Quality In, Quality Out: Learning from Actual Mistakes

Frédéric Blain[1] **Nikolaos Aletras**[1] **Lucia Specia**[1,2]
[1]Department of Computer Science, University of Sheffield
[2]Department of Computing, Imperial College London
United Kingdom
{f.blain,n.aletras,l.specia}@sheffield.ac.uk

Abstract

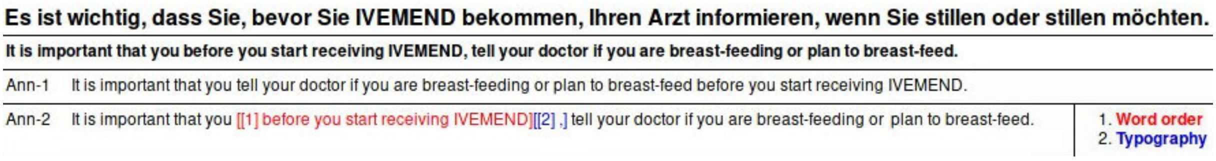
Approaches to Quality Estimation (QE) of machine translation have shown promising results at predicting quality scores for translated sentences. However, QE models are often trained on noisy approximations of quality annotations derived from the proportion of post-edited words in translated sentences instead of direct human annotations of translation errors. The latter is a more reliable ground-truth but more expensive to obtain. In this paper, we present the first attempt to model the task of predicting the proportion of *actual* translation errors in a sentence while minimising the need for direct human annotation. For that purpose, we use transfer-learning to leverage large scale noisy annotations and small sets of high-quality human annotated translation errors to train QE models. Experiments on four language pairs and translations obtained by statistical and neural models show consistent gains over strong baselines.

1 Introduction

Quality Estimation (QE) for Machine Translation (MT) is the task of predicting the overall quality of an automatically generated translation *e.g.*, on either word, sentence or document level (Blatz et al., 2004; Ueffing and Ney, 2007). In opposition to automatic metrics and manual evaluation which rely on gold standard reference translations, QE models can produce quality estimates on unseen data, and at runtime. QE has already proven its usefulness in many applications such as improving productivity in post-editing of MT, and recent neural-based approaches to QE have been shown to provide promising performance in predicting quality of neural MT output (Fonseca et al., 2019).

QE models are trained under full supervision, which requires to have quality-labelled training data at hand. Obtaining annotated data for all the domains and languages of interest is costly and often impractical. As a result, QE models can suffer from the same limitations as neural MT models themselves, such as drastic degradation of their performance on out-of-domain data. As an alternative, QE models are often trained under weak supervision, using training instances labelled from noisy or limited sources (e.g. data labelled with automatic metrics for MT).

Here, we focus on sentence-level QE, where given a pair of sentences (the source and its translation), the aim is to train supervised Machine Learning (ML) models that can predict a quality label as a numerical value. The most widely used label for sentence-level QE is the Human-mediated Translation Edit Rate (HTER) (Snover et al., 2006), which represents the *post-editing effort*. HTER consists of the minimum number of edits a human language expert is required to make in order to fix the translation errors in a sentence, taking values between 0 and 1. The main limitation of HTER is that it does not represent an actual translation error rate, but its noisy approximation. The noise stems mostly from errors in the heuristics used to automatically align the machine translation and its post-edited version, but also from the fact that some edits represent preferential choices of humans, rather than errors. To overcome such limitations, QE models can be improved by using data that has been



Es ist wichtig, dass Sie, bevor Sie IVEMEND bekommen, Ihren Arzt informieren, wenn Sie stillen oder stillen möchten.

It is important that you before you start receiving IVEMEND, tell your doctor if you are breast-feeding or plan to breast-feed.

Ann-1	It is important that you tell your doctor if you are breast-feeding or plan to breast-feed before you start receiving IVEMEND.	
Ann-2	It is important that you [[1] before you start receiving IVEMEND][[2] ,] tell your doctor if you are breast-feeding or plan to breast-feed.	1. Word order 2. Typography

Figure 1: Example of a German sentence (top) and its automatic translation into English. The HTER between the translation and its post-edited version (ANN-1) is 0.091, while the proportion of fine-grained expert-annotated MT errors (ANN-2), is $6/23 = 0.261$.

directly annotated for translation errors by human experts. Figure 1 shows an example of the discrepancy between the HTER score and the proportion of *actual* errors from expert annotation, for a raw translation and its post-edited version.

Annotations of MT errors usually follow fine-grained error taxonomies such as the Multidimensional Quality Metrics (MQM) framework (Lommel et al., 2014). While such annotations provide highly reliable labelled data, they are more expensive to produce than HTER. This often results in datasets that are orders of magnitude smaller than HTER-based ones. This makes it hard to only use such high-quality resources for training neural-based QE models, which typically require large amounts of training data.

In this paper, we use transfer-learning to develop QE models by exploiting the advantages of both noisy and high-quality labelled data. We leverage information from large amounts of HTER data and small amounts of MQM annotations to train more reliable sentence-level QE models. Our aim is to predict the proportion of *actual* errors in MT outputs. More fine-grained error prediction is left for future work.

