eeldustes ja maailmapildis ning nendega enamasti kaasnevad vaidlused on teadusrevolutsioonidele iseloomulikud .
MT	Changes in research assumptions and in the world picture , and most of the disputes that accompany them , are symptomatic of scientific revolutions .
PE	Scientific revolutions are characterised by changes in research assumptions and in worldview , as well as the disputes that usually accompany them .
(ii)	Johannes teeb talle teatavaks , et Jaanusel on uus pruut .
MT	Johannes tells him that Jaani has a new carrot .
PE	Johannes tells him that Jaanus has a new girlfriend .

Table 2: The examples of a source, its translation (MT) and its translation's post-edited version (PE) (Fomicheva et al., 2020c, MLQE-PE Et-En dataset).

To more clearly show the difference between these two metrics, we chose two examples with completely opposite values for HTER and DA (Table 2). In the first

[2]HTER and DA scores are available: https://github.com/sheffieldnlp/mlqe-pe

case, the translation sounds a little clumsy, but the meaning is preserved. Therefore, the annotators rated the quality of translation as very good (92.8 out of 100), whereas the HTER value (0.609) indicates that many changes need to be made to get the correct result. The second example demonstrates the opposite situation: the source's meaning is distorted (DA=11.5), while only two substitutions need to be made to get a reasonable translation, which gives us a low HTER score of 0.2.

3. ATTENTION WEIGHTS EXTRACTED FROM RNN-BASED MT SYSTEMS (PUBLICATIONS I AND II)

After Bahdanau et al. (2015) introduced the attention mechanism, models based on RNNs with attention mechanisms quickly became state-of-the-art in machine translation. It enhanced the quality of translations and provided an exciting way to inspect alignments between the source and target tokens.

In this chapter, we explore how we can use attention weights to predict translation quality. The fascinating thing is that we can apply our Quality Estimation approach based on attention weights not only to translations generated by machine translation models with attention mechanisms but also to **any** translations — to translations generated MT systems without the attention mechanism or theoretically to human translations.

To get attention weights for any translation, we replace the decoding part of the neural MT system with computing the probability of the given translation under a neural MT model for this language pair. This way beam search and selecting the output token with the highest predicted probability is replaced by selecting the next given output token; in other words, force-decoding is done. Thus, we can get attention weights for any source-translation pair without even knowing anything about the system that produced the original translation. We call these attention weights force-decoded.

This chapter describes the use of attention weights for both supervised and unsupervised QE tasks. We compare the performance of QE models based on internal (extracted from the neural MT system which produces the translation) and force-decoded attention weights.

3.1. Why do we use attention weights?

Rikters et al. (2017a) propose using attention weights, or more precisely, metrics based on them, for confidence estimation tasks. We extrapolate this approach to assess translation quality, as attention weights represent the strength of connection between the source and output tokens, that may indicate translation quality.

Figure 6 depicts a visualisation of attention distribution between the source and output tokens of a partly good translation. The first part of the sentence "Warnung vor einer unmittelbar drohenden Gefahr" is translated quite well and we see the strong connections between the source and output tokens. The translation of the part "zum Tode oder zu schwersten Verletzungen" is almost missing, which is reflected in Figure 6 as a lack of connections.

Compared to widespread QE approaches of that time (Specia et al., 2013; Martins et al., 2017; Kim et al., 2017), which required additional data, the attention-based method requires only a neural MT system with the attention mechanism and

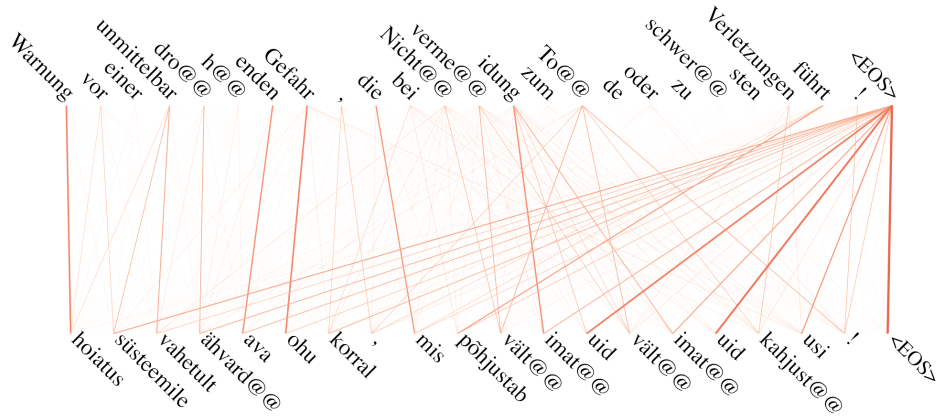


Figure 6: Attention alignment visualization of a partly good translation (Rikters et al., 2017b).

gold labels in the case of supervised tasks.

Metrics

For all of our experiments described in this chapter, we used the following metrics proposed by Rikters et al. (2017a):

- **Coverage Deviation Penalty** (*CDP*) penalizes the sum of attentions per input token, so tokens with less or too much attention get a lower score.

$$CDP = -\frac{1}{J}\sum_{j=1}^{J}\log\left(1+(1-\sum_{i=1}^{I}\alpha_{ji})^2\right),$$

  where α represents attention weights, J is the length of the input sentence and I is the number of target tokens.
- **Entropy or Absentmindedness Penalty** (Ent_{in} and Ent_{out}) computes the dispersion via the entropy of the attention distribution of input and output tokens.

$$Ent_{in} = -\frac{1}{J}\sum_{j=1}^{J}\sum_{i=1}^{I}\alpha_{ij}\cdot\log\alpha_{ij}$$

$$Ent_{out} = -\frac{1}{I}\sum_{i=1}^{I}\sum_{j=1}^{J}\alpha_{ji}\cdot\log\alpha_{ji}$$

- **Total** is a sum of all metrics described above.

$$Total = CDP + Ent_{in} + Ent_{out}$$

In addition to the metrics listed above, we also compute the ratio between input and output entopies:

$$Ent_{ratio} = \frac{Ent_{in}}{Ent_{out}}$$

3.2. QE as a Classification Task (Publication I)

Quality Estimation is typically considered as a regression task. However, there are scenarios in which it makes more sense to use classification, for example, if the goal is to filter out the worst translations.

Data and Settings

We have done this work in close collaboration with Estonian translation agency Grata OÜ. The main goal was to detect the worst translations because they slow down the post-editing process. As we mentioned in Section 2.5.2, we usually take the ratio between the number of edits and the length of the reference sentence to compute post-editing effort (HTER). However, we chose not to normalize HTER scores since denormalized scores correlate better with post-editing time than normalized. In our case, post-editing time is the most important metric.

Since the MT systems generated translations used the attention mechanism, we could build a classification model with internal weights. To compare this model with the model with force-decoded weights, we trained new MT models. As a classifier, we used `sklearn`[1] application of Random Forest (Ho, 1995), all attention-based metrics described above were used as its input features and denormalized HTER scores[2] as labels. To evaluate the performance of the algorithm, we mainly computed the Matthews correlation coefficient (MCC) that works well for unbalanced data:

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}},$$

where TP stands for true positive values, TN — true negative, FP - false positive and FN — false negative.

As data, we had 5444 sentences for German-Estonian and 4541 sentences for English-Estonian. All original English and German sentences were taken from technical texts. We used about 10% of the data as test sets, and the rest was used for training and validation.

Results

Table 3 shows the results for German/English-Estonian language pairs. It can be seen that there is no noticeable difference between the metrics based on internal

[1] https://scikit-learn.org/stable

[2] Post-edits of translations were done by our industrial partner Grata OÜ.

	Supervised		Unsupervised	
	Internal	Force-decoded	Internal	Force-decoded
De-Et	0.593	0.598	0.668	0.467
En-Et	0.659	0.556	0.689	0.412

Table 3: Results of experiments for German-Estonian (De-Et) and English-Estonian (En-Et) language pairs. We used Matthews correlation coefficient for supervised tasks and absolute values of Pearson correlation coefficient for unsupervised tasks.

and force-decoded attention weights for supervised approach.

To see if attention-based metrics can be used as an unsupervised metric, we calculated the Pearson correlation coefficient between the `total` metric and denormalized HTER scores. According to our results (Table 3), internal weights perform better than force-decoded.

3.3. QE as a Regression Task (Publication II)

In this section, we extend the proposed attention-based approach to regression problems and examine its performance for four different language pairs.

