Direct Speech-to-Text Translation Models as Students of Text-to-Text Models

Marco Gaido*
Fondazione Bruno Kessler, Università di Trento

Matteo Negri**
Fondazione Bruno Kessler

Marco Turchi†
Fondazione Bruno Kessler

Direct speech-to-text translation (ST) is an emerging approach that consists in performing the ST task with a single neural model. Although this paradigm comes with the promise to outperform the traditional pipeline systems, its rise is still limited by the paucity of speech-translation paired corpora compared to the large amount of speech-transcript and parallel bilingual corpora available to train previous solutions. As such, the research community focused on techniques to transfer knowledge from automatic speech recognition (ASR) and machine translation (MT) models trained on huge datasets. In this paper, we extend and integrate our recent work (Gaido et al. 2020b) analysing the best performing approach to transfer learning from MT, which is represented by knowledge distillation (KD) in sequence-to-sequence models. After the comparison of the different KD methods to understand which one is the most effective, we extend our previous analysis of the effects – both in terms of benefits and drawbacks – to different language pairs in high-resource conditions, ensuring the generalisability of our findings. Altogether, these extensions complement and complete our investigation on KD for speech translation leading to the following overall findings: i) the best training recipe involves a word-level KD training followed by a fine-tuning step on the ST task, ii) word-level KD from MT can be detrimental for gender translation and can lead to output truncation (though these problems are alleviated by the fine-tuning on the ST task), and iii) the quality of the ST student model strongly depends on the quality of the MT teacher model, although the correlation is not linear.

1. Introduction

The translation of speech segments into their textual content in a different language is referred in literature as the task of speech-to-text translation (ST). ST involves two logical sub-tasks: automatic speech recognition (ASR), i.e. the modality conversion from the source audio into text, and machine translation (MT), i.e. the translation of the transcribed text into the target language. As a natural consequence of this logical division, the first ST architectures were based on a *pipeline* (or *cascade*) approach that combined an ASR and an MT model, where the output of the ASR system constituted the input of the MT system (Stentiford and Steer 1988; Waibel et al. 1991).

* FBK MT Unit - Via Sommarive, 18, 38123 Povo (TN), Italy. E-mail: mgaido@fbk.eu
** E-mail: negri@fbk.eu
† E-mail: turchi@fbk.eu

The recent rise of deep neural networks (LeCun, Bengio, and Hinton 2015) not only revolutionised the ASR and MT fields but also suggested a *direct* (or *end-to-end*) approach to ST, in which a single deep network performs the whole task at once, dealing both with the modality and language transformation (Bérard et al. 2016; Weiss et al. 2017). This paradigm has been proposed to overcome the limitations of the cascade solution, namely: *i)* the impact of ASR errors on the MT system ability to understand the content – as MT has no cues to recover from them – leading to error propagation (Ruiz and Federico 2014), *ii)* the information loss (e.g. prosody) caused by the mediated access (via the transcript) to the input audio, and *iii)* the higher latency introduced by the sequential inference of the two models with respect to a single model architecture.

Despite the above-mentioned advantages, direct models have not yet substituted the cascade solutions in industrial/real-world applications. The main reason lies in the lower quality of the generated translations, a performance gap that has been significantly reduced (if not closed) only recently, with direct models reaching comparable (sometimes better) quality of state-of-the-art cascade solutions and producing outputs that are indistinguishable for the end user (Bentivogli et al. 2021). The initial gap was mainly caused by the scarcity of parallel *audio-translation* corpora for ST, while plenty of ASR and MT parallel data are available. Research has overcome this shortage by means of data augmentation techniques (Jia et al. 2019; Bahar et al. 2019; Nguyen et al. 2020) and by transferring to the ST models the knowledge acquired by ASR/MT models trained on the two data-rich sub-tasks (Weiss et al. 2017; Anastasopoulos and Chiang 2018; Bérard et al. 2018; Bansal et al. 2019; Liu et al. 2019; Bahar, Bieschke, and Ney 2019). Along this latter direction, while the weights of an ASR model are commonly used to initialize the ST encoder (Bahar, Bieschke, and Ney 2019), the initialization of the decoder with the weights of an MT model has not consistently proved to be beneficial (Bahar, Bieschke, and Ney 2019; Gaido et al. 2020a; Inaguma et al. 2020). As an alternative to decoder initialization, another method to transfer knowledge from MT has been proposed and successfully exploited (Liu et al. 2019): knowledge distillation (KD).

Following this promising line of research, in this paper we extend in several ways the analysis by (Gaido et al. 2020b) on KD for direct ST models. Specifically, we complement the analysis on the specific knowledge learned by the ST model as well as of the drawbacks introduced by KD in rich data conditions by validating that the previous findings generalise to different language pairs. As such, we show on English→{French, German, Italian} that distilling knowledge at word-level yields the highest performance but often prevents the ST model from producing the correct gender of the speaker and to translate long utterances made of more than one sentence. Within this richer evaluation setting, we also confirm that a further fine-tuning without KD solves these issues retaining the quality improvements brought by KD. Finally, we analyse the correlation and dependency of the quality of ST student models with the quality of their MT teachers. Our experiments show that, regardless of the KD method adopted, a better MT teacher always leads to a better ST student, but the gains become lower at higher MT quality scores.

2. Background

2.1 Knowledge Distillation

KD has been introduced to transfer knowledge from a big model into a small, compressed one (Hinton, Vinyals, and Dean 2015). The goal is to have a small model –

named *student* in the KD learning procedure – that performs similarly to its big counterpart – named *teacher* – while being usable on low-resource devices (e.g. mobile phones). Specifically, the student is trained to learn to mimic the probability distribution of the teacher when processing the same input. This is obtained by using the probabilities generated by the teacher as reference when training the student, instead of the usual reference distribution, in which the correct label is assigned probability 1 and all the others 0. In practice, this means that the student is not trained to optimize the cross entropy loss function, but to minimize the distance between the probability distribution it generates and the one generated by the teacher. The distance between the two probability distributions is computed with the KL-divergence (Kullback and Leibler 1951), which is formally defined as:

$$KL(P||Q) = \sum P(x) * log\frac{P(x)}{Q(x)} \tag{1}$$

which measures the *closeness* of Q to P, i.e. how much information is lost when using Q to approximate P. In our case, defining $p(y|x)$ as the probability distribution over the target labels Y generated by the teacher for the input x and $q(y|x)$ the probability distribution generated by the student, Eq. 1 becomes:

$$\begin{aligned} KL(P||Q) &= \sum_{x \in X} \sum_{y \in Y} p(y|x) * log\frac{p(y|x)}{q(y|x)} \\ &= \sum_{x \in X} \sum_{y \in Y} p(y|x) * log(p(y|x)) - \sum_{x \in X} \sum_{y \in Y} p(y|x) * log(q(y|x)) \end{aligned} \tag{2}$$