Main contributions: (1) We introduce a new task of predicting the proportion of actual translation errors using transfer-learning for QE[1], by leveraging large scale noisy HTER annotations and smaller but of higher quality expert MQM annotations; (2) we show that our simple yet effective approach using transfer-learning yields better performance at predicting the proportion of actual errors in MT, compared to models trained directly on expert-annotated MQM or HTER-only data; (3) we report experiments on four language pairs and both statistical and neural MT systems.

2 Related Work

Quality labels for sentence-level QE Quirk (2004) introduced the use of manually created quality labels for evaluating MT systems. With a rather small dataset (approximately 350 sentences), they reported better results than those obtained with a much larger set of instances annotated automatically. Similarly, Specia et al. (2009) proposed the use of a (1-4) *Likert* scale representing a translator's perception on quality with regard to the degree of difficulty to fix a translation. However, sentence-level quality annotations appear to be subjective while agreement between annotators is generally low (Specia, 2011). More recently, sentence-level QE models are most typically trained on HTER scores (Bojar et al., 2013; Bojar et al., 2014; Bojar et al., 2015; Bojar et al., 2016; Bojar et al., 2017; Specia et al., 2018; Fonseca et al., 2019).

Transfer-learning for QE Transfer-learning (TL) is a machine learning approach where models trained on a *source* task are adapted to a related *target* task (Pan et al., 2010; Yosinski et al., 2014). Transfer-learning methods have been widely used in NLP, *e.g.*, machine translation (Zoph et al., 2016) and text classification (Howard and Ruder, 2018). Previous work on TL for QE focused on adapting models for labels produced by different annotators (Cohn and Specia, 2013; Shah and Specia, 2016) which is different to this work.

More recent work on TL techniques for QE explore pre-trained word representations. This was first done by POSTECH (Kim et al., 2017), best performing neural-based architecture in the QE shared task at WMT'17 (Bojar et al., 2017). POSTECH re-purposes a recurrent neural network encoder pre-trained on large parallel corpora, to predict HTER scores using multi-task learning at different levels of granularity (*e.g.*, word, phrase, or sentence). Then, Kepler et al. (2019) used a predictor-estimator architecture similar to POSTECH alongside very large scale pre-trained representations from BERT (Devlin et al., 2018) and XLM (Lample and Conneau, 2019), and ensembling techniques, to win the QE tasks at WMT'19 (Fonseca

[1] https://github.com/sheffieldnlp/tlqe

et al., 2019). These models are pre-trained on unlabelled data, as opposed to noisier labelled data, and aim to predict HTER scores, which is different to the focus of this paper.

To the best of our knowledge, this paper is the first attempt to repurpose a QE model pre-trained on one quality label to a model that predicts another quality label; we first train a model on noisy HTER data to predict post-editing effort, and leverage its knowledge to train a model capable of predicting the *actual* proportion of translation errors using expert-annotated MQM data.

3 Transfer-Learning Approach

We use inductive transfer-learning (Pan et al., 2010), where given a source learning task $\mathcal{T}_S$ and a target task $\mathcal{T}_T$, the aim is to improve performance in the latter by re-using knowledge from $\mathcal{T}_S$, where $\mathcal{T}_S \neq \mathcal{T}_T$. Here, $\mathcal{T}_S$ corresponds to predicting *post-editing effort* based on noisy HTER annotations, and $\mathcal{T}_T$ to predicting the proportion of *actual* proportion of errors based on MQM annotations.

3.1 Source task QE model

BiRNN-HTER We use the BiRNN model proposed by Ive et al. (2018) as our base model to predict HTER scores. Figure 2 illustrates the high-level architecture of the model. Words in source and translated sentences are first mapped into embedding vectors. Then, the word embeddings are passed through bidirectional Gated Recurrent Unit encoders (Cho et al., 2014) to learn context-aware word representations in both the source and target sentences. The two sentence representations are learned independently from each other before being concatenated as a weighted sum of their word vectors, generated by an attention mechanism. The concatenated representation is finally passed through a dense layer with sigmoid activation to generate the quality estimate. BiRNN performed competitively in the WMT'18 shared task on QE (Specia et al., 2018) without relying on any parallel data nor expensive pre-training regimes such as the POSTECH approach (Section 2). Overall, it is easier and faster to train with a smaller number of parameters compared to POSTECH, which makes it more suitable for this task.

3.2 Adaptation to the target task

Our target task is to predict the proportion (between 0 and 1) of actual MQM errors in a translated sentence. Therefore, we adapt our BiRNN-HTER model to the target task.

BiRNN-MQM$_{TL}$ We first replace the BiRNN-HTER output layer with two new layers: (1) a fully-connected layer followed by a rectified linear unit (Nair and Hinton, 2010) as the activation function; and (2) a fully-connected output layer with a sigmoid activation to produce the predictions. We train these two layers on target task data by freezing the rest of the model.

BiRNN-MQM$_{TL}$+FT We further *fine-tune* our BiRNN-MQM$_{TL}$ model on the target task data using a small learning rate following (Howard and Ruder, 2018).