Data and Settings

We used the QE Shared Task data from WMT18 (Specia et al., 2018). The organizers provided data for four language pairs: English-Czech (En-Cs), English-German (En-De), English-Latvian (En-Lv) and German-English (De-En). For En-De and En-Lv, translations were produced by statistical machine translation (SMT) systems as well as neural machine translation systems. For De-En and En-Cs, there were translations generated only by SMT systems. Table 4 shows the amount of data for all language pairs.

	EN-DE		DE-EN	EN-CS	EN-LV	
	nmt	smt	smt	smt	nmt	smt
train	13442	26299	26032	40254	12936	11251
dev	1000	1000	1000	1000	1000	1000
test	1023	1926	1254	1920	1448	1315

Table 4: Number of sentences for each language pair and each machine translation system.

Since neural MT systems produced translations were not available to WMT18 participants, we used only force-decoded attention weights. To extract them, we adopted NMT models prepared by the University of Edinburgh (Sennrich et

al., 2017) for English-German, German-English and English-Czech; for English-Latvian, we used a different NMT model trained separately.

We trained regression QE models using Random Forest (Ho, 1995) with a grid search algorithm for the optimization of parameters. Five attention-based metrics described in 3.1 are used as input features of the regressor and HTER values provided by WMT18 organizers as gold-labels. To evaluate models' performance, Pearson correlation coefficients between predicted values and labels were computed.

Results

As shown in Table 5, we obtained a reasonably high linear correlation for German-English and a close-to-moderate/moderate correlation for English-Latvian (NMT). However, QE systems for other language pairs demonstrate lower results. The considerable performance gap can be caused by data belonging to different domains — German-English and English-Latvian data were taken from pharmaceutical texts, and sentences English-German and English-Czech from IT. In addition, attention weights were extracted from general-domain NMT systems without fine-tuning on a specific domain.

	EN-DE		DE-EN	EN-CS	EN-LV	
	smt	nmt	smt	smt	smt	nmt
AttW	0.249	0.219	0.533	0.319	0.323	0.438
QuEst	0.369	0.354	0.220	0.389	0.389	0.462
AttW + QuEst	**0.426**	**0.373**	**0.554**	**0.451**	**0.402**	**0.531**

Table 5: The absolute values of Pearson correlation coefficients between HTER scores and predicted values (`AttW`) for test datasets. `QuEst` (Specia et al., 2013) is the baseline computed by organizers of WMT18.

To investigate how the choice of a neural MT system affects the Pearson correlation between an automatic prediction and post-editing effort, we compared the results of our neural MT system and the University of Edinburgh's MT system for the German-English language pair. The resulting scores differ but not significantly (0.562 and 0.533, respectively). On the one hand, this suggests that the choice of a neural MT system is not essential; on the other hand, both compared MT systems are general-domain models rather than specific-domain.

3.4. Summary

In this chapter, we have examined the first research task — *examination of attention weights extracted from MT systems based on recurrent neural networks (RNNs) as a QE indicator* and have answered to both research questions:

- **RG1-Q1**: *How well do attention-based QE models perform for supervised and unsupervised QE tasks?*

In Publication I, we have shown that unsupervised models with internal weights have a strong linear correlation with post-editing effort. The results of the supervised models, studied in Publication II, vary markedly across language pairs. That may be due to the fact that the data was taken from different domains and we extracted attention weights from out-of-domain neural MT systems.

- **RG1-Q2**: *Can these models be applied to translations produced by any unknown machine translation system?*
  Yes, it is possible to use an external MT system with the attention mechanism to get attention distribution for the translation generated by any MT system. In Publication I, we have compared the performance of the models based on internal and force-decoded attention weights. Although the models based on external attention weights demonstrate a moderate linear correlation with post-editing effort, they show worse performance than the models using internal attention weights.

It is worth noting, we cannot directly compare the results of the models in Publication I and II, because we used different types of tasks: classification and regression, and hence different metrics.

4. ATTENTION WEIGHTS EXTRACTED FROM TRANSFORMER-BASED MT SYSTEMS (PUBLICATIONS III-V)

In the previous chapter, we discussed attention weights obtained from RNN-based machine translation models, however in 2017 a new architecture Transformer (Vaswani et al., 2017) quickly supplanted RNN models and became state-of-the-art. For this reason, the use of QE models based on attention weights derived from RNN models with attention mechanism has become deprecated. However, the idea of using attention weights can be applied to the new architecture as well.

As we mentioned earlier, there are two different attention mechanisms in Transformer: self-attention and encoder-decoder attention. In our experiments, we only work with the encoder-decoder attention mechanism because it "binds" the source and output sentences together. It is similar to the attention mechanism used in RNN, but, by comparison, it consists of several attention matrices. Encoder-decoder attention matrices are computed for each head (H) and layer (L) of the decoder; as a result, there are $[H \times L]$ matrices. The main question arises — how to handle $[H \times L]$ matrices.

4.1. Unsupervised and semi-supervised approaches for DA (Publication III)

Metrics

We proposed two unsupervised ways to summarize information from all layers and heads:

- the minimum entropy across all computed entropies:

$$\text{Ent-Min} = min_{\{hl\}}(\text{Att-Ent}_{\text{hl}})$$

- the average of all computed entropies:

$$\text{Ent-Avg} = \frac{1}{H \times L} \sum_{h=1}^{H} \sum_{l=1}^{L} \text{Att-Ent}_{\text{hl}},$$

where *Att-Ent* is the entropy of the attention distribution as known as AP_{out} from the previous chapter:

$$\text{Att-Ent} = -\frac{1}{I} \sum_{i=1}^{I} \sum_{j=1}^{J} \alpha_{ji} \log \alpha_{ji}$$

and one semi-supervised:

- "oracle": figure out the layer/head combination that shows the better performance, for example, using a small annotated dataset.

Data and settings

In our experiments, we used 1000 sentences taken from six language pairs: English-German (En-De), English-Chinese (En-Zh), Romanian-English (Ro-En), Estonian-English (Et-En), Sinhala-English (Si-En) and Nepali-English (Ne-En). To evaluate the performance of the proposed metrics, we calculated Pearson correlation coefficients between them and DA scores[1] (in contrast to the previous chapter, where we worked with HTER scores).

The Transformer architecture was used to train all MT models; since the most important parameters for our study are the number of heads and layers, we omit other training parameters. For all languages except Si-En and Ne-En, the MT models were trained based on the standard six-layer, eight-head Transformer architecture; for Si-En and Ne-En used Big-Transformer with six layers and 16 heads. The presence of neural MT systems allowed us to extract the internal weights of attention.

	et-en	ro-en	si-en	ne-en	en-de	en-zh
Ent-Min	0.329	**0.524**	0.097	0.265	0.000	0.067
Ent-Avg	0.377	0.382	0.10	0.205	0.090	0.112
best layer/head	**0.416**	**0.636**	**0.255**	**0.381**	**0.241**	**0.168**

Table 6: Absolute values of Pearson correlation coefficients between unsupervised metrics and human DA judgments for 1000 sentences. Results that are not significantly outperformed by any method are marked in bold.

Results and Discussion

As shown in Table 6, the performance of the metrics varies significantly across languages, from no linear correlation to moderate. There are several plausible explanations for this. First of all, there is no direct mapping between words in different languages, so low entropy does not necessarily indicate a high quality of translation. Secondly, simply combining information from different heads and layers may not be the optimal solution.

Voita et al. (2019) show that different heads of attention are responsible for different functions. To study the behaviour of attention heads in more detail from QE perspective, we calculated Pearson correlation coefficients between human DA judgments and `Att-Ent` for all heads. The best layer-head combination ("oracle") is superior to summarized metrics, `Ent-Min` and `Ent-Avg`, in predicting translation quality in almost all language pairs (Table 6). While its use gives the best results, it requires validation on an annotated dataset and can be considered as a semi-supervised approach.

[1]Original sentences, their translations and DA scores can be found https://github.com/facebookresearch/mlqe

	Head1	Head2	Head3	Head4	Head5	Head6	Head7	Head8
Layer1	0.307	0.209	0.233	0.286	0.320	0.267	0.304	0.336
Layer2	0.314	0.416	0.306	0.185	0.275	0.227	0.277	0.306
Layer3	0.317	0.332	0.343	0.332	0.327	0.340	0.356	0.361
Layer4	0.146	0.364	0.378	0.298	0.283	0.299	0.344	0.274
Layer5	0.181	0.179	0.311	0.335	0.353	0.219	0.168	0.267
Layer6	0.163	0.246	0.179	0.239	0.220	0.252	0.197	0.188

Table 7: Absolute values of Pearson correlation coefficients between entropy and human DA judgments for Estonian-English (1000 sentences).