Since the first term does not depend on the student output, the loss function that is optimized in knowledge distillation omits it and Eq. 2 becomes:

$$L(X) = -\sum_{x \in X} \sum_{y \in Y} p(y|x) * log(q(y|x)) \tag{3}$$

Notice that if we replace the teacher output distribution with the real target distribution, which is 1 for the correct target label y' and 0 for all the other target labels, we obtain the standard cross entropy loss:

$$L(X) = -\sum_{x \in X} log(q(y'|x)) \tag{4}$$

One of the reasons behind the success of KD has been individuated in the capability of the student model to learn from the *dark knowledge* (Hinton, Vinyals, and Dean 2015) of the teacher, defined as the information present in the teacher model that is not exposed by looking only at the final output, which considers only the most likely label. By learning to mimic the behavior of the teacher also for the less likely labels, the student is indirectly exposed to such internal knowledge. To increase the relevance and contribution of the dark knowledge, (Hinton, Vinyals, and Dean 2015) introduce

the hyper-parameter *temperature* (T) that smooths[1] the output distribution. With T, in particular, the $softmax$ operation converting the logits z_i into the corresponding probabilities p_i becomes:

$$p_i = \frac{e^{z_i/T}}{\sum(e^{z_i/T})} \tag{5}$$

When it is not stated otherwise, T is assumed to be set to 1, which corresponds to the standard $softmax$ operation without any smoothing factor.

2.2 Sequence-level Knowledge Distillation

KD has been proposed in the context of classification, where one label has to be predicted for every input. However, ST is a sequence-to-sequence task, so the output is not a single label but a sequence of variable length. Therefore, KD cannot be applied in its original form. For sequence-to-sequence tasks, (Kim and Rush 2016) proposed three different techniques to distill knowledge at sequence level: *i)* word-level KD, *ii)* sequence-level KD, and *iii)* sequence interpolation.

Word-level KD (henceforth `Word-KD`) is the most similar method to the original KD definition introduced in Section 2.1. In this case, the KL divergence between the teacher and student outputs is computed for every element (time-step) of the target sequence and the final distance is the sum of the divergences over all the elements (time-steps) of the sequence. Hence, Eq. 3 becomes:

$$L(X) = -\sum_{x \in X} \sum_{t \in [1, len(X)]} \sum_{y \in Y} p(y_t|x, y_0, ..., y_{t-1}) * log(q(y_t|x, y_0, ..., y_{t-1})) \tag{6}$$

Sequence-level KD (henceforth `Seq-KD`) consists in replacing the target reference (in our case the translation provided in the training corpora) with the sequence of tokens (in our case the automatic translation) generated by the teacher model. The loss function can be either the cross entropy or one of its variants, such as the label smoothed cross entropy.

Sequence interpolation (henceforth `Seq-Inter`) relies as well on the predictions of the teacher model. In this case, though, the N most likely sequences resulting from the beam search are re-scored and the one with the highest similarity with the ground truth is chosen as surrogate reference. In the case of textual outputs, such as in MT and ST, the similarity with the ground truth is computed using the BLEU score (Papineni et al. 2002).

Finally, `Word-KD` can be combined with the other two methods, resulting in two additional possible methods: `Word-KD+Seq-KD` and `Word-KD+Seq-Inter`.

2.3 Knowledge Distillation in ST

KD has been applied to direct ST for motivations different from the originary model compression. (Liu et al. 2019) train a direct ST model with a MT teacher to transfer

1 Notice that this is true only for $T > 1$. If $T < 1$, the distribution is sharpened.

knowledge from the easier MT task,[2] in which models obtain better performance, and hence improve the quality of the resulting ST student model. (Gaido et al. 2020a; Papi et al. 2021), instead, leverage KD from an MT model trained on a large amount of data to distill into the ST student model information that such a model could not directly access because of the different input modality. All these works employ the `Word-KD` method. (Jia et al. 2019), instead, generate synthetic data by translating the transcripts of ASR corpora with an MT model. Although presented as a data augmentation method, this can also be interpreted as an application of the `Seq-KD` method, although the benefits of KD cannot be isolated from those due to the additional data.

3. Models

In this section, we describe the architectures and the parameters that were used in our experiments. For the sake of further easing the reproducibility of our work and facilitate building up on our work, the code is open source and can be found at `https://github.com/mgaido91/FBK-fairseq-ST`.

3.1 MT Architecture

Our MT models are plain Transformer (Vaswani et al. 2017) models with 6 Transformer encoder layers and 6 Transformer decoder layers. In the experiments discussed in Section 4, we use a small model with 512 hidden features and 8 attention heads in all attention layers and 1,024 hidden features in the feed-forward networks (FFNs) of the Transformer layers. In the other experiments, as they involve a larger amount of data, all these hyper-parameters are doubled.

3.2 ASR and ST Architecture

In our experiments, we use an architecture based on Transformer with some adaptations for the input modality (audio) that is different from that (text) for which Transformer has been introduced (Dong, Xu, and Xu 2018; Di Gangi et al. 2019). One of the key challenges of using Transformer for speech is represented by the higher length of the input sequence (usually ~10 times longer than the corresponding textual representation), because Transformer's memory requirements grow quadratically with the input sequence length. Thus, to avoid out-of-memory issues and enable trainings with speech sources, the input features are processed with two 2D convolutions, each having stride 2, that reduce the sequence length by a factor of four (Bérard et al. 2018; Di Gangi et al. 2019). This sequence is then fed to the Transformer encoder, whose self-attention layers are modified by biasing the attention matrix toward close elements with a logarithmic distance penalty (Di Gangi et al. 2019).