Hybrid Finally, we hypothesise that linguistic information (*e.g.*, number of tokens in the source/target sentence, language model probability of source/target sentence, etc.) might be complementary to the source-target representations obtained by our BiRNN-MQM$_{TL}$+FT model. For that purpose, we first extract a representation of the source and translated sentence by removing the BiRNN-MQM$_{TL}$+FT output layer and then we concatenate it with the widely used 17 black-box sentence-level QE features extracted with the open-source QuEst++ toolkit (Specia et al., 2015). The joint neural and linguistic information of the source and target sentences is fed into a linear regression[2] model using a L2 regularisation penalty.

4 Experimental Setup

4.1 Data

For our experiments, we use the freely available QT21 dataset[3] (Specia et al., 2017) used in the QE shared task (Bojar et al., 2017; Specia et al., 2018). This dataset contains both post-edited (HTER) and error-annotated (MQM) data in four language pairs: English into German, Latvian and Czech, and German into English; and phrase-based statistical (PBMT) and neural (NMT) translation models. The annotation for errors was produced by professional translators using the MQM taxonomy

[2] We also tried to jointly feed the features during fine-tuning but did not yield better performance.

[3] http://www.qt21.eu/resources/data/

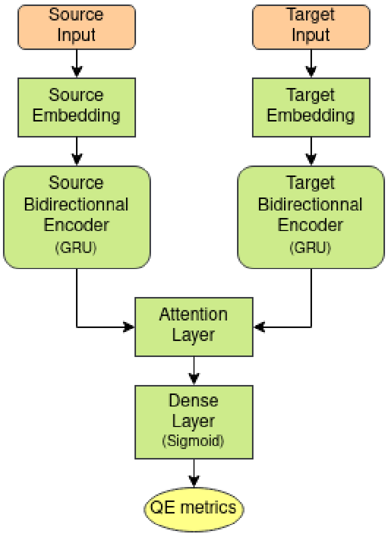


Figure 2: High-level architecture of the BiRNN sentence-level QE model.

	HTER data (Source)		MQM data (Target)	
	# sentences		# sentences	
	PBMT	NMT	PBMT	NMT
EN-DE	25,305	12,564	2,655	3,386
EN-LV	10,561	11,116	3,284	3,244
DE-EN	25,922	–	3,374	–
EN-CS	37,725	–	3,460	–

Table 1: Statistics for HTER and MQM data for statistical (PBMT) and neural (NMT) translation systems across language pairs.

with 21 error categories (*e.g.*, mistranslation, morphology, etc.). To obtain a score for the entire sentence, we divide the number of words annotated with any error category by the length of the sentence. Predicting the actual type of MQM errors is left for future work. Note that the MQM-annotated sentences are a subset of the HTER data (*i.e.* some of them have both annotations), so we removed these from the HTER data.

By design, all sentences selected for MQM annotation have at least one error. In order to increase the size and variety of the MQM dataset, we doubled the number of MQM-annotated sentences by taking sentences for which no edit was made during PE (*i.e.* perfect translations with zero MQM errors). Table 1 summarises the statistics of the labelled data used for our experiments.

4.2 Baseline and comparison models

To assess our models, we compare them against the following baselines.

BiRNN-HTER A BiRNN-HTER model trained on the HTER data and used as is, to predict the proportion of MQM errors. That is using the source task base model to predict the scores in the target task.

BiRNN-MQM This is the same BiRNN architecture as our source task model (BiRNN-HTER) but trained from-scratch on the MQM data without transfer-learning.

LR-QEfeat A feature-based approach used in the WMT shared tasks as an official baseline. We use the 17 black-box sentence-level QE features introduced above (see Section 3.2) to train a linear regression[4] model with a L2 regularization penalty.

4.3 Model hyper-parameters

For the BiRNN-HTER model, we use default parameters as in (Ive et al., 2018). For the BiRNN-MQM$_{TL}$, we use a 5-fold Cross Validation approach. We use a dense layer[5] of 50 and choose the number of epochs in $\{1, .., 40\}$, training learning rate in $\{1e^{-2}, 1e^{-3}\}$ and fine-tuning learning rate in $\{1e^{-3}, 1e^{-4}\}$ on a validation set, by minimising the Mean Absolute Error (MAE) between the predicted score and gold standard labels. We also experimented with two approaches for fine-tuning: (1) unfreezing all the layers at the same time; and (2) a gradual unfreezing approach proposed by (Howard and Ruder, 2018). We use Adam (Kingma and Ba, 2014) with default parameters, and a batch size of 100. For the Hybrid model, we optimise the L2 regularisation penalty.

Table 2 reports on the optimal values determined by hyper-parameters optimisation.

5 Results

Tables 3 and 4 show respectively the average absolute Pearson's r correlation co-efficient and the Root Mean Square Error (the official metrics for this task (Graham, 2015)) between actual and predicted MQM error proportions in six combinations of MT models (PBMT, NMT) and language pairs (EN-DE, EN-LV, DE-EN and EN-CS).

First, we observe that the baseline model (LR-QEfeat) performs fairly well on predicting the proportion of errors, especially for the EN-DE and EN-CS PBMT. However, it is not robust across language pairs and types of translation systems.