	Head1	Head2	Head3	Head4	Head5	Head6	Head7	Head8
Layer 1	0.078	0.124	0.073	0.188	0.152	0.183	0.266	0.598
Layer 2	0.011*	0.214	0.006*	0.108	0.213	0.115	0.636	0.343
Layer 3	0.142	0.301	0.501	0.202	0.222	0.529	0.067	0.406
Layer 4	0.196	0.448	0.483	0.449	0.261	0.090	0.393	0.022*
Layer 5	0.428	0.311	0.484	0.195	0.257	0.492	0.439	0.409
Layer 6	0.213	0.173	0.258	0.186	0.490	0.099	0.295	0.259

Table 8: Absolute values of Pearson correlation coefficients between entropy and human DA judgments for Romanian-English (1000 sentences). Correlation coefficients with p-value > 0.05 are marked *

Tables 7 and 8 show computed correlation coefficients for each head as a heatmap for Et-En and Ro-En, with darker color indicating better performance. We can see that correlation differs noticeably across the heads for both language pairs, for example, from 0.146 to 0.416 for Et-En. Although for both language pairs the highest correlation coefficients come from the second layer, we need to run more experiments to extend this knowledge to other languages.

4.2. Supervised approaches for DA and HTER (Publications IV and V)

In the previous section, we discussed the use of attention weights of Transformer MT models to predict translation quality in unsupervised or semi-supervised ways. Compared to the models using internal attention weights of RNN-based MT models (Section 3.2), the performance of QE models based on Transformer attention weights is lower and varies significantly across language pairs.

In this section, we examine the behaviour of supervised models based on Transformer attention weights using not only summarized information, as in the first publications, but also "pure" attention weights. In addition, for the first time, we have DA and HTER scores for the same data, which allowed us to compare the performance of the proposed models for both metrics.

Data and Settings

Data We focused on two language pairs: Estonian-English (Et-En), as the language pair performed relatively well on the unsupervised task, and English-German (En-De), which performed the worst. For our supervised methods, we used 7 000 sentences for the training dataset, 1 000 sentences for the development set and two test datasets of 1 000 sentences each[2].

Models with entropies as input features To run experiments with the summarized information — in other words, with entropies (`Ent-Mod`), — we chose Random Forest (Ho, 1995), as a relatively fast approach, and an ensemble building based on (Caruana et al., 2004). To run experiments with Random Forest models (`RF-Ent`), we used the `sklearn`[3], set a randomized search on the hyperparameters and performed 5-fold cross-validation. To conduct tests with ensemble building models (`Ens-Ent`), we used the `mljar`[4], Random Forest and CatBoost (Prokhorenkova et al., 2018), set Pearson as the evaluation metric and ran 5-fold cross-validation.

Convolutional Neural Network -based models The summarization of attention weights into metrics is already, to a certain extent, their interpretation. Is it a necessary step? To test it, we used attention weights as input features for a relatively simple convolutional neural network (CNN).

The base architecture of proposed CNN models is presented in Figure 7. The model's input is attention weights with a shape (`[Heads`$\times$`Layers]`, number of the source tokens, number of the target tokens). The number of `[Heads`$\times$`Layers]`[5] is constant for all weights obtained from the same system, whereas the number of source and target tokens of each sentence can vary noticeably. To reduce the amount of padding added to each batch, we sort all sentences by the number of

[2] All data can be found http://www.statmt.org/wmt21/quality-estimation-task.html

[3] https://scikit-learn.org/stable

[4] https://supervised.mljar.com/

[5] 8×6 for En-Et and En-De neural MT systems

source/target tokens (max(src, tgt)) and only after that form a batch. Each CNN-based model consists of two or three CNN blocks, each of them comprises 2D-CNN, Batch Normalization, MaxPooling and Dropout Layers. We use `Relu` as the activation function. To handle the variable size of input batches, we use the Adaptive Max pooling layer. The last block of the model consists of three feed-forward layers. As a result, the model is trained to produce the desirable score: DA or HTER. We optimised our neural models with Adam (Kingma et al., 2015).

We ran experiments with several models; the main difference between them is the type of gold labels used — human-annotated or synthetic.

To predict DA scores, we considered three models:

`CNN-DA`: we use DA human-annotated data: 7 000 for the training dataset and 1 000 for the development;

`CNN-BLEURT`: we experiment with pre-training on synthetic data and for that we compute the BLEURT (Sellam et al., 2020) score for randomly chosen 300 000 sentences[6] and use them as labels for the training and development datasets. We have chosen BLEURT to get artificial labels due to its good agreement with human judgments (Mathur et al., 2020);

`CNN-BLEURT+`: we fine-tune the model `CNN-BLEURT` on human-annotated data.

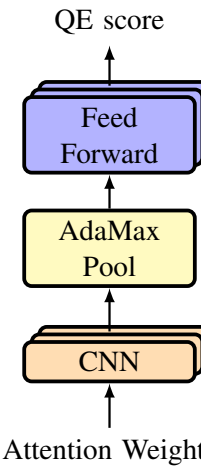


Figure 7: The architecture of the CNN-QE models.

To predict HTER scores, we presented two models:

`CNN-HTER`: we train a model with HTER human-annotated data;

`CNN-HTERart`: we use synthetically computed HTER between the translations and their references. Though the preliminary experiments demonstrated a poor performance compared to `CNN-HTER`, but this setting can be used in the absence of the human annotated training data.

Results

Results of DA-predicting models As expected, the supervised attention-based models predicting DA outperform unsupervised ones (see Table 9). Both summarizing and non-summarizing methods demonstrate nearly moderate to moderate linear correlation. It is worth pointing out that the performance of CNN-BLEURT,

[6] Additional parallel corpora to train CNN models: the OpenSubtitles (Lison et al., 2016), JRC Acquis (Steinberger et al., 2006), EuroParl (Koehn, 2005), DGT and EMEA (Tiedemann, 2012) corpora

the model that does not require expensive and time-consuming human labels, is relatively high.

	En-De		Et-En	
	dataset1	dataset2	dataset1	dataset2
RF-Ent	0.373	0.301	0.499	0.455
Ens-Ent	**0.395**	0.341	0.517	0.480
CNN-DA	0.220	0.210	0.518	0.464
CNN-BLEURT	0.383	0.357	0.577	0.526
CNN-BLEURT+	0.381	**0.369**	**0.599**	**0.547**
Ent-Min	0.000	**	0.329	**
Ent-Avg	0.090	**	0.377	**
best head/layer	0.241	**	0.416	**

Table 9: The absolute value of Pearson correlation coefficients between human DA scores and predicted values for both test datasets. **We did not compute correlations between true labels and unsupervised predicted values for the second dataset.

To measure how predicted outputs of entropy-based and CNN-based models are correlated, we computed Pearson correlation coefficients for dataset2's predictions. As expected, we got a moderate correlation (0.36-0.46) for En-De and a strong correlation (0.61-0.75) for Et-En. As shown in Table 10, averaging outputs of both, entropy- and CNN-based, models results in better performance than the individual model.

	En-De	Et-En
Ens-Ent + CNN-DA	0.344	0.517
Ens-Ent + CNN-BLEURT	0.409	0.547
Ens-Ent + CNN-BLEURT+	0.416	0.561

Table 10: The absolute value of Pearson correlation coefficients between human DA scores and predicted values for the second dataset.

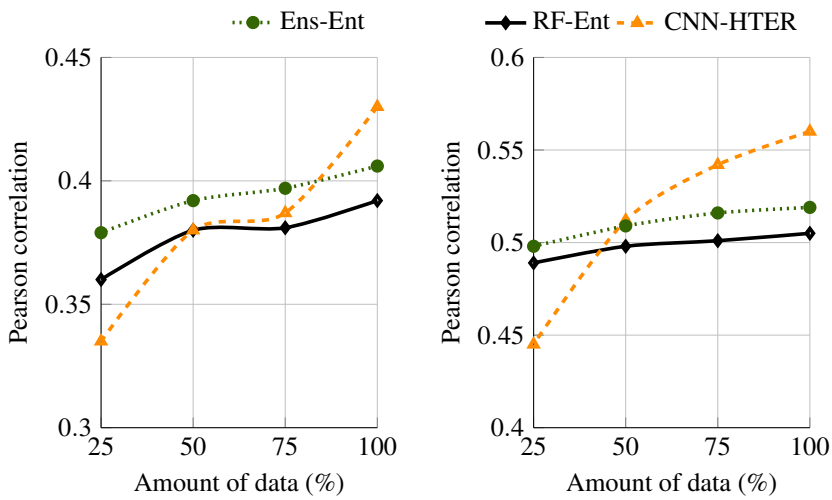
Results of HTER-predicting models Like DA-predicted models, all proposed HTER-models perform well (see Table 11), showing a moderate linear correlation for both language pairs and outperforming unsupervised metrics.