In the experiments of Section 4, we use a small model, with 256 hidden features and 4 attention heads in all attention layers and 1,024 hidden features in the FFNs of Transformer layers. The number of Transformer encoder layers is 8 and the number of Transformer decoder layers is 6.

In all other experiments, as we train on larger corpora, following (Gaido et al. 2020a) we use 11 Transformer encoder layers and 4 Transformer decoder layers for our ST

2 ST does not only involve translating from a source to a target language, but also recognising the speech content.

models, while the ASR models used for the pre-training have 8 Transformer encoder layers and 6 Transformer decoder layers. When loading the pre-trained encoder layers, the additional 3 layers of the ST model are randomly initialized and behave as adapter layers (Jia et al. 2019; Bahar, Bieschke, and Ney 2019). Moreover, we increase the size of the models that have 512 hidden features and 8 attention heads in the attention layers and 2,048 hidden features in the FFNs.

3.3 Training Parameters

In all our trainings we choose Adam (Kingma and Ba 2015) using betas $(0.9, 0.98)$ as optimizer and, in case the loss is not the KL-divergence, we use label smoothing (Szegedy et al. 2016) with smoothing factor 0.1. For ASR, the objective function also includes a Connectionist Temporal Classification (CTC) loss (Graves et al. 2006), which is summed to the cross entropy. The CTC is computed on the encoder output (with the transcripts as target), and its role is only to aid model convergence and improve the final quality of the model (Kim, Hori, and Watanabe 2017). In all trainings, the learning rate is increased linearly for $4,000$ updates, up to the value of $5 * 10^{-3}$, and then decays with the inverse square root policy. In the fine-tunings, instead, the learning rate is kept fixed and is $1 * 10^{-4}$. The dropout is set to 0.2.

Each mini-batch contains 8 samples but updates are delayed to reach an overall batch size of 512.[3] All our models are trained on K80 GPUs with 12 GB of RAM.

The input audio is pre-processed by extracting 40 features using Mel filter bank with overlapping windows of 25 ms and 10 ms step size. The extracted features are then normalized per speaker. This pre-processing is performed with XNMT (Neubig et al. 2018). Samples resulting in more than 2,000 vectors (i.e. longer than 20 s) are discarded to avoid excessive memory requirements at training time. Text, instead, is tokenized after punctuation normalization with Moses (Koehn et al. 2007) and segmented into sub-word units using 8,000 BPE (Sennrich, Haddow, and Birch 2016) merge rules, as suggested in (Di Gangi et al. 2020). The BPE merge rules are jointly learned on the two languages of the MT dataset.

4. Comparison of the KD Methods

As per (Gaido et al. 2020b), we compared the three KD methods in a controlled setting, using only the data from Librispeech (Kocabiyikoglu, Besacier, and Kraif 2018), which contains $132,553$ *(audio, transcript, translation)* triplets for the English→French language direction. The MT teacher model is trained on the *(transcript, translation)* pairs, the ASR model used to initialize the ST encoder on the *(audio, transcript)* pairs, and the ST model on the *(audio, translation)* pairs. Within this controlled setting, the benefits brought by KD on the ST students are not due to the indirect exposure to additional MT data, but to the easiness to learn by extracting knowledge from the better performing MT teacher.

3 The value of the update frequency hyper-parameter depends on the number of GPUs used in the training, as the final batch size is given by the product of the mini-batch size, the number of GPUs, and the update frequency. We trained our models either on 4 GPUs (with update frequency 16) or on 8 GPUs (with update frequency 8).

4.1 Word-KD Computation

4.1.1 Output Distribution Truncation

The definition of the `Word-KD` method exposed in Section 2.2 implies that the whole output distribution of the teacher model is compared with the whole output distribution of the student. In practice, this is highly inefficient since pre-computing and storing the output probabilities for each token of each sequence requires huge storage capacity (e.g. with ~100,000 samples of average length 100 and 8,000 labels in the output distribution, we would need to store 80,000,000,000 floats, corresponding to more than ~320 GB of storage). On the other hand, re-computing the teacher target label at every iteration entails a forward pass on the teacher network for every input batch, leading to a significant increase in the training time.

Considering that the $softmax$ operation produces peaky outputs that tend to concentrate most of the probability distribution across up to 3-4 tokens, we hypothesize that truncating the output distribution and reducing the loss computation to only the K most likely labels can speed up the training without compromising the quality of the resulting model.

Table 1 reports the results for different K values. As the output is required to be a valid probability distribution, after the truncation the probabilities are re-scaled to sum up to 1. As per the formulated hypothesis based on the $softmax$ behavior, limiting the KL-divergence computation to a small number of labels does not impact performance. On the contrary, the best result is obtained with 8 labels. Indeed, predictions with very low probabilities are likely to be uninformative and noisy and do not carry useful information about the internal knowledge of the teacher. In light of these results, hereinafter all experiments with `Word-KD` assume that the KL-divergence is only computed setting $K = 8$, i.e. on the top 8 output labels of the teacher distribution.

Table 1
Results (BLEU score) with different K values, where K is the number of tokens considered for `Word-KD`.

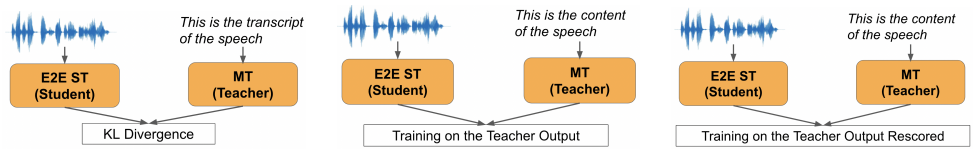
Top K	BLEU
4	16.43
8	**16.50**
64	16.37
1024	16.34

4.1.2 Temperature

As mentioned in Section 2.1, KD has been proposed with an hyper-parameter, the *temperature*, that controls the smoothness of the output distribution and increases/decreases the importance of the so-called *dark knowledge*. In our work, we tested multiple values aimed to smooth the probability distributions and favor the learning of such *dark knowledge*. According to the results shown in Table 2, the best BLEU score is achieved by setting the temperature to 1.0, which means by training without any smoothing factor. This finding suggests that ST models – as they need to learn a more complex task – have a limited capacity with respect to MT models and therefore focusing only on the mode of the MT model distributions is more convenient. Accordingly, in the following experiments we do not apply smoothing, by setting to 1.0 the temperature hyper-parameter.