[4] We have also tested a Support Vector Regression with a radial basis function kernel, but it yielded lower performance.

[5] We did not observe noticeable differences in performance using smaller or larger size in early experimentation.

	Training		Fine-tuning	
	Epochs	Learning rate	Epochs/Method	Learning rate
EN-DE$_{NMT}$	22	0.01	gradual unfreezing	0.001
EN-LV$_{NMT}$	16	0.001	gradual unfreezing	0.001
EN-DE$_{PBMT}$	15	0.001	1	0.001
EN-LV$_{PBMT}$	18	0.01	gradual unfreezing	0.001
DE-EN$_{PBMT}$	19	0.01	1	0.001
EN-CS$_{PBMT}$	18	0.001	gradual unfreezing	0.001

Table 2: Optimal values selected for the adaptation of the source task sentence-level BiRNN QE model (BiRNN-HTER) to the target task (*i.e.* proportion of *actual* MT error in MT). For each language pair: number of epochs and learning rates for the training, and number of epochs or method used for the fine-tuning of the model.

	EN-DE$_{NMT}$	EN-LV$_{NMT}$	EN-DE$_{PBMT}$	EN-LV$_{PBMT}$	DE-EN$_{PBMT}$	EN-CS$_{PBMT}$
(1) LR-QEfeat	0.152 ±0.06	0.404 ±0.19	0.585 ±0.02	0.471 ±0.06	0.329 ±0.02	0.635 ±0.02
(2) BiRNN-HTER	0.297 ±0.04	0.003 ±0.09	0.146 ±0.06	0.110 ±0.05	0.113 ±0.07	0.426 ±0.05
(3) BiRNN-MQM	$\underline{0.584}$ ±0.04	0.542 ±0.05	$\underline{0.619}$ ±0.05	0.583 ±0.03	0.606 ±0.08	$\underline{0.757}$ ±0.01
(4) BiRNN-MQM$_{TL}$	$\underline{0.575}$ ±0.04	0.596 ±0.06	$\underline{0.644}$ ±0.02	0.612 ±0.03	0.594 ±0.02	$\underline{0.787}$ ±0.03
(5) BiRNN-MQM$_{TL}$+FT	**$\underline{0.649}$ ±0.05**	**$\underline{0.612}$ ±0.06**	$\underline{0.648}$ ±0.04	$\underline{0.649}$ ±0.04	$\underline{0.601}$ ±0.05	$\underline{0.793}$ ±0.02
(6) Hybrid	$\underline{0.644}$ ±0.05	0.522 ±0.28	**$\underline{0.658}$ ±0.04**	**$\underline{0.655}$ ±0.03**	**$\underline{0.610}$ ±0.05**	**$\underline{0.795}$ ±0.02**

Table 3: Average absolute **Pearson's** r correlation between actual and **predicted MQM error proportions** across all folds in six combinations of MT models and language pairs: **(1)** feature-based baseline (LR-QEfeat) – **(2)** BiRNN model trained on HTER data, and used as is – **(3)** BiRNN model trained from scratch on MQM annotated data – **(4)** BiRNN MQM trained with transfer-learning, *i.e.* trained on HTER data and adapted using MQM data – **(5)** BiRNN-MQM$_{TL}$ model fine-tuned with additional training epochs – **(6)** fine-tuned BiRNN-MQM$_{TL}$+FT model used as feature extractor along with the 17 sentence-level QE features and a linear regression algorithm (Hybrid). Measurements not significantly outperformed by any other overall, are underlined. Significance is computed with Hotelling-Williams test (Williams, 1959).

	EN-DE$_{NMT}$	EN-LV$_{NMT}$	EN-DE$_{PBMT}$	EN-LV$_{PBMT}$	DE-EN$_{PBMT}$	EN-CS$_{PBMT}$
(1) LR-QEfeat	0.112 ±0.01	0.157 ±0.10	0.161 ±0.01	0.114 ±0.01	0.115 ±0.00	0.175 ±0.01
(2) BiRNN-HTER	0.117 ±0.01	0.523 ±0.01	0.250 ±0.01	0.460 ±0.01	0.605 ±0.03	0.333 ±0.01
(3) BiRNN-MQM	0.093 ±0.01	0.108 ±0.01	0.157 ±0.01	0.110 ±0.01	**0.097** ±0.00	0.152 ±0.01
(4) BiRNN-MQM$_{TL}$	0.094 ±0.01	**0.102** ±0.01	0.158 ±0.01	0.108 ±0.01	0.110 ±0.00	0.145 ±0.01
(5) BiRNN-MQM$_{TL}$+FT	0.091 ±0.01	0.105 ±0.01	0.152 ±0.01	0.100 ±0.01	0.100 ±0.00	0.139 ±0.01
(6) Hybrid	**0.087** ±0.01	0.212 ±0.26	**0.149** ±0.01	**0.098** ±0.01	**0.097** ±0.00	**0.138** ±0.01

Table 4: Average absolute **RMSE** between actual and **predicted MQM error proportions** across all folds in six combinations of MT models and language pairs: **(1)** feature-based baseline (LR-QEfeat) – **(2)** BiRNN model trained on HTER data, and used as is – **(3)** BiRNN model trained from scratch on MQM annotated data – **(4)** BiRNN MQM trained with transfer-learning, *i.e.* trained on HTER data and adapted using MQM data – **(5)** BiRNN-MQM$_{TL}$ model fine-tuned with additional training epochs – **(6)** fine-tuned BiRNN-MQM$_{TL}$+FT model used as feature extractor along with the 17 sentence-level QE features and a linear regression algorithm (Hybrid). Measurements not significantly outperformed by any other overall, are underlined. Significance is computed with Hotelling-Williams test (Williams, 1959).