How many sentences do we need to train a QE model? Getting DA as well as HTER scores is a time-consuming and expensive task, so the less annotated data required, the better. To examine how much labelled data is needed to train models, we ran ten tests and averaged the obtained correlation coefficients for each examined amount of data (25%, 50%, 75%). According to our experiments, both discussed approaches, Ent-Mod and CNN-HTER/DA, demonstrate comparable higher performance even with a small amount of training/validation data. As shown in Figure 8, the performance of Ent-Mod models for En-De (left) and Et-En (right) language pairs worsens slightly as the amount of training data

	En-De		Et-En	
	dataset1	dataset2	dataset1	dataset2
RF-Ent	0.389	0.519	0.505	0.534
Ens-Ent	0.408	**0.531**	0.519	**0.561**
CNN-HTER	**0.430**	0.503	**0.580**	0.549
CNN-HTERart	0.334	**	0.482	**
Ent-Min	0.000	**	0.386	**
Ent-Avg	0.000	**	0.289	**
best head/layer	0.269	**	0.407	**

Table 11: The absolute value of Pearson correlation coefficients between HTER scores and predicted values for both test sets. **We did not compute correlations between true labels and CNN-HTERart/unsupervised predicted values for the second dataset.

decreases. The performance of CNN-HTER models declines more markedly, but remains relatively high. All models show a moderate linear correlation with post-editing effort, especially in the case of the En-Et language pair, even using 2000 training / validation examples (1750 for training and 250 for validation).



(a) En-De (b) Et-En

Figure 8: Absolute value of Pearson correlation coefficient between predicted values of the first dataset and HTER scores for En-De and Et-En language pairs.

Does combining several glass-box features improve performance? One way to improve performance of glass-box QE models is to use several glass-box features together. In collaboration with our colleagues at the University of Sheffield, we examined the behavior of QE models based on three types of features: (1) attention weights; (2) the output probability distribution from a deterministic neural MT system (`Probability` features) and (3) output probability

distribution with use uncertainty quantification based on the Monte Carlo dropout method (`Dropout` features)[7].

		Et-En	Ro-En	Si-En	Ne-En	En-De	En-Zh
Type of features	`Attention`	0.519	0.722	0.455	0.583	0.382	0.353
	`Dropout`	**0.669**	0.751	0.548	0.638	0.206	0.352
	`Probability`	0.525	0.670	0.508	0.568	0.189	0.329
	`Dropout+Probability`	**0.670**	0.754	**0.556**	0.632	0.194	0.381
	`Attention+Probability`	0.611	0.700	**0.550**	0.629	**0.454**	0.406
	`Attention+Dropout`	**0.679**	**0.791**	**0.554**	**0.659**	**0.452**	**0.429**
	All	**0.678**	**0.793**	**0.556**	**0.657**	**0.464**	**0.427**

Table 12: Pearson correlation coefficients between human DA scores and predicted values for the first dataset. Results marked in bold are not significantly outperformed by any other method (We use the Hotelling-Williams test to compute significance of the difference between dependent correlations (Williams, 1959) with p-value < 0.05.

In our experiments with several glass-box features, we used language pairs described in Section 4.1 with 7 000 sentences for training set, 1 000 for development set and 1 000 for test set[8]. We trained models with different combinations of the above groups of features using XG-Boost (Chen et al., 2016) from `xgboost`[9] package. As can be seen from Table 12, the best results among the individual groups of features are obtained for either Dropout features (Et/Ro/Si/Ne-En and En-Zh) or Attention features (En-De/Zh). The combination of all three groups of features and the combination of Dropout and Attention show the best results for all language pairs. Thus, while QE models based on a single metric may not perform well, a combination of different metrics results in much better performance.

4.3. Summary

In this chapter, we have discussed the second research task — *exploration of the attention distribution extracted from transformer-based MT systems in terms of translation quality* and have answered to following research questions:

- **RG2-Q1**: *What are the possible solutions to overcome the difficulties encountered when working with attention weights of transformer MT systems?* Transformer's encoder-decoder attention mechanism consists of several attention matrices. So, the main difficulty arises — how to handle them? In Publication III-V, we have proposed to compute entropy of each matrices. In the case of supervised tasks, all computed entropies are used as input

[7] More details about Probability and Dropout features can be found in Publications III and IV.

[8] Sentences, DA labels and MT models are available: https://statmt.org/wmt20/quality-estimation-task.html

[9] https://xgboost.readthedocs.io/en/latest/python/index.html

features of a regression model. Regarding unsupervised tasks, we have suggested to calculate the minimum entropy across all computed entropies and the average of all computed entropies. In Publication V, we have also proposed to use attention matrices as an input of convolutional neural networks for supervised tasks.

- **RG2-Q2**: *How well do attention-based QE models perform for unsupervised and supervised QE tasks, including "zero-shot" supervised models requiring only synthetic data for labels?*

  The performance of unsupervised models varies significantly across language pairs, so it might be more practical and safe to use semi-supervised or supervised models. All supervised models considered above are superior in efficiency to unsupervised and semi-supervised ones, but also differs across language pairs. Supervised models work relatively well even with synthetic labels.

- **RG2-Q3**: *How much annotated data is needed to train attention-based supervised models?*

  Both, CNN-based and entropy-based, supervised models show comparable higher performance with relatively small amounts of human-annotated data. However, the performance of these models degrades as the amount of training data decreases.

- **RG2-Q4**: *What are the limitations of the attention-based approach and how can they be overcome?*

  As mentioned above, the performance of the proposed models differs notably across language pairs. In the case of unsupervised models, the linear correlation between entropies and human labels varies remarkably across all heads, that is why using the minimum or average entropy may not be the optimal solution. As we have not found the optimal solution, we have proposed to use a semi-supervised approach — using a small annotated dataset to help identify the head that shows the better performance. To improve performance of supervised models, we have suggested to use the combination of several glass-box features.

5. CONCLUSION

Machine translation has become a part of the life of not only linguists and professional translators, but almost everyone. Most people who have used machine translation have come across funny and sometimes completely incorrect translations that turn the meaning of a sentence upside down. Thus, in addition to translation, we need to use a scoring mechanism that informs people about translation quality.

The most prevalent QE approaches are based on black-box features because they provide the best performance. However, methods based on glass-box features can be interesting not only for researchers, but can also prove fruitful in several specific scenarios. For example, when additional data is not available or is difficult to collect, or when we need to reduce the computational resources required to train a QE model.

In this work, we examined attention weights of neural MT systems in terms of translation quality. Initially, the PhD thesis tackled the first research goal *"examination of attention weights extracted from MT systems based on recurrent neural networks (RNNs) as a QE indicator"*. In Chapter 3, we examined the performance of attention-based QE models in unsupervised and supervised ways. We have shown that these models can be used when translations produced by any MT system. It was noticed that the performance of the proposed models differs markedly across language pairs.

When transformer-based MT systems have become state-of-the-art, we examined *the attention distribution extracted from transformer-based MT systems in terms of quality estimation* (the second research goal). Unlike the attention mechanism applied in RNN-based systems, where there is only one attention matrix, there are several attention matrices in systems based on transformers. In Chapter 4, we have proposed two ways how to handle transformer's attention matrices: (1) to compute entropy and (2) to use all attention matrices as an input of convolutional neural networks (CNNs). The first approach can be used for supervised as well as unsupervised tasks, while the second one is only suitable for supervised tasks. We have shown that all proposed supervised models require a small amount of training data. We have also demonstrated that CNN-based approach can be used in a zero-shot setting when human-labelled data is not available. There is the same issue with the transformer's attention-based approaches—the performance varies significantly across language pairs. One possible way to mitigate it and improve performance is to combine attention weights with other glass-box features.

Multilingual machine translation systems are becoming more and more popular. Therefore, it would be interesting to explore how glass-box approaches work for them. Could we transfer a model trained for one language pair of a multilingual MT system to other language pairs of the same system? Especially if we are talking about a zero-shot MT system when the system has explicitly not seen some language pairs.