Table 2
Results with different temperatures (T). All differences are statistically significant with $p = 0.05$.

T	BLEU
1.0	**16.50**
4.0	16.11
8.0	14.27



(a) Word-level KD. (b) Sequence-level KD. (c) Sequence interpolation.

Figure 1
Illustration of the KD methods.

4.2 Word-KD, Seq-KD, Seq-Inter and their Combination

We now compare the standard cross entropy loss – which we consider our baseline – with the KD methods described in Section 2.2 and summarized in Figure 1. The comparison is also carried out considering different combinations of such techniques. These can be performed in two ways: by applying both techniques together in the same training, or by first training with one technique and then fine-tuning the resulting ST model with the other. We also experimented with fine-tuning (FT) without KD after the application of a KD method. The results are reported in Table 3.

Table 3
Results of the small model on Librispeech with different KD methods and combining them in a single training or in consecutive trainings through a fine-tuning (FT). The "†" symbol indicates that improvements over `Word-KD` are statistically significant with $p = 0.05$.

	BLEU
Baseline	9.4
`Word-KD`	16.5
`Seq-KD`	13.4
`Seq-Inter`	13.3
`Seq-KD` + `Word-KD`	15.7
`Word-KD` + FT `Seq-KD`	16.7†
`Seq-KD` + FT `Word-KD`	**16.8**†
`Word-KD` + FT w/o KD	**16.8**†

Looking at the Baseline and the three KD techniques, we can conclude that all KD methods improve significantly over the Baseline, with gains that range from 3.9 to 7.1 BLEU points. Moreover, `Word-KD` is a clear winner among them, with a 3.1 BLEU margin over `Seq-KD`. Combining `Word-KD` and `Seq-KD` in a single training (`Seq-KD` + `Word-KD`) does not bring advantages; conversely, the result is worse (-0.8 BLEU) than the training with only `Word-KD`. The quality of the resulting model is instead improved when `Word-KD` and `Seq-KD` are applied sequentially, i.e. when a first training with either of them is followed by a fine-tuning with the other (see `Word-KD` + FT `Seq-KD`

and Seq-KD + FT Word-KD). Both solutions yield small gains of 0.2-0.3 BLEU points over the Word-KD method alone. The same result is also obtained when training on Word-KD and fine-tuning on the ground truth references with label smoothed cross entropy, i.e. without KD (Word-KD + FT w/o KD).

Although they are in line with previous work on KD for ST from MT (Liu et al. 2019), our results do not confirm the trends shown in (Kim and Rush 2016), where KD is used to compress MT models. Indeed, in our case Word-KD is a clear winner. This suggests that the effectiveness of different KD methods in a sequence-to-sequence scenario varies depending on the peculiarities of the task.

5. Effects and Analysis in High Resource Conditions

Once defined the best KD practice, we validate its effects in a realistic, high-resource scenario, in which large parallel MT corpora are available, together with a considerable amount of speech hours with the corresponding transcripts (ASR data).[4] In this case, we train our models on three language pairs: English→French (en-fr), English→German (en-de) and English→Italian (en-it).

The MT data is a selection of the OPUS corpora (Tiedemann 2016), filtered using the cleaning utilities of ModernMT (Bertoldi et al. 2017). OPUS contains parallel sentences automatically extracted from the web. As such, their nature is very different from the ASR and ST data, which is based on recorded sessions (TED or European Parliament talks) or book/manual readings and whose utterances can contain more than one sentence. The ASR data include How2 (Sanabria et al. 2018), Librispeech (Kocabiyikoglu, Besacier, and Kraif 2018), Mozilla Common Voice,[5] TED-LIUM 3 (Hernandez et al. 2018), and MuST-C (Cattoni et al. 2021), which also constitutes our ST corpus together with Europarl-ST (Iranzo-Sánchez et al. 2020). Both ASR and ST trainings augment source audio with SpecAugment (Park et al. 2019), using 0.5 as probability, 13 as *frequency masking pars*, 20 as *time masking pars*, 2 as *frequency masking num*, and 2 as *time masking num*.

The ST training is carried out in three phases: *i)* a training with Word-KD on the ASR corpora, whose transcripts are translated into the target language with the MT model (i.e. a Word-KD + Seq-KD training on the ASR data); *ii)* a fine-tuning with Word-KD on the ST corpora; *iii)* a fine-tuning without KD, as per the best training method in our experiments in Section 4.2. The ST encoder is initialized with that of an ASR model trained on the above-listed corpora and scoring 10.2 WER on the MuST-C test set.

Table 4 reports the scores of the MT teachers, the ST students after the first two training steps (those including Word-KD), and the final ST score after the last fine-tuning without KD. These results emphasize the importance of the last fine-tuning without KD to obtain state-of-the-art results. Indeed, we can see that in the real scenario, where there is a significant mismatch between the MT and the ST training data, distilling the MT knowledge brings information and benefits that mostly emerge in the overall scores after the final fine-tuning. Our hypothesis is that the additional useful knowledge is counterbalanced by the negative effect of learning patters that are valid only for the MT training data. In the following, we study what these spurious patterns and negative effects are.

4 Although large ST corpora are not available, plenty of ASR and MT data can be collected to build models for real use cases.

5 https://voice.mozilla.org/

Table 4
BLEU scores of the MT teachers and ST students on the MuST-C tst-COMMON set for English→French,German,Italian.