Second, the BiRNN-HTER model, trained on HTER data and used as is, is not able to predict the proportion of actual MQM errors. Surprisingly, the BiRNN-MQM model trained on MQM data directly achieves relatively good performance for all language pairs. This seems to confirm that (i) the BiRNN architecture, as simple as it may be, allows to train models that perform well while keeping low the computational resources required; and (ii) that HTER is a noisy approximation of the quality of a translation and post-edits are not actually well-aligned to actual translation errors.

Overall, the best performing model is BiRNN-MQM$_{TL}$ with transfer-learning and fine-tuning, while our Hybrid model seems to further improve performance in predicting quality on statistical MT output. This is in line with recent findings demonstrating the benefits of feature-based approaches for predicting the quality of statistical MT, but not for predicting the quality of neural MT, which is better modelled with learned representations using neural networks (Specia et al., 2018). This also confirms our main hypothesis that noisy data, but from a closely related task, encapsulates useful information that our TL model is able to leverage.

6 Leveraging Pre-trained Token-level Representations

As reported in (Fonseca et al., 2019), state-of-the-art models for supervised QE follow current trend in the NLP community in 2019: leveraging large-scale pre-trained language models to compute word- or sentence-level representations. Following (Kepler et al., 2019) and their Transformer-based Predictor-Estimator model, we considered two variants of our BiRNN-HTER model introduced in Section 3:

LM-BiRNN By default, the weights of both the source and target bidirectional GRU encoders of the BiRNN model are first randomly initiated and then learned, simultaneously, during training of the task at hand. In this variant, we first learn the weights of each encoder independently in a language modelling fashion with a Cross-Entropy loss, using the additional resources provided by the organisers of the WMT'18 QE shared task[6]. We then reuse the learned weights to initiate each encoder of the BiRNN model.

BERT-BiRNN In this variant of the BiRNN model, the token-level representations are extracted from a pre-trained multilingual base cased BERT (Devlin et al., 2018) model. Concretely, we replace both the source and the target embedding layers in Figure 2 by a single custom *BERT embedding* layer. During training, we fine-tune the weights of the word embeddings layer, as well as the weights of the last 4 encoding layers of the BERT model.

In the rest of the paper, and similarly to the naming of our models in Sections 3.1 and 3.2, we will refer to as "BERT-BiRNN-HTER", "BERT-BiRNN-MQM" and "BERT-BiRNN-MQM$_{TL}$", the three variants of this model trained from scratch on the source task (-HTER), on the target task (-MQM) and adapted to the target task using TL (-MQM$_{TL}$), respectively.

6.1 Experimental Results

We evaluate the benefit of using pre-trained token-level representations, by comparing the performance of our previously introduced BERT variants, against our base BiRNN model.

Predicting HTER

Table 5 summarises the performance of each model at predicting HTER scores on the HTER data described in Table 1. We include the BiRNN-HTER models from Tables 3 and 4 (row (2)) for direct comparison when trained at predicting HTER.

First, we observe that, overall, relying on pre-trained token representation helps to improve the performance of our BiRNN model, confirming the findings in (Fonseca et al., 2019). Second, while relying on advanced token representations such as those extracted from BERT significantly help improving across language pairs and types of translation, relying on simpler representations seems to mainly help on neural-based MT output, and with limited gains.

However, pre-trained representations usually require to be fine-tuned for the task at hand. In our scenario of application, where only a few datapoints of the target task is available, this may be a challenging task when using complex and deep architectures such as the BERT model, which contains millions of parameters trained on large scale training data (BERT models are trained on the Wikipedia dataset).

Predicting MQM with Transfer-Learning

We replicated the experimental settings for inductive transfer-learning described in Section 4, by considering this time the BERT variant of our base BiRNN model. Our experimental results are summarised in Tables 6 and 7, which report on Pearson's r correlation and RMSE, respectively. We include LR-QEfeat, the feature-based approach, as well as the default BiRNN-HTER and BiRNN-MQM models from Tables 3 and 4 (rows (1)-(4)) for direct comparison when trained at predicting MQM error proportions.

First, we observe that when our BiRNN model is trained at predicting the source task (HTER) and used as is to predict on the target task (MQM), more advanced representations can help improve its performance (rows (2) *vs.* (b)). However, both variants are usually outperformed by the baseline model (LR-QEfeat) on predicting the proportion of errors, apart from EN-DE NMT.