In this work, we have explored only the quality of translation at the sentence level. We believe that attention-based approaches can be easily adapted to word-level tasks as well as critical error detection problems. Although, some particular difficulties might arise, for example, how we should handle English phrasal verbs, the meaning of which depends heavily on the preposition of the verb, or German verbs with prefixes where the meaning relies on the prefix.

In conclusion, translation quality is an important part of the machine translation pipeline. Although there are already several quality estimation systems showing impressive performance, so far only for some language pairs, and at the same time requiring quite a lot of resources; therefore, there is still a lot of work to be done.

BIBLIOGRAPHY

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

Schuster, M., & Paliwal, K. K. (1997). Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, *45*(11), 2673–2681. https://doi.org/10.1109/78.650093

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1985). *Learning internal representations by error propagation* (tech. rep.). California Univ San Diego La Jolla Inst for Cognitive Science.

Bahdanau, D., Cho, K., & Bengio, Y. (2015). Neural machine translation by jointly learning to align and translate. *CoRR*, *abs/1409.0473*.

Cho, K., van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning phrase representations using RNN encoder–decoder for statistical machine translation. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, 1724–1734. https://doi.org/10.3115/v1/D14-1179

Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, & K. Q. Weinberger (Eds.), *Advances in neural information processing systems*. Curran Associates, Inc.

Wu, Y., Schuster, M., Chen, Z., Le, Q. V., Norouzi, M., Macherey, W., Krikun, M., Cao, Y., Gao, Q., Macherey, K., Klingner, J., Shah, A., Johnson, M., Liu, X., Kaiser, Ł., Gouws, S., Kato, Y., Kudo, T., Kazawa, H., … Dean, J. (2016). Google's neural machine translation system: Bridging the gap between human and machine translation. *CoRR*, *abs/1609.08144*.

Snover, M., Dorr, B., Schwartz, R., Micciulla, L., & Makhoul, J. (2006). A study of translation edit rate with targeted human annotation. *Proceedings of association for machine translation in the Americas*, *200*(6).

Graham, Y., Baldwin, T., Moffat, A., & Zobel, J. (2017). Can machine translation systems be evaluated by the crowd alone. *Natural Language Engineering*, *23*(1), 3–30. https://doi.org/10.1017/S1351324915000339

Graham, Y., Baldwin, T., & Mathur, N. (2015). Accurate evaluation of segment-level machine translation metrics. *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 1183–1191. https://doi.org/10.3115/v1/N15-1124

Guzmán, F., Chen, P.-J., Ott, M., Pino, J., Lample, G., Koehn, P., Chaudhary, V., & Ranzato, M. (2019). Two new evaluation datasets for low-resource machine translation: Nepali-english and sinhala-english. *Proceedingsof the 2019 Conference on Empirical Meth-ods in Natural Language Processing and the9th International Joint Conference on Natu-ral Language Processing (EMNLP-IJCNLP)*, 6097–6110.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, 5998–6008.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Ba, J., Kiros, J. R., & Hinton, G. E. (2016). Layer normalization. *ArXiv abs/1607.06450*.

LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., & Jackel, L. D. (1989). Backpropagation applied to handwritten zip code recognition. *Neural Computation*, *1*(4), 541–551. https://doi.org/10.1162/neco.1989.1.4.541

Rikters, M., & Fishel, M. (2017a). Confidence through attention. *Proceedings of MT Summit XVI*, 299–311.

Ho, T. K. (1995). Random decision forests. *Document analysis and recognition, 1995., proceedings of the third international conference on*, *1*, 278–282.

Sennrich, R., Birch, A., Currey, A., Germann, U., Haddow, B., Heafield, K., Miceli Barone, A. V., & Williams, P. (2017). The University of Edinburgh's Neural MT Systems for WMT17. *Proceedings of the Second Conference on Machine Translation, Volume 2: Shared Task Papers*.

Specia, L., Blain, F., Logacheva, V., Astudillo, R. F., & Martins, A. (2018a). Findings of the wmt 2018 shared task on quality estimation. *Proceedings of the Third Conference on Machine Translation, Volume 2: Shared Task Papers*.

Rikters, M., Fishel, M., & Bojar, O. (2017b). Visualizing Neural Machine Translation Attention and Confidence. *The Prague Bulletin of Mathematical Linguistics*, *109*, 1–12.

Voita, E., Talbot, D., Moiseev, F., Sennrich, R., & Titov, I. (2019). Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 5797–5808.

Caruana, R., Niculescu-Mizil, A., Crew, G., & Ksikes, A. (2004). Ensemble selection from libraries of models. *Proceedings of the twenty-first international conference on Machine learning*, 18.

Sellam, T., Das, D., & Parikh, A. P. (2020). Bleurt: Learning robust metrics for text generation. *Proceedings of ACL*.

Mathur, N., Wei, J., Freitag, M., Ma, Q., & Bojar, O. (2020). Results of the wmt20 metrics shared task. *Proceedings of the Fifth Conference on Machine Translation*, 688–725.

Specia, L., Logacheva, V., Blain, F., Fernandez, R., & Martins, A. (2018b). WMT18 quality estimation shared task training and development data. http://hdl.handle.net/11372/LRT-2619

Blatz, J., Fitzgerald, E., Foster, G., Gandrabur, S., Goutte, C., Kulesza, A., Sanchis, A., & Ueffing, N. (2004). Confidence Estimation for Machine Translation. *COLING 2004: Proceedings of the 20th International Conference on Computational Linguistics*, 315–321.

Specia, L., Shah, K., De Souza, J. G. C., & Cohn, T. (2013). QuEst-A Translation Quality Estimation Framework. *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, 79–84.

Martins, A. F. T., Kepler, F., & Monteiro, J. (2017). Unbabel's participation in the wmt17 translation quality estimation shared task. *Proceedings of the Second Conference on Machine Translation, Volume 2: Shared Task Papers*, 569–574.

Kim, H., Lee, J.-H., & Na, S.-H. (2017). Predictor-estimator using multilevel task learning with stack propagation for neural quality estimation. *Proceedings of the Second Conference on Machine Translation, Volume 2: Shared Task Papers*, 562–568.

Specia, L., Blain, F., Fomicheva, M., Zerva, C., Li, Z., Chaudhary, V., & Martins, A. F. T. (2021). Findings of the WMT 2021 shared task on quality estimation. *Proceedings of the Sixth Conference on Machine Translation*, 684–725. https://aclanthology.org/2021.wmt-1.71

Conneau, A., Khandelwal, K., Goyal, N., Chaudhary, V., Wenzek, G., Guzmán, F., Grave, E., Ott, M., Zettlemoyer, L., & Stoyanov, V. (2020). Unsupervised cross-lingual representation learning at scale. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 8440–8451. https://doi.org/10.18653/v1/2020.acl-main.747

Williams, E. J. (1959). *Regression Analysis* (Vol. 14). Wiley.

Wang, J., Wang, K., Chen, B., Zhao, Y., Luo, W., & Zhang, Y. (2021). QEMind: Alibaba's submission to the WMT21 quality estimation shared task. *Proceedings of the Sixth Conference on Machine Translation*, 948–954. https://aclanthology.org/2021.wmt-1.100

Zerva, C., van Stigt, D., Rei, R., Farinha, A. C., Ramos, P., C. de Souza, J. G., Glushkova, T., Vera, M., Kepler, F., & Martins, A. F. T. (2021). IST-unbabel 2021 submission for the quality estimation shared task. *Proceedings of the Sixth Conference on Machine Translation*, 961–972. https://aclanthology.org/2021.wmt-1.102

Gal, Y., & Ghahramani, Z. (2016). Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning. *International Conference on Machine Learning*, 1050–1059.

Moura, J., Vera, M., van Stigt, D., Kepler, F., & Martins, A. F. (2020). Ist-unbabel participation in the wmt20 quality estimation shared task. *Proceedings of the Fifth Conference on Machine Translation*, 1029–1036.