Language Pair	MT Teacher	ST after `Word-KD` (step *ii)*)	ST after fine-tuning (step *iii)*)
en-de	32.1	25.8	27.6
en-fr	46.0	36.5	40.3
en-it	32.7	22.8	27.7

5.1 KD and Gender Translation

One possible explanation of the efficacy of the fine-tuning on the ST task after the `Word-KD` training is the consideration that the ST input (audio) contains information that is not present in the MT input (the corresponding transcript). As an example, the sentence *"I am a student"* can be translated in Italian either as *"Sono uno studente"* or as *"Sono una studentessa"* depending on the gender of the speaker. As this information is completely missing in the textual English input, a MT model is likely to produce the more frequent masculine forms with a representational harm for women [Savoldi et al. 2021], while in the audio the speaker pitch can be used as a gender cue to disambiguate the correct form. Although in general biological features should not be considered as gender cues,[6] our dataset (MuST-C) contains a strong correlation between speakers' vocal characteristics and gender forms in the reference translations, so ST models can learn and leverage this gender cue in our setting.

We validate the hypothesis by testing our models on the Category 1 of MuST-SHE (Bentivogli et al. 2020), which contains *(audio, translation)* pairs in which gender-marked terms related to the speaker are annotated to evaluate system's ability to produce correct gender forms in the translation. As a baseline, we report both the ST system developed by (Bentivogli et al. 2020) – where target text is represented at character level – and the BPE-based system by (Gaido et al. 2021), as they demonstrated that target-text segmentation is an important factor for systems' ability to translate gender and our systems segment target text with BPE, as this text segmentation method leads to the best translation quality. We measure the ability in translating gender with gender accuracy (Gaido et al. 2020c), i.e. the percentage of correct gender realizations among the words produced by the system and annotated in MuST-SHE.

The results are reported in Table 5 for two language pairs English→French/Italian. First, we can notice that fine-tuning on ST data indeed improves gender accuracy of the feminine forms from 20.9-26.9% to 33.6-32.6% respectively on en-it and en-fr, reducing the bias towards generating masculine forms. Second, the gap with a BPE-based ST system (`Base BPE ST`) is closed (en-it – 33.6% vs 33.2%) or becomes small (en-fr – 32.3% vs 37.2%), so the fine-tuning seems to completely solve the limitation of the ST student compared to a normal ST system. The gap with the ST systems by (Bentivogli et al. 2020) is still large (33.6% vs 49.5% on en-it, 32.3% vs 46.5% on en-fr), but it is motivated by the different text segmentation (char vs BPE). The study of hybrid solutions that go beyond the trade-off between the translation quality of BPE and the

6 The adoption of physical cues can lead to reductionist gender classifications [Zimman 2020] and be harmful for a diverse range of users.

gender accuracy of char-based segmentation is left as topic of other works [Gaido et al. 2021] and future research.

Table 5
BLEU score and Gender Accuracy on Category 1F (female speakers) and 1M (male speakers) of the MuST-SHE test set.

	BLEU	Female Gender Acc.	Male Gender Acc.
en-it			
Base Char ST (Bentivogli et al. 2020)	21.5	**49.5%**	87.2%
Base BPE ST (Gaido et al. 2021)	21.8	33.2%	**88.5%**
MT	**33.6**	16.3%	**88.5%**
`Seq-KD` + `Word-KD` + FT `Word-KD`	23.6	20.9%	84.9%
+ FT w/o KD	27.5	33.6%	80.5%
en-fr			
Base Char ST (Bentivogli et al. 2020)	27.9	**46.3%**	86.2%
Base BPE ST (Gaido et al. 2021)	25.9	37.2%	75.4%
MT	**39.6**	16.2%	**89.6%**
`Seq-KD` + `Word-KD` + FT `Word-KD`	32.0	26.9%	79.4%
+ FT w/o KD	34.3	32.3%	79.6%

All in all, the experiments show that the final fine-tuning mitigates the additional gender bias introduced by distilling knowledge from an MT teacher. However, the better gender translation alone does not explain the huge fine-tuning gains. As such, the next section describes the other negative effects of KD, detected via a manual analysis.

5.2 Other KD Negative Effects

We conducted a manual analysis on the en-it outputs, as en-it shows the highest gain (+4.9 BLEU, while en-fr has a +3.8 BLEU and en-de a +1.8 BLEU improvement – see Table 4). In particular, we selected and inspected the samples with the highest TER (Snover et al. 2006) gains after fine-tuning. This analysis revealed two main types of output improvements.

Avoid Truncation. The ST student often generates only the first sentence of an utterance and terminates the generation after it, regardless of whether the utterance really contains a single sentence or more than one. In this second case, hence, the output turns out to be truncated. Most likely, the root cause can be attributed to the nature of the data the MT teacher is trained on: indeed, MT corpora contain mostly parallel sentences and rarely a sample contains more than one sentence. As such, the MT teacher (and, in turn, its ST student) learn to terminate the sentence after the dot. Fine-tuning on the ST task, however, solves the issue: upon manual inspection, none of the outputs of the fine-tuned model exhibits truncations.

Verbal Tense and Lexical Choices. The ST student often chooses verbal tenses that are more common and less accurate. For instance, *"That meant I was going to be on television"* has been translated by the ST student as *"Questo significava che **stavo andando** in tv"*. Although it might be considered acceptable in a colloquial scenario, this translation is grammatically wrong. The fine-tuned model, instead, produces the correct translation with the grammatically-correct verbal tense *"Questo significava che **sarei andata** in televisione"*. Similarly, in some cases the ST student prefers common, generic words. For instance, *"She has taken a **course** in a **business school**, and she has become a veterinary doctor"* should be translated as *"Ha seguito un **corso** in una **scuola di business**, ed ÃÍ diventata una veterinaria"*. However, the ST student produces *lezione* (lesson) instead of

corso and *economia* (economics) instead of *scuola di business*. After fine-tuning, the models uses the correct terms *corso* and *business school*. Though important in terms of final score, these improvements may be also considered as an adaptation to a different domain and linguistic style (less colloquial), mostly due to the domain mismatch between the MT training data (web-crawled sentence pairs) and the ST data (TED talks).

As mentioned, the fine-tuning enhancements are mostly adaptations to the ST data and domain, which have peculiarities that differentiate them for the MT corpus used to train the MT teacher. This explains also the reason why the gains obtained with the fine-tuning are smaller in Section 4.2, where the MT and ST data coincide.