Second, when trained from scratch on MQM annotated data, the BERT-BiRNN model is significantly outperformed by our base BiRNN model across all language pairs and types of translation (rows (3) *vs.* (c)). While we previously observed the benefit of using advanced representations from BERT when at least 10,000 training datapoints are available (see Table 5), we now observe degraded performances when the number of training set is lower than 4,000 datapoints.

Third, when trained on HTER data and adapted

[6] http://statmt.org/wmt18/quality-estimation-task.html

	EN-DE$_{NMT}$	EN-LV$_{NMT}$	EN-DE$_{PBMT}$	EN-LV$_{PBMT}$	DE-EN$_{PBMT}$	EN-CS$_{PBMT}$
(2) BiRNN-HTER	0.290	0.436	0.347	0.416	0.505	0.480
(a) LM-BiRNN-HTER	0.372	0.443	0.395	0.384	0.495	0.476
(b) BERT-BiRNN-HTER	**0.390**	**0.561**	**0.612**	**0.520**	**0.641**	**0.537**

Table 5: Absolute **Pearson's** r correlation between actual and **predicted HTER scores**, for the HTER data introduced in Table 1: **(2)** default BiRNN model trained on HTER data – **(a)** BiRNN model with the weights of each source and target encoders pre-trained in a language modelling fashion using the additional resources of the QE shared task at WMT'18 – **(b)** BiRNN model with token-level representations extracted from a pre-trained multilingual base cased BERT model. Measurements not significantly outperformed by any other overall, are underlined. Significance is computed with Hotelling-Williams test (Williams, 1959).

	EN-DE$_{NMT}$	EN-LV$_{NMT}$	EN-DE$_{PBMT}$	EN-LV$_{PBMT}$	DE-EN$_{PBMT}$	EN-CS$_{PBMT}$
(1) LR-QEfeat	0.152 ±0.06	0.404 ±0.19	0.585 ±0.02	0.471 ±0.06	0.329 ±0.02	0.635 ±0.02
(2) BiRNN-HTER	0.297 ±0.04	0.003 ±0.09	0.146 ±0.06	0.110 ±0.05	0.113 ±0.07	0.426 ±0.05
(b) BERT-BiRNN-HTER	0.211 ±0.03	0.220 ±0.04	0.467 ±0.04	0.302 ±0.05	0.311 ±0.09	0.175 ±0.03
(3) BiRNN-MQM	**0.584 ±0.04**	**0.542 ±0.05**	**0.619 ±0.05**	**0.583 ±0.03**	**0.606 ±0.08**	**0.757 ±0.01**
(c) BERT-BiRNN-MQM	0.227 ±0.05	0.343 ±0.07	0.445 ±0.02	0.451 ±0.05	0.276 ±0.06	0.461 ±0.05
(4) BiRNN-MQM$_{TL}$	**0.575 ±0.04**	**0.596 ±0.06**	**0.644 ±0.02**	**0.612 ±0.03**	**0.594 ±0.02**	**0.787 ±0.03**
(d) BERT-BiRNN-MQM$_{TL}$	0.189 ±0.06	0.349 ±0.06	0.510 ±0.03	0.491 ±0.07	0.083 ±0.03	0.477 ±0.06

Table 6: Average absolute **Pearson's** r correlation between actual and **predicted MQM error proportions** across all folds in six combinations of MT models and language pairs: **(1)** feature-based baseline (LR-QEfeat) – **(2)** default BiRNN model trained on HTER data, and used as is – **(b)** BERT-BiRNN model trained on HTER data, and used as is – **(3)** BiRNN model trained from scratch on MQM annotated data – **(c)** BERT-BiRNN model trained from scratch on MQM annotated data – **(4)** BiRNN-MQM model trained with transfer-learning, *i.e.* trained on HTER data and adapted using MQM data. **(d)** BERT-BiRNN-MQM model trained with transfer-learning, *i.e.* trained on HTER data and adapted using MQM data. Measurements not significantly outperformed by any other overall, are underlined. Significance is computed with Hotelling-Williams test (Williams, 1959).

	EN-DE$_{NMT}$	EN-LV$_{NMT}$	EN-DE$_{PBMT}$	EN-LV$_{PBMT}$	DE-EN$_{PBMT}$	EN-CS$_{PBMT}$
(1) LR-QEfeat	0.112 ±0.01	0.157 ±0.10	0.161 ±0.01	0.114 ±0.01	0.115 ±0.00	0.175 ±0.01
(2a) BiRNN-HTER	0.117 ±0.01	0.523 ±0.01	0.250 ±0.01	0.460 ±0.01	0.605 ±0.03	0.333 ±0.01
(b) BERT-BiRNN-HTER	0.117 ±0.01	0.249 ±0.01	0.184 ±0.01	0.146 ±0.00	0.206 ±0.01	0.294 ±0.01
(3) BiRNN-MQM	**0.093 ±0.01**	**0.108 ±0.01**	**0.157 ±0.01**	**0.110 ±0.01**	**0.097 ±0.00**	**0.152 ±0.01**
(c) BERT-BiRNN-MQM	0.113 ±0.01	0.121 ±0.01	0.189 ±0.01	0.128 ±0.02	0.120 ±0.01	0.204 ±0.01
(4) BiRNN-MQM$_{TL}$	**0.094 ±0.01**	**0.102 ±0.01**	**0.158 ±0.01**	**0.108 ±0.01**	**0.110 ±0.00**	**0.145 ±0.01**
(d) BERT-BiRNN-MQM$_{TL}$	0.116 ±0.01	0.123 ±0.01	0.178 ±0.01	0.116 ±0.01	0.137 ±0.01	0.207 ±0.02