Fomicheva, M., Sun, S., Yankovskaya, L., Blain, F., Guzmán, F., Fishel, M., Aletras, N., Chaudhary, V., & Specia, L. (2020a). Unsupervised Quality Es-

timation for Neural Machine Translation. *Transactions of the Association for Computational Linguistics*, *8*, 539–555. https://doi.org/10.1162/tacl_a_00330

Fomicheva, M., Sun, S., Yankovskaya, L., Blain, F., Chaudhary, V., Fishel, M., Guzmán, F., & Specia, L. (2020b). BERGAMOT-LATTE Submissions for the WMT20 Quality Estimation Shared Task. *Proceedings of the Fifth Conference on Machine Translation*, 1010–1017. https://aclanthology.org/2020.wmt-1.116

Fomicheva, M., Sun, S., Fonseca, E. R., Blain, F., Chaudhary, V., Guzmán, F., Lopatina, N., Specia, L., & Martins, A. F. T. (2020c). MLQE-PE: A multilingual quality estimation and post-editing dataset. *CoRR*, *abs/2010.04480*. https://arxiv.org/abs/2010.04480

Prokhorenkova, L., Gusev, G., Vorobev, A., Dorogush, A. V., & Gulin, A. (2018). Catboost: Unbiased boosting with categorical features. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, & R. Garnett (Eds.), *Advances in neural information processing systems*. Curran Associates, Inc.

Koehn, P. (2005). Europarl: A parallel corpus for statistical machine translation. *MT summit*, *5*, 79–86.

Steinberger, R., Pouliquen, B., Widiger, A., Ignat, C., Erjavec, T., Tufiş, D., & Varga, D. (2006). The JRC-Acquis: A multilingual aligned parallel corpus with 20+ languages. *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06)*.

Tiedemann, J. (2012). Parallel data, tools and interfaces in opus. In N. C. (Chair), K. Choukri, T. Declerck, M. U. Dogan, B. Maegaard, J. Mariani, J. Odijk, & S. Piperidis (Eds.), *Proceedings of the eight international conference on language resources and evaluation (lrec'12)*. European Language Resources Association (ELRA).

Lison, P., & Tiedemann, J. (2016). Opensubtitles2016: Extracting large parallel corpora from movie and tv subtitles.

Chen, T., & Guestrin, C. (2016). Xgboost: A scalable tree boosting system. *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, 785–794.

Kingma, D. P., & Ba, L. (2015). J. adam: A method for stochastic optimization. *International Conference on Learning Representations*.

ACKNOWLEDGEMENTS

I wrote and rewrote this part of my thesis several times, but each time I could not find the right words, and the list of people to whom I am deeply grateful is so immense that it takes up more pages than the entire dissertation.

So, everyone — with whom I worked, who read my thesis and gave feedback, who walked, went hiking, climbed, or traveled with me, who wrote and talked to me, who asked me how I am feeling, who helped me wade through the bureaucracy — you helped me complete this thesis. Without your support, I would not have been able to complete my work or even to start it. Thank you!

SISUKOKKUVÕTE

Kvaliteedi hindamine tähelepanu abil

Viimastel aastatel on masintõlkesüsteemide (MT) kasutamine järsult kasvanud. Tänapäeval ei kasuta masintõlget mitte ainult suurettevõtted, riigiasutused ja tõlkebürood, vaid ka inimesed, kes tahavad näiteks teada, millest nende lemmiklaul räägib. Närvivõrgu põhiste mudelite tulekuga on masintõlkesüsteemid teinud olulisi edusamme, saavutades kõrge ressursiga keelepaaride puhul inimlähedase kvaliteedi. Tõlke kvaliteet ei ole aga keelepaaride, domeenide, andmekogumite ja isegi sama MT-süsteemi puhul järjepidev. See on eriti problemaatiline vähemate ressurssidega stsenaariumide puhul, kus treenimisandmeid ei ole piisavalt ja tõlke kvaliteet jääb märkimisväärselt maha. Lisaks loovad kaasaegsed masintõlkesüsteemid tavaliselt ladusaid tõlkeid, kuid mõned neist tõlgetest võivad olulised üksikasjad vahele jätta või originaallause täiesti valesti esitada. Seega peame hindama iga süsteemi tõlget, et tõlge ei moonutaks algse lause tähendust.

Tõlkebüroode puhul toimetavad masintõlke tulemusi professionaalsed tõlkijad. Mõne stsenaariumi korral, näiteks veebipõhiste masintõlkesüsteemide puhul, ei ole aga võimalik tõlke kvaliteeti inimtoimetajate abiga hinnata. Seetõttu on automatiseeritud tõlkekvaliteedi hindamise süsteemid masintõlke töövoo oluline osa.

Tõlkekvaliteedi hindamiseks on olemas kahte tüüpi automatiseeritud süsteeme: võrdlustõlge(te)ga ja ilma. Esimesi nimetatakse sageli mõõdikuteks või etalonipõhisteks mõõdikuteks; teisi nimetatakse kvaliteedihinnangu (quality estimation, QE) mõõdikuteks. Tavaliselt kasutame MT-süsteemide treenimisel masintõlke väljundi kvaliteedi hindamiseks etalonipõhiseid mõõdikuid, samas aga saame kvaliteedihinnangu mõõdikuid kasutada otse ka veebipõhiste MT-süsteemide tõlkekvaliteedi mõõtmiseks, ilma etalontõlgeteta.

Selles doktoritöös keskendume kvaliteedihinnangu mõõdikutele ja käsitleme tõlke kvaliteedi näitajana tähelepanumehhanismi ennustatud jaotusi, mis on üks kaasaegsete neuromasintõlke (NMT) süsteemide sisemistest parameetritest. Kõigepealt rakendame seda rekurrentsetel närvivõrkudel (RNN) põhinevate NMT-süsteemide genereeritud tõlgetele (Publikatsioonid I ja II). Näitame, et pakutud mudelid võivad töötada nii juhendatud kui ka juhendamata viisil rakendatuna. Juhendatud mudelite peamiseks puuduseks on annoteeritud inimandmete kasutamine, kuna professionaalsete tõlkijate poolt andmete märgistamine on aeganõudev ja kulukas ülesanne. Seetõttu on oluline arendada ka juhendamata lähenemisi. Lisaks demonstreerime, et antud lähenemisviis on rakendatav tõlgetele, mis on tehtud mistahes tundmatu masintõlkesüsteemiga. Uurime pakutud mudeli tulemuslikkust erinevates keelepaarides, nagu eesti-saksa, inglise-saksa, inglise-läti, ja erinevates domeenides (IT, tehniline ja farmakoloogiline).

Kuna RNN-põhised MT-süsteemid on nüüdseks asendunud transformeritega, mis muutusid peamiseks tipptaseme masintõlke tehnoloogiaks, kohandasime oma

lähenemisviisi ka transformeri arhitektuurile. Uurime, kuidas see toimib juhendamata (Publikatsioon III), osaliselt juhendatud (Publikatsioon III) ja juhendatud (Publikatsioonid IV-V) masinõppe ülesannetes. Juhendatud treenimise puhul uurime, kui palju annoteeritud andmeid on vaja kvaliteedihinnangu mudeli treenimiseks, ja näitame, et need nõuavad tegelikult väikest kogust treenimese andmeid. Sellele lisaks näitame, et juhendatud masinõppe mudelid saavutavad mõistliku korrelatsiooni inimeste hinnangutega isegi sünteetiliste märgistatud andmete kasutamisel. Demonstreerime, et parimaid tulemusi saavutatakse tähelepanukaalude ja muude MT mudelist saadud tunnuste kombineerimisel (Publikatsioon IV). Nagu kahes eelmises publikatsioonis, testime oma lähenemist erinevates keeltes — eesti-inglise ja inglise-saksa (juhendamata, osaliselt juhendatud ja juhendatud mudelid), inglise-hiina, nepali-inglise, rumeenia-inglise ja singali-inlglise (osaliselt juhendatud ja juhendatud mudelid).

Me rakendame oma lähenemisviisi ainult lausetaseme kvaliteedi hindamisel, kuid seda võiks kohandada kvaliteedikontrolli jaoks muudel tasanditel, näiteks sõna, fraasi või dokumendi jaoks.