6. The Importance of Teacher Quality

So far we analyzed *what* the ST student learns from the MT teacher. However, we have not yet addressed the question: *how much* does the ST student learn from the MT teacher? How important is the quality of the MT teacher for the ST student quality? To answer these questions, we experimented using MT teachers of different quality (controlled by adding/removing data) to train ST students on the MuST-C en-it section, the same used in our previous analysis. We tested both `Word-KD` and `Seq-KD` to understand whether the quality of the teacher is a factor to be considered when choosing the KD method, e.g. whether with low-performing teachers one method is preferable, while with strong teachers the other one is superior.

We consider four teachers with different quality levels and the resulting teacher quality is controlled by sampling the training data. In particular, the best teacher (scoring 32.7 BLEU) is trained on the whole Opus corpus (60M sentence pairs). Then 10M, 1M and 250K (the size of the MuST-C dataset) sentences are sampled to define the training sets for the other three teachers, ensuring that all the sentences included in one training set are also present in the bigger datasets. The teachers trained on these smaller datasets score respectively 30.1, 26.1, and 20.3 BLEU. Unsurprisingly, the score of the MT system trained on the MuST-C dataset (28 BLEU) is significantly higher than the results of the MT models trained on a similar amount of out-of-domain data, and we need to increase by 40 times the size of the training data to obtain better scores. Although the scores are relatively low, this represents a normal working condition when using KD as a source of potentially useful external knowledge, as MT models are usually trained on large generic training corpora.

Looking at Figure 2, first we can confirm the intuition that a better teacher leads to a better student, although the students' training set is the same and the margin with the teacher is huge even with the worst teacher (+3.7 BLEU). Second, we can notice that the student is able to only partially learn the additional knowledge of the teacher: the gap between the MT teacher and the ST student's quality increases with the teacher quality and the ST student BLEU score has not a linear dependency with the teacher's BLEU, as the benefits become smaller at higher BLEU scores (the `Word-KD` student gains only 0.3 BLEU when the teacher improves from 30.1 to 32.7). We can conclude that the student is able to learn only part of the teacher's knowledge and the lower scores are not only due to a lower capacity of the student architecture, since the student has a large margin of improvement even with bad teachers, but improves significantly with a better teacher.

Finally, the comparison of `Word-KD` and `Seq-KD` results in similar trends and scores. The two methods behave similarly both with low and high quality teachers and they show the same performance. Indeed, the very small BLEU differences can be ascribed to statistical fluctuations and one method is not always better than the other. These results do not confirm the superiority of `Word-KD` shown in Section 4.2, but the

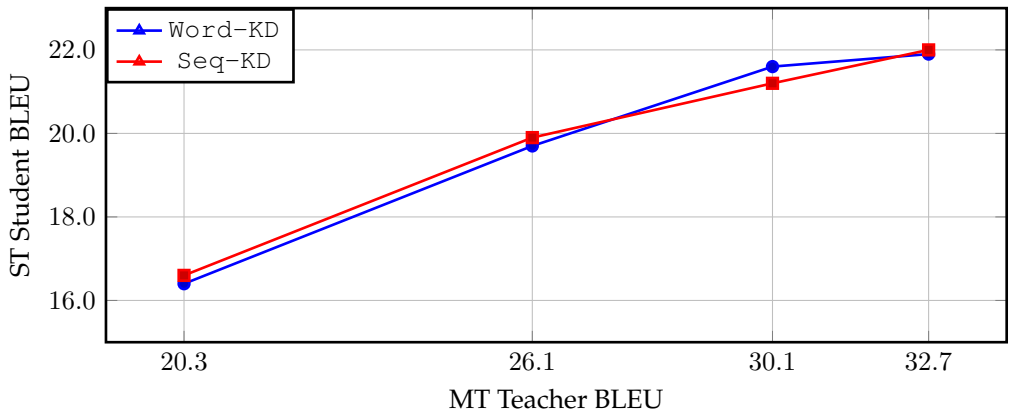


Figure 2
ST student performance (y axis – BLEU score) according to the MT teacher quality (x axis – BLEU score), when using `Word-KD` and `Seq-KD`, on the MuST-C en-it dataset.

difference can be explained with the different setting and scenario: in Section 4.2 the training set of the MT teacher is the same set on which the ST student is trained, while here the MT teacher is trained on different, out-of-domain corpora.

All in all, this analysis indicates that only part of the knowledge of the teacher can be learned by the student. Future research might try and explain which information can be learned by the student to provide insights on methods to create models that are better teacher as they focus on what can be learned by student models or to understand how to inject in the student the knowledge of the teacher that current KD methods do not allow to learn.

7. Conclusions

In the wake of previous preliminary work showing promising results obtained by distilling knowledge from an MT model to improve direct ST models, in this study we conducted a more systematic and meticulous analysis on the application of KD techniques to train an ST system. First, we compared the methods proposed in literature to distill knowledge in sequence-to-sequence models, as MT and ST systems are. Our experiments show the superiority of the `Word-KD` technique and the importance of fine-tuning an ST student on the ST data without KD. Second, we studied and showed the benefits and limitations introduced by distilling knowledge from an MT teacher. The different modality and lower information richness of the input also lead to limitations and drawbacks – such as an increased gender bias in gender-marked words that are related to the speaker, and sentence truncation and omission in multi-sentential utterances – that can be overcome with the simple above-mentioned fine-tuning without KD. Third, we demonstrated that the quality of the MT teacher is essential to have good ST systems and that a better MT teacher leads to a better ST student, although the student gains tends to saturate when the teacher scores are high. Overall, our results show that distilling knowledge from MT is a good knowledge transfer technique, which allows to benefit from the abundance of parallel textual data in the ST task. However, it requires some adroitness, as shown by the importance of a KD-independent fine-tuning to solve

the undesirable side-effect of learning behaviors of the MT teacher that can be harmful for the task at hand.

Acknowledgments

This work is part of the "End-to-end Spoken Language Translation in Rich Data Conditions" project,[7] which has been financially supported by an Amazon AWS ML Grant.

References

Anastasopoulos, Antonios and David Chiang. 2018. Tied Multitask Learning for Neural Speech Translation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 82–91, New Orleans, Louisiana, June.

Bahar, Parnia, Tobias Bieschke, and Hermann Ney. 2019. A Comparative Study on End-to-end Speech to Text Translation. In *Proceedings of International Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pages 792–799, Sentosa, Singapore, December.