Table 7: Average absolute **RMSE** between actual and **predicted MQM error proportions** across all folds in six combinations of MT models and language pairs: **(1)** feature-based baseline (LR-QEfeat) – **(2)** default BiRNN model trained on HTER data, and used as is – **(b)** BERT-BiRNN model trained on HTER data, and used as is – **(3)** BiRNN model trained from scratch on MQM annotated data – **(c)** BERT-BiRNN model trained from scratch on MQM annotated data – **(4)** BiRNN-MQM model trained with transfer-learning, *i.e.* trained on HTER data and adapted using MQM data. **(d)** BERT-BiRNN-MQM model trained with transfer-learning, *i.e.* trained on HTER data and adapted using MQM data. Measurements not significantly outperformed by any other overall, are underlined. Significance is computed with Hotelling-Williams test (Williams, 1959).

using MQM data (rows (4) *vs.* (d)), we observe that the performance of the BERT-BiRNN model slightly improve compared to training from scratch on MQM data (row (c)) across all language pairs but EN-DE$_{NMT}$ and DE-EN$_{PBMT}$. For the latter, we even observe a significant drop in the performance of the model. There is no obvious explanations for that, so we hope that further experiments would help us to understand the reasons behind it. On the one hand, this confirms that fine-tuning deep architectures such as BERT to extract advanced token level representation is a challenging task when only a few training instances is available. On the other hand, we saw the benefit of using advanced representation from pre-trained models such as BERT, and plan to continue working towards that research direction.

7 Conclusions

We introduced a new task of predicting the proportion of actual errors in a translated sentence as an alternative to the commonly used noisy estimate HTER. The reported results from using inductive transfer-learning are particularly encouraging considering the simplicity of our BiRNN model. Our transfer-learning method helps to train models which are better at predicting the proportion of actual errors for different language pairs and translation systems, compared to models trained on the target task only.

However, whereas we were expecting to observe significant gains with the use of more advanced token-level pre-trained representations (here from BERT), we report drastic degradation in performances for this configuration when re-purposing the QE models via transfer-learning. These somewhat counter-intuitive results are an indication that further work can be done in this area to refine our transfer-learning approach, as the use of large scale pre-trained representations has become a common practice in NLP applications, including QE.

In addition to this, we plan in furture to estimate the quality of machine translation using more fine-grained MQM annotations for subsentence-level QE.

Acknowledgements

This work was supported by the Bergamot project (EU H2020 Grant No. 825303).

References

Blatz, John, Erin Fitzgerald, George Foster, Simona Gandrabur, Cyril Goutte, Alex Kulesza, Alberto Sanchis, and Nicola Ueffing. 2004. Confidence estimation for machine translation. In *COLING*.

Bojar, Ondrej, Christian Buck, Chris Callison-Burch, Christian Federmann, Barry Haddow, Philipp Koehn, Christof Monz, Matt Post, Radu Soricut, and Lucia Specia. 2013. Findings of the 2013 Workshop on Statistical Machine Translation. In *Eighth Workshop on Statistical Machine Translation*, WMT, pages 1–44, Sofia, Bulgaria.

Bojar, Ondrej, Christian Buck, Christian Federmann, Barry Haddow, Philipp Koehn, Johannes Leveling, Christof Monz, Pavel Pecina, Matt Post, Herve Saint-Amand, Radu Soricut, Lucia Specia, and Aleš Tamchyna. 2014. Findings of the 2014 workshop on statistical machine translation. In *Ninth Workshop on Statistical Machine Translation*, WMT, pages 12–58, Baltimore, Maryland.

Bojar, Ondřej, Rajen Chatterjee, Christian Federmann, Barry Haddow, Matthias Huck, Chris Hokamp, Philipp Koehn, Varvara Logacheva, Christof Monz, Matteo Negri, Matt Post, Carolina Scarton, Lucia Specia, and Marco Turchi. 2015. Findings of the 2015 Workshop on Statistical Machine Translation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 1–46, Lisbon, Portugal, September.

Bojar, Ondřej, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Matthias Huck, Antonio Jimeno Yepes, Philipp Koehn, Varvara Logacheva, Christof Monz, Matteo Negri, Aurelie Neveol, Mariana Neves, Martin Popel, Matt Post, Raphael Rubino, Carolina Scarton, Lucia Specia, Marco Turchi, Karin Verspoor, and Marcos Zampieri. 2016. Findings of the 2016 conference on machine translation. In *Proceedings of the First Conference on Machine Translation*, pages 131–198, Berlin, Germany, August.