PUBLICATIONS

CURRICULUM VITAE

Personal data

Name: Elizaveta (Lisa) Yankovskaya
Contact: lisa.yankovskaya@ut.ee

Education

2017–... University of Tartu, Institute of Computer Science, PhD studies

2015–2017 University of Tartu, Institute of Computer Science, Master's studies

Employment

2019–... Junior Research Fellow in Natural Language Processing, Institute of Computer Science, University of Tartu

Teaching

- Teaching Assistant for Machine Learning course at University of Tartu in fall 2020;
- Leading Research Seminar in Computational Humanities at University of Tartu in spring 2021

Scientific work

Main fields of interest:

- machine learning
- natural language processing
- data science

Publications (related to NLP)

2021 "Direct Exploitation of Attention Weights for Translation Quality Estimation", Lisa Yankovskaya and Mark Fishel, *Proceedings of the Sixth Conference on Machine Translation*, pp. 955-960, 2021

2021 “Backtranslation Feedback Improves User Confidence in MT, Not Quality”, Vilém Zouhar, Michal Novák, Matús Zilinec, Ondrej Bojar, Mateo Obregón, Robin L. Hill, Frédéric Blain, Marina Fomicheva, Lucia Specia and Lisa Yankovskaya, *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 151–161, 2021

2020 “BERGAMOT-LATTE Submissions for the WMT20 Quality Estimation Shared Task”, Marina Fomicheva, Shuo Sun, Lisa Yankovskaya, Frédéric Blain, Vishrav Chaudhary, Mark Fishel, Francisco Guzmán and Lucia Specia, *Proceedings of the Fifth Conference on Machine Translation*, pp.1010-1017, 2020

2020 “Unsupervised Quality Estimation for Neural Machine Translation”, Marina Fomicheva, Shuo Sun, Lisa Yankovskaya, Frédéric Blain, Francisco Guzmán, Mark Fishel, Nikolaos Aletras, Vishrav Chaudhary and Lucia Specia, *Transactions of the Association for Computational Linguistics (TACL)*, vol.8, pp.539-555, 2020

2020 “A Study in Improving BLEU Reference Coverage with Diverse Automatic Paraphrasing”, Rachel Bawden, Biao Zhang, Lisa Yankovskaya, Andre Tättar and Matt Post, *Findings of the Association for Computational Linguistics: EMNLP 2020*, pp.918–932, 2020

2019 “Findings of the WMT 2019 Shared Tasks on Quality Estimation”, Erick Fonseca, Lisa Yankovskaya, André F. T. Martins, Mark Fishel, and Christian Federmann, *Proceedings of the Fourth Conference on Machine Translation*, vol.3, pp.1-10, 2019

2019 “Quality Estimation and Translation Metrics via Pre-trained Word and Sentence Embeddings”, Lisa Yankovskaya, Andre Tättar, and Mark Fishel, *Proceedings of the Fourth Conference on Machine Translation*, vol.3, pp.101-105, 2019

2018 “Quality Estimation with Force-Decoded Attention and Cross-lingual Embeddings”, Elizaveta Yankovskaya, Andre Tättar, and Mark Fishel, *Proceedings of the Third Conference on Machine Translation*, pp. 829-834, 2018

2018 “Low-Resource Translation Quality Estimation for Estonian”, Elizaveta Yankovskaya and Mark Fishel, *Human Language Technologies - The Baltic Perspective*, pp. 175-182, 2018

ELULOOKIRJELDUS

Isikuandmed

Nimi: Elizaveta (Lisa) Yankovskaya
E-post: lisa.yankovskaya@ut.ee

Haridus

2017–... Tartu Ülikool, Arvutiteaduse Instituut, doktoriõpe
2015–2017 Tartu Ülikool, Arvutiteaduse Instituut, magistriõpe

Teenistuskäik

2019–... keeletehnoloogia nooremteadur, Arvutiteaduse Instituut, Tartu Ülikool

Õppetöö

- Tartu Ülikool, loodus- ja täppisteaduste valdkond, arvutiteaduse instituut, õppeassistent aines masinõpe, sügis 2020
- Digihumanitaaria teadusseminari juhendaja, kevad 2021

Teadustegevus

Main fields of interest:

- masinõpe
- keeletehnoloogia
- andmeteadus

Väljaanded (keeletehnoloogiaga seotud)

2021 “Direct Exploitation of Attention Weights for Translation Quality Estimation”, Lisa Yankovskaya and Mark Fishel, *Proceedings of the Sixth Conference on Machine Translation*, pp. 955-960, 2021

2021 “Backtranslation Feedback Improves User Confidence in MT, Not Quality”, Vilém Zouhar, Michal Novák, Matús Zilinec, Ondrej Bojar, Mateo Obregón, Robin L. Hill, Frédéric Blain, Marina Fomicheva, Lucia Specia and Lisa Yankovskaya, *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 151–161, 2021

2020 "BERGAMOT-LATTE Submissions for the WMT20 Quality Estimation Shared Task", Marina Fomicheva, Shuo Sun, Lisa Yankovskaya, Frédéric Blain, Vishrav Chaudhary, Mark Fishel, Francisco Guzmán and Lucia Specia, *Proceedings of the Fifth Conference on Machine Translation*, pp.1010-1017, 2020

2020 "Unsupervised Quality Estimation for Neural Machine Translation", Marina Fomicheva, Shuo Sun, Lisa Yankovskaya, Frédéric Blain, Francisco Guzmán, Mark Fishel, Nikolaos Aletras, Vishrav Chaudhary and Lucia Specia, *Transactions of the Association for Computational Linguistics (TACL)*, vol.8, pp.539-555, 2020

2020 "A Study in Improving BLEU Reference Coverage with Diverse Automatic Paraphrasing", Rachel Bawden, Biao Zhang, Lisa Yankovskaya, Andre Tättar and Matt Post, *Findings of the Association for Computational Linguistics: EMNLP 2020*, pp.918–932, 2020

2019 "Findings of the WMT 2019 Shared Tasks on Quality Estimation", Erick Fonseca, Lisa Yankovskaya, André F. T. Martins, Mark Fishel, and Christian Federmann, *Proceedings of the Fourth Conference on Machine Translation*, vol.3, pp.1-10, 2019

2019 "Quality Estimation and Translation Metrics via Pre-trained Word and Sentence Embeddings", Lisa Yankovskaya, Andre Tättar, and Mark Fishel, *Proceedings of the Fourth Conference on Machine Translation*, vol.3, pp.101-105, 2019

2018 "Quality Estimation with Force-Decoded Attention and Cross-lingual Embeddings", Elizaveta Yankovskaya, Andre Tättar, and Mark Fishel, *Proceedings of the Third Conference on Machine Translation*, pp. 829-834, 2018

2018 "Low-Resource Translation Quality Estimation for Estonian", Elizaveta Yankovskaya and Mark Fishel, *Human Language Technologies - The Baltic Perspective*, pp. 175-182, 2018

DISSERTATIONES INFORMATICAE PREVIOUSLY PUBLISHED IN DISSERTATIONES MATHEMATICAE UNIVERSITATIS TARTUENSIS

19. **Helger Lipmaa.** Secure and efficient time-stamping systems. Tartu, 1999, 56 p.
22. **Kaili Müürisep.** Eesti keele arvutigrammatika: süntaks. Tartu, 2000, 107 lk.
23. **Varmo Vene.** Categorical programming with inductive and coinductive types. Tartu, 2000, 116 p.
24. **Olga Sokratova.** Ω-rings, their flat and projective acts with some applications. Tartu, 2000, 120 p.
27. **Tiina Puolakainen.** Eesti keele arvutigrammatika: morfoloogiline ühestamine. Tartu, 2001, 138 lk.
29. **Jan Villemson.** Size-efficient interval time stamps. Tartu, 2002, 82 p.
45. **Kristo Heero.** Path planning and learning strategies for mobile robots in dynamic partially unknown environments. Tartu 2006, 123 p.
49. **Härmel Nestra.** Iteratively defined transfinite trace semantics and program slicing with respect to them. Tartu 2006, 116 p.
53. **Marina Issakova.** Solving of linear equations, linear inequalities and systems of linear equations in interactive learning environment. Tartu 2007, 170 p.
55. **Kaarel Kaljurand.** Attempto controlled English as a Semantic Web language. Tartu 2007, 162 p.
56. **Mart Anton.** Mechanical modeling of IPMC actuators at large deformations. Tartu 2008, 123 p.
59. **Reimo Palm.** Numerical Comparison of Regularization Algorithms for Solving Ill-Posed Problems. Tartu 2010, 105 p.
61. **Jüri Reimand.** Functional analysis of gene lists, networks and regulatory systems. Tartu 2010, 153 p.
62. **Ahti Peder.** Superpositional Graphs and Finding the Description of Structure by Counting Method. Tartu 2010, 87 p.
64. **Vesal Vojdani.** Static Data Race Analysis of Heap-Manipulating C Programs. Tartu 2010, 137 p.
66. **Mark Fišel.** Optimizing Statistical Machine Translation via Input Modification. Tartu 2011, 104 p.
67. **Margus Niitsoo**. Black-box Oracle Separation Techniques with Applications in Time-stamping. Tartu 2011, 174 p.
71. **Siim Karus.** Maintainability of XML Transformations. Tartu 2011, 142 p.
72. **Margus Treumuth.** A Framework for Asynchronous Dialogue Systems: Concepts, Issues and Design Aspects. Tartu 2011, 95 p.
73. **Dmitri Lepp.** Solving simplification problems in the domain of exponents, monomials and polynomials in interactive learning environment T-algebra. Tartu 2011, 202 p.