Bahar, Parnia, Albert Zeyer, Ralf Schlüter, and Hermann Ney. 2019. On Using SpecAugment for End-to-End Speech Translation. In *Proceedings of the International Workshop on Spoken Language Translation (IWSLT)*, Hong Kong, China, November.

Bansal, Sameer, Herman Kamper, Karen Livescu, Adam Lopez, and Sharon Goldwater. 2019. Pre-training on High-resource Speech Recognition Improves Low-resource Speech-to-text Translation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 58–68, Minneapolis, Minnesota, June.

Bentivogli, Luisa, Mauro Cettolo, Marco Gaido, Alina Karakanta, Alberto Martinelli, Matteo Negri, and Marco Turchi. 2021. Cascade versus Direct Speech Translation: Do the Differences Still Make a Difference? In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics*, Virtual, August.

Bentivogli, Luisa, Beatrice Savoldi, Matteo Negri, Mattia A. Di Gangi, Roldano Cattoni, and Marco Turchi. 2020. Gender in danger? Evaluating Speech Translation Technology on the MuST-SHE Corpus. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL 2020)*, pages 6923–6933, Virtual, July.

Bérard, Alexandre, Laurent Besacier, Ali Can Kocabiyikoglu, and Olivier Pietquin. 2018. End-to-End Automatic Speech Translation of Audiobooks. In *Proceedings of the 2018 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 6224–6228, Calgary, Alberta, Canada, April.

Bérard, Alexandre, Olivier Pietquin, Christophe Servan, and Laurent Besacier. 2016. Listen and Translate: A Proof of Concept for End-to-End Speech-to-Text Translation. In *NIPS Workshop on end-to-end learning for speech and audio processing*, Barcelona, Spain, December.

Bertoldi, Nicola, Roldano Cattoni, Mauro Cettolo, et al. 2017. MMT: New Open Source MT for the Translation Industry. In *Proceedings of the 20th Annual Conference of the European Association for Machine Translation (EAMT)*, pages 86–91, Prague, Czech Republic, May.

Cattoni, Roldano, Mattia A. Di Gangi, Luisa Bentivogli, Matteo Negri, and Marco Turchi. 2021. MuST-C: A multilingual corpus for end-to-end speech translation. *Computer Speech & Language*, 66:101–155.

Di Gangi, Mattia A., Marco Gaido, Matteo Negri, and Marco Turchi. 2020. On Target Segmentation for Direct Speech Translation. In *Proceedings of the 14th Conference of the Association for Machine Translation in the Americas (Volume 1: Research Track)*, pages 137–150, Virtual, October. Association for Machine Translation in the Americas.

Di Gangi, Mattia Antonino, Matteo Negri, Roldano Cattoni, Roberto Dessi, and Marco Turchi. 2019. Enhancing Transformer for End-to-end Speech-to-Text Translation. In *Proceedings of Machine Translation Summit XVII*, pages 21–31, Dublin, Ireland, August.

Dong, Linhao, Shuang Xu, and Bo Xu. 2018. Speech-transformer: A no-recurrence sequence-to-sequence model for speech recognition. In *Proceedings of the 2018 IEEE*

7 https://ict.fbk.eu/units-hlt-mt-e2eslt/

International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 5884–5888, Calgary, Alberta, Canada, April.

Gaido, Marco, Mattia A. Di Gangi, Matteo Negri, and Marco Turchi. 2020a. End-to-End Speech-Translation with Knowledge Distillation: FBK@IWSLT2020. In *Proceedings of the 17th International Conference on Spoken Language Translation*, pages 80–88, Virtual, July.

Gaido, Marco, Mattia Antonino Di Gangi, Matteo Negri, and Marco Turchi. 2020b. On Knowledge Distillation for Direct Speech Translation. In *Proceedings of the Seventh Italian Conference on Computational Linguistics, (CLiC-it 2020)*, volume 2769 of *CEUR Workshop Proceedings*, Bologna, Italy, March 1-3.

Gaido, Marco, Beatrice Savoldi, Luisa Bentivogli, Matteo Negri, and Marco Turchi. 2020c. Breeding Gender-aware Direct Speech Translation Systems. In *Proceedings of the 28th International Conference on Computational Linguistics (COLING 2020)*, Virtual, December.

Gaido, Marco, Beatrice Savoldi, Luisa Bentivogli, Matteo Negri, and Marco Turchi. 2021. How to Split: the Effect of Word Segmentation on Gender Bias in Speech Translation. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3576–3589, Online, August. Association for Computational Linguistics.

Graves, Alex, Santiago Fernández, Faustino J. Gomez, and Jürgen Schmidhuber. 2006. Connectionist Temporal Classification: Labelling Unsegmented Sequence Data with Recurrent Neural Networks. In *Proceedings of the 23rd international conference on Machine learning (ICML)*, pages 369–376, Pittsburgh, Pennsylvania, June.

Hernandez, François, Vincent Nguyen, Sahar Ghannay, Natalia Tomashenko, and Yannick Estève. 2018. TED-LIUM 3: Twice as Much Data and Corpus Repartition for Experiments on Speaker Adaptation. In *Proceedings of the Speech and Computer - 20th International Conference (SPECOM)*, pages 198–208, Leipzig, Germany, September.

Hinton, Geoffrey, Oriol Vinyals, and Jeff Dean. 2015. Distilling the Knowledge in a Neural Network. In *Proceedings of NIPS Deep Learning and Representation Learning Workshop*, Montréal, Canada.

Inaguma, Hirofumi, Shun Kiyono, Kevin Duh, Shigeki Karita, Nelson Yalta, Tomoki Hayashi, and Shinji Watanabe. 2020. ESPnet-ST: All-in-One Speech Translation Toolkit. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 302–311, Virtual, July.

Iranzo-Sánchez, Javier, Joan Albert Silvestre-Cerdà, Javier Jorge, Nahuel Roselló, Giménez. Adrià, Albert Sanchis, Jorge Civera, and Alfons Juan. 2020. Europarl-ST: A Multilingual Corpus For Speech Translation Of Parliamentary Debates. In *Proceedings of the 2020 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 8229–8233, Barcelona, Spain, May.