Bojar, Ondřej, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Shujian Huang, Matthias Huck, Philipp Koehn, Qun Liu, Varvara Logacheva, Christof Monz, Matteo Negri, Matt Post, Raphael Rubino, Lucia Specia, and Marco Turchi. 2017. Findings of the 2017 conference on machine translation (wmt17). In *Proceedings of the Second Conference on Machine Translation, Volume 2: Shared Task Papers*, pages 169–214, Copenhagen, Denmark, September.

Cho, Kyunghyun, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*.

Cohn, Trevor and Lucia Specia. 2013. Modelling annotator bias with multi-task gaussian processes: An application to machine translation quality estimation. In *51st Annual Meeting of the Association for Computational Linguistics*, ACL, pages 32–42, Sofia, Bulgaria.

Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Fonseca, Erick, Lisa Yankovskaya, André FT Martins, Mark Fishel, and Christian Federmann. 2019. Findings of the WMT 2019 Shared Tasks on Quality Estimation. In *Proceedings of the Fourth Conference on Machine Translation (Volume 3: Shared Task Papers, Day 2)*, pages 1–10.

Graham, Yvette. 2015. Improving evaluation of machine translation quality estimation. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1804–1813, Beijing, China, July. Association for Computational Linguistics.

Howard, Jeremy and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification. *arXiv preprint arXiv:1801.06146*.

Ive, Julia, Frédéric Blain, and Lucia Specia. 2018. DeepQuest: a framework for neural-based quality estimation. In *Proceedings of COLING 2018, the 27th International Conference on Computational Linguistics: Technical Papers*, Santa Fe, new Mexico.

Kepler, Fábio, Jonay Trénous, Marcos Treviso, Miguel Vera, António Góis, M Amin Farajian, António V Lopes, and André FT Martins. 2019. Unbabel's participation in the wmt19 translation quality estimation shared task. *arXiv preprint arXiv:1907.10352*.

Kim, Hyun, Jong-Hyeok Lee, and Seung-Hoon Na. 2017. Predictor-estimator using multilevel task learning with stack propagation for neural quality estimation. In *Proceedings of the Second Conference on Machine Translation*, pages 562–568.

Kingma, Diederik P and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Lample, Guillaume and Alexis Conneau. 2019. Cross-lingual language model pretraining. *arXiv preprint arXiv:1901.07291*.

Lommel, Arle Richard, Aljoscha Burchardt, and Hans Uszkoreit. 2014. Multidimensional quality metrics (MQM): A framework for declaring and describing translation quality metrics. *Tradumàtica: tecnologies de la traducció*, 0(12):455–463, 12.

Nair, Vinod and Geoffrey E Hinton. 2010. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pages 807–814.

Pan, Sinno Jialin, Qiang Yang, et al. 2010. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359.

Quirk, Christopher. 2004. Training a sentence-level machine translation confidence measure. In *LREC*. Citeseer.

Shah, Kashif and Lucia Specia. 2016. Large-scale multitask learning for machine translation quality estimation. In *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 558–567, San Diego, California.

Snover, Matthew, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. 2006. A study of translation edit rate with targeted human annotation. In *Proceedings of association for machine translation in the Americas*, volume 200.

Specia, Lucia, Marco Turchi, Nicola Cancedda, Marc Dymetman, and Nello Cristianini. 2009. Estimating the Sentence-Level Quality of Machine Translation Systems. In *13th Annual Conference of the European Association for Machine Translation*, EAMT, pages 28–37, Barcelona, Spain.

Specia, Lucia, Gustavo Paetzold, and Carolina Scarton. 2015. Multi-level translation quality prediction with quest++. *Proceedings of ACL-IJCNLP 2015 System Demonstrations*, pages 115–120.

Specia, Lucia, Kim Harris, Frédéric Blain, Aljoscha Burchardt, Viviven Macketanz, Inguna Skadina, Matteo Negri, , and Marco Turchi. 2017. Translation quality and productivity: A study on rich morphology languages. In *Machine Translation Summit XVI*, pages 55–71, Nagoya, Japan.

Specia, Lucia, Frédéric Blain, Varvara Logacheva, Ramón Astudillo, and André F. T. Martins. 2018. findings of the wmt 2018 shared task on quality estimation. In *Proceedings of the Third Conference on Machine Translation, Volume 2: Shared Task Papers*, pages 702–722, Belgium, Brussels, October.

Specia, Lucia. 2011. Exploiting objective annotations for measuring translation post-editing effort. In *Proceedings of the 15th Conference of the European Association for Machine Translation*, pages 73–80.

Ueffing, Nicola and Hermann Ney. 2007. Word-level confidence estimation for machine translation. *Computational Linguistics*, 33(1):9–40.

Williams, Evan James. 1959. *Regression Analysis*, volume 14. Wiley, New York, USA.

Yosinski, Jason, Jeff Clune, Yoshua Bengio, and Hod Lipson. 2014. How transferable are features in deep neural networks? In *Advances in neural information processing systems*, pages 3320–3328.

Zoph, Barret, Deniz Yuret, Jonathan May, and Kevin Knight. 2016. Transfer learning for low-resource neural machine translation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1568–1575.