74. **Meelis Kull.** Statistical enrichment analysis in algorithms for studying gene regulation. Tartu 2011, 151 p.
77. **Bingsheng Zhang.** Efficient cryptographic protocols for secure and private remote databases. Tartu 2011, 206 p.
78. **Reina Uba.** Merging business process models. Tartu 2011, 166 p.
79. **Uuno Puus.** Structural performance as a success factor in software development projects – Estonian experience. Tartu 2012, 106 p.
81. **Georg Singer.** Web search engines and complex information needs. Tartu 2012, 218 p.
83. **Dan Bogdanov.** Sharemind: programmable secure computations with practical applications. Tartu 2013, 191 p.
84. **Jevgeni Kabanov.** Towards a more productive Java EE ecosystem. Tartu 2013, 151 p.
87. **Margus Freudenthal.** Simpl: A toolkit for Domain-Specific Language development in enterprise information systems. Tartu, 2013, 151 p.
90. **Raivo Kolde.** Methods for re-using public gene expression data. Tartu, 2014, 121 p.
91. **Vladimir Šor.** Statistical Approach for Memory Leak Detection in Java Applications. Tartu, 2014, 155 p.
92. **Naved Ahmed.** Deriving Security Requirements from Business Process Models. Tartu, 2014, 171 p.
94. **Liina Kamm.** Privacy-preserving statistical analysis using secure multi-party computation. Tartu, 2015, 201 p.
100. **Abel Armas Cervantes.** Diagnosing Behavioral Differences between Business Process Models. Tartu, 2015, 193 p.
101. **Fredrik Milani.** On Sub-Processes, Process Variation and their Interplay: An Integrated Divide-and-Conquer Method for Modeling Business Processes with Variation. Tartu, 2015, 164 p.
102. **Huber Raul Flores Macario.** Service-Oriented and Evidence-aware Mobile Cloud Computing. Tartu, 2015, 163 p.
103. **Tauno Metsalu.** Statistical analysis of multivariate data in bioinformatics. Tartu, 2016, 197 p.
104. **Riivo Talviste.** Applying Secure Multi-party Computation in Practice. Tartu, 2016, 144 p.
108. **Siim Orasmaa.** Explorations of the Problem of Broad-coverage and General Domain Event Analysis: The Estonian Experience. Tartu, 2016, 186 p.
109. **Prastudy Mungkas Fauzi.** Efficient Non-interactive Zero-knowledge Protocols in the CRS Model. Tartu, 2017, 193 p.
110. **Pelle Jakovits.** Adapting Scientific Computing Algorithms to Distributed Computing Frameworks. Tartu, 2017, 168 p.
111. **Anna Leontjeva.** Using Generative Models to Combine Static and Sequential Features for Classification. Tartu, 2017, 167 p.
112. **Mozhgan Pourmoradnasseri.** Some Problems Related to Extensions of Polytopes. Tartu, 2017, 168 p.

113. **Jaak Randmets.** Programming Languages for Secure Multi-party Computation Application Development. Tartu, 2017, 172 p.
114. **Alisa Pankova.** Efficient Multiparty Computation Secure against Covert and Active Adversaries. Tartu, 2017, 316 p.
116. **Toomas Saarsen.** On the Structure and Use of Process Models and Their Interplay. Tartu, 2017, 123 p.
121. **Kristjan Korjus.** Analyzing EEG Data and Improving Data Partitioning for Machine Learning Algorithms. Tartu, 2017, 106 p.
122. **Eno Tõnisson.** Differences between Expected Answers and the Answers Offered by Computer Algebra Systems to School Mathematics Equations. Tartu, 2017, 195 p.

DISSERTATIONES INFORMATICAE UNIVERSITATIS TARTUENSIS

1. **Abdullah Makkeh**. Applications of Optimization in Some Complex Systems. Tartu 2018, 179 p.
2. **Riivo Kikas**. Analysis of Issue and Dependency Management in Open-Source Software Projects. Tartu 2018, 115 p.
3. **Ehsan Ebrahimi**. Post-Quantum Security in the Presence of Superposition Queries. Tartu 2018, 200 p.
4. **Ilya Verenich**. Explainable Predictive Monitoring of Temporal Measures of Business Processes. Tartu 2019, 151 p.
5. **Yauhen Yakimenka**. Failure Structures of Message-Passing Algorithms in Erasure Decoding and Compressed Sensing. Tartu 2019, 134 p.
6. **Irene Teinemaa**. Predictive and Prescriptive Monitoring of Business Process Outcomes. Tartu 2019, 196 p.
7. **Mohan Liyanage.** A Framework for Mobile Web of Things. Tartu 2019, 131 p.
8. **Toomas Krips.** Improving performance of secure real-number operations. Tartu 2019, 146 p.
9. **Vijayachitra Modhukur.** Profiling of DNA methylation patterns as biomarkers of human disease. Tartu 2019, 134 p.
10. **Elena Sügis.** Integration Methods for Heterogeneous Biological Data. Tartu 2019, 250 p.
11. **Tõnis Tasa.** Bioinformatics Approaches in Personalised Pharmacotherapy. Tartu 2019, 150 p.
12. **Sulev Reisberg.** Developing Computational Solutions for Personalized Medicine. Tartu 2019, 126 p.
13. **Huishi Yin.** Using a Kano-like Model to Facilitate Open Innovation in Requirements Engineering. Tartu 2019, 129 p.
14. **Faiz Ali Shah.** Extracting Information from App Reviews to Facilitate Software Development Activities. Tartu 2020, 149 p.
15. **Adriano Augusto**. Accurate and Efficient Discovery of Process Models from Event Logs. Tartu 2020, 194 p.
16. **Karim Baghery.** Reducing Trust and Improving Security in zk-SNARKs and Commitments. Tartu 2020, 245 p.
17. **Behzad Abdolmaleki.** On Succinct Non-Interactive Zero-Knowledge Protocols Under Weaker Trust Assumptions. Tartu 2020, 209 p.
18. **Janno Siim.** Non-Interactive Shuffle Arguments. Tartu 2020, 154 p.
19. **Ilya Kuzovkin.** Understanding Information Processing in Human Brain by Interpreting Machine Learning Models. Tartu 2020, 149 p.
20. **Orlenys López Pintado.** Collaborative Business Process Execution on the Blockchain: The Caterpillar System. Tartu 2020, 170 p.
21. **Ardi Tampuu.** Neural Networks for Analyzing Biological Data. Tartu 2020, 152 p.

22. **Madis Vasser.** Testing a Computational Theory of Brain Functioning with Virtual Reality. Tartu 2020, 106 p.
23. **Ljubov Jaanuska.** Haar Wavelet Method for Vibration Analysis of Beams and Parameter Quantification. Tartu 2021, 192 p.
24. **Arnis Parsovs.** Estonian Electronic Identity Card and its Security Challenges. Tartu 2021, 214 p.
25. **Kaido Lepik.** Inferring causality between transcriptome and complex traits. Tartu 2021, 224 p.
26. **Tauno Palts.** A Model for Assessing Computational Thinking Skills. Tartu 2021, 134 p.
27. **Liis Kolberg.** Developing and applying bioinformatics tools for gene expression data interpretation. Tartu 2021, 195 p.
28. **Dmytro Fishman.** Developing a data analysis pipeline for automated protein profiling in immunology. Tartu 2021, 155 p.
29. **Ivo Kubjas.** Algebraic Approaches to Problems Arising in Decentralized Systems. Tartu 2021, 120 p.
30. **Hina Anwar.** Towards Greener Software Engineering Using Software Analytics. Tartu 2021, 186 p.
31. **Veronika Plotnikova.** FIN-DM: A Data Mining Process for the Financial Services. Tartu 2021, 197 p.
32. **Manuel Camargo.** Automated Discovery of Business Process Simulation Models From Event Logs: A Hybrid Process Mining and Deep Learning Approach. Tartu 2021, 130 p.
33. **Volodymyr Leno.** Robotic Process Mining: Accelerating the Adoption of Robotic Process Automation. Tartu 2021, 119 p.
34. **Kristjan Krips.** Privacy and Coercion-Resistance in Voting. Tartu 2022, 173 p.