Jia, Ye, Melvin Johnson, Wolfgang Macherey, Ron J. Weiss, Yuan Cao, Chung-Cheng Chiu, Naveen Ari, Stella Laurenzo, and Yonghui Wu. 2019. Leveraging Weakly Supervised Data to Improve End-to-End Speech-to-Text Translation. In *Proceedings of the 2019 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 7180–7184, Brighton, UK, May.

Kim, Suyoun, Takaaki Hori, and Shinji Watanabe. 2017. Joint CTC-Attention based End-to-End Speech Recognition using Multi-task Learning. In *Proceedings of the 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4835–4839, New Orleans, Louisiana, March.

Kim, Yoon and Alexander M. Rush. 2016. Sequence-Level Knowledge Distillation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1317–1327, Austin, Texas, November.

Kingma, Diederik and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *Proceedings of 3rd International Conference on Learning Representations (ICLR)*, San Diego, California, May.

Kocabiyikoglu, Ali Can, Laurent Besacier, and Olivier Kraif. 2018. Augmenting librispeech with French translations: A multimodal corpus for direct speech translation evaluation. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan, May.

Koehn, Philipp, Hieu Hoang, Alexandra Birch, et al. 2007. Moses: Open Source Toolkit for Statistical Machine Translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 177–180, Prague, Czech Republic, June.

Kullback, Solomon and Richard Arthur Leibler. 1951. On information and sufficiency. *The Annals of Mathematical Statistics*, 22(1):79–86.

LeCun, Yann, Yoshua Bengio, and Geoffrey Hinton. 2015. Deep learning. *Nature*, 521(7553):436–444, May.

Liu, Yuchen, Hao Xiong, Jiajun Zhang, Zhongjun He, Hua Wu, Haifeng Wang, and Chengqing Zong. 2019. End-to-End Speech Translation with Knowledge Distillation. In *Proceedings of Interspeech 2019, 20th Annual Conference of the International Speech Communication Association*, pages 1128–1132, Graz, Austria, September.

Neubig, Graham, Matthias Sperber, Xinyi Wang, Matthieu Felix, Austin Matthews, Sarguna Padmanabhan, Ye Qi, Devendra Sachan, Philip Arthur, Pierre Godard, John Hewitt, Rachid Riad, and Liming Wang. 2018. XNMT: The eXtensible Neural Machine Translation Toolkit. In *Proceedings of the 13th Conference of the Association for Machine Translation in the Americas*, pages 185–192, Boston, MA, March.

Nguyen, Thai-Son, Sebastian Stueker, Jan Niehues, and Alex Waibel. 2020. Improving Sequence-to-sequence Speech Recognition Training with On-the-fly Data Augmentation. In *Proceedings of the 2020 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Barcelona, Spain, May.

Papi, Sara, Marco Gaido, Matteo Negri, and Marco Turchi. 2021. Dealing with training and test segmentation mismatch: FBK@IWSLT2021. In *Proceedings of the 17th International Conference on Spoken Language Translation*, Virtual, July.

Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a Method for Automatic Evaluation of Machine Translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, July.

Park, Daniel S., William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin D. Cubuk, and Quoc V. Le. 2019. SpecAugment: A Simple Data Augmentation Method for Automatic Speech Recognition. In *Proceedings of Interspeech 2019, 20th Annual Conference of the International Speech Communication Association*, pages 2613–2617, Graz, Austria, September.

Ruiz, Nicholas and Marcello Federico. 2014. Assessing the Impact of Speech Recognition Errors on Machine Translation Quality. In *Proceedings of the 11th Conference of the Association for Machine Translation of the Americas*, pages 261–274, Vancouver, Canada, October.

Sanabria, Ramon, Ozan Caglayan, Shruti Palaskar, Desmond Elliott, Loïc Barrault, Lucia Specia, and Florian Metze. 2018. How2: A Large-scale Dataset For Multimodal Language Understanding. In *Proceedings of Visually Grounded Interaction and Language (ViGIL)*, Montréal, Canada, December.

Savoldi, Beatrice, Marco Gaido, Luisa Bentivogli, Matteo Negri, and Marco Turchi. 2021. Gender Bias in Machine Translation. *Transactions of the Association for Computational Linguistics*, 9:845–874, 08.

Sennrich, Rico, Barry Haddow, and Alexandra Birch. 2016. Neural Machine Translation of Rare Words with Subword Units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany, August. Association for Computational Linguistics.

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 the 7th Conference of the Association for Machine Translation in the Americas*, pages 223–231, Cambridge, Massachusetts, USA, August.

Stentiford, Fred W. M. and Martin G. Steer. 1988. Machine translation of speech. *British Telecom Technology Journal*, 6(2):116–122.

Szegedy, Christian, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. 2016. Rethinking the Inception Architecture for Computer Vision. In *Proceedings of 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2818–2826, Las Vegas, Nevada, United States, June.

Tiedemann, Jörg. 2016. Opus – parallel corpora for everyone. *Baltic Journal of Modern Computing*, page 384. Special Issue: Proceedings of the 19th Annual Conference of the European Association of Machine Translation (EAMT).

Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is All You Need. In *Proceedings of Advances in Neural Information Processing Systems 30 (NIPS)*, pages 5998–6008, Long Beach, California, December.

Waibel, Alex, Ajay N. Jain, Arthur E. McNair, Hiroaki Saito, Alexander G. Hauptmann, and Joe Tebelskis. 1991. JANUS: A Speech-to-Speech Translation System Using Connectionist and Symbolic Processing Strategies. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing, ICASSP 1991*, pages 793–796, Toronto, Canada, April.

Weiss, Ron J., Jan Chorowski, Navdeep Jaitly, Yonghui Wu, and Zhifeng Chen. 2017. Sequence-to-Sequence Models Can Directly Translate Foreign Speech. In *Proceedings of Interspeech 2017, 8th Annual Conference of the International Speech Communication Association*, pages 2625–2629, Stockholm, Sweden, August.

Zimman, Lal. 2020. Transgender language, transgender moment: Toward a trans linguistics. In Kira Hall and Rusty Barrett, editors, *The Oxford Handbook of Language and Sexuality*. Oxford University Press.