Scaling Down DeepSpeech
End-to-End Transfer Learning for Low-Resource Speech Recognition

Anonymous Authors[1]

Abstract

This paper investigates a simple end-to-end transfer learning approach for automatic speech recognition which bypasses the need for linguistic resources. We copy certain layers from a trained source model, initialize new layers for a target language, stitch the old and new layers together, and train the new layers via gradient descent. We additionally investigate effects of fine-tuning the original, copied layers.

When keeping the copied layers frozen during backpropagation, we observe largest improvements on average from transferring the bottom three fully-connected layers from Mozilla's English DeepSpeech. When additionally fine-tuning the copied layers, we observe the first four layers (i.e. 3 Fully Connected + 1 LSTM) show largest improvement to a new language on average. This finding is independent to the identity of the target language. Furthermore, we observe an across the board improvement when fine-tuning compared to keeping the copied layers frozen during backprop.

Finally, we investigate the quality of the embedding spaces learned by each layer of the original DeepSpeech model. We present results from linguistically motivated logistic regression tasks which are trained on top of feature-spaces at different model depths.

We present results for transfer-learning from English to the following twelve languages: German, French, Italian, Turkish, Catalan, Slovenian, Welsh, Irish, Breton, Tatar, Chuvash, and Kabyle. For most of these languages, these are the first ever published results on end-to-end automatic speech recognition. For Chuvash and Breton, these are the first ever published results on automatic speech recognition.

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

1. Introduction

We present results from transfer learning experiments on end-to-end ASR, using 12 target languages and the v0.3.0 release of Mozilla's pre-trained English DeepSpeech*. This model is a unidirectional variation of Baidu's first DeepSpeech paper (Hannun et al., 2014a), trained via CTC† loss on approximately 3,000 hours of English. The speech data for the 12 target languages was collected via Mozilla's Common Voice initiative.‡

End-to-end approaches for ASR are especially interesting for low-resource languages, because they do not require a pronunciation dictionary. However, all end-to-end training techniques (CTC (Hannun et al., 2014a; Amodei et al., 2016), Transducer (Rao et al., 2017; Battenberg et al., 2017), Attention (Chan et al., 2016; Bahdanau et al., 2016), Convolutional (Collobert et al., 2016; Pratap et al., 2018)) require thousands of hours of data to train, because the model must learn a mapping of audio directly to characters without any explicit intermediate linguistic representations. As such, end-to-end approaches are interesting for smaller datasets, but the typical training approach will not work. Transfer learning is an ideal solution for this problem because it it possible to leverage the knowledge from a source domain in order to bootstrap a model for the target task. Furthermore, the approach doesn't require access to a source domain dataset, but rather, a source domain model — which is often desirable due to copyright and privacy concerns.

The transfer learning technique in the current study is implemented as follows: (1) a certain number of trained layers are copied from Mozilla's English DeepSpeech model, then (2) a certain number of new layers are initialized, (3) the new layers are appended to the old layers, (4) the newly initialized layers are updated via gradient descent. The original layers are optionally fine-tuned via the same gradient signal from the target language. We experimented with ten different transfer scenarios per target language, varying (1) the number of layers copied (i.e. between one and five layers)

*https://github.com/mozilla/DeepSpeech/releases/tag/v0.3.0

†Connectionist Temporal Classification (Graves et al., 2006).

‡http://voice.mozilla.org

from English and (2) whether or not the copied layers are updated during backprop (i.e. fine-tuned vs. frozen).

2. Background

Speech recognition training techniques have been developed with high-resource languages in mind. The end-to-end approaches require upwards of 10,000 hours of transcribed audio (Hannun et al., 2014a), or even more with data-augmentation (Amodei et al., 2016). Nevertheless, low-resource languages started getting attention in the research community with IARPA's BABEL program (Gales et al., 2014), and various working approaches have been developed since then.

Previous low-resource work exists for end-to-end ASR (Bataev et al., 2018; Toshniwal et al., 2018; Rosenberg et al., 2017; Cui et al., 2017), as well as traditional DNN-HMM hybrid ASR (Cui et al., 2015; Grézl et al., 2014; Knill et al., 2013; Vu et al., 2014; Ghoshal et al., 2013). Typically these techniques involve leveraging data from as many languages as possible and then fine-tuning the model to one target language of interest. With regards to required resources, the end-to-end teacher-student approach in (Cui et al., 2017) is most similar to our approach in that it assumes only a trained source model and target language data. The algorithm in (Ghoshal et al., 2013) is most similar to our copy-paste transfer approach. While these approaches may work, they typically assume that the developer has access to multilingual data and enough GPUs to make use of that data — in practice this is often not the case.

3. Transfer Learning

This paper investigates the effects of transfer learning for low-resource ASR.[§] The goal of transfer learning is to use source-domain knowledge as an inductive bias during parameter estimation for a target-domain task (Pan et al., 2010). Source-domain knowledge can be found either *directly* in a source-domain dataset, or *indirectly* in a trained, source-domain model. While using a source dataset allows more fine-grained control of how bias is transferred (e.g. as in Multi-Task Learning), it is easier in practice to use a source model (given concerns about licensing, compute time, disk space, etc). Given the availability of Mozilla's pre-trained CTC model, we investigate best practices for transfer learning from this English model to multiple target languages.

Our current work is in many ways an ASR analog to the (Yosinski et al., 2014) work on transferability of ImageNet models and the work of (Oquab et al., 2014) on transferability of the mid-level layers for image recognition tasks. We investigate how similar (or dissimilar) transfer learning in speech recognition is compared to image classification.

The authors in (Yosinski et al., 2014) reached two major conclusions with regards to the transferability of features for image classification. Firstly, given the co-adaptation of layers in the source model, transferring layers isn't always as simple as deep vs. shallow. With regards to ImageNet, it was the case that multiple adjacent layers were co-adapted such that splitting them apart resulted in a degradation of performance on the target task. Secondly, higher level features may be too adjusted to the source domain to transfer well to the target task. We investigate both of these findings with regards to end-to-end ASR by transferring pre-trained layers from a source domain (i.e. English) to a target task (i.e. target language).

Speech recognition and image classification are fundamentally different tasks. All spoken languages are produced by the human vocal tract, and as such they have features in common. However, the smallest unit of spoken language (i.e. the phoneme) is not language-independent, and not all acoustic features are contrastive in all languages. For end-to-end ASR models, it is unknown where language-independent representations are encoded. This study is an initial foray into this research direction. We investigate the embedding space of each layer of an end-to-end ASR model by training logistic regressions of linguistically-motivated tasks on top of each layer.

4. Languages

The languages in our sample are typologically and genetically diverse. There are eight Indo-European languages, including three Romance languages (French, Italian, and Catalan), three Celtic languages (Welsh, Breton, and Irish), one Slavic language (Slovenian), and one Germanic language (German). In addition to this there are three Turkic languages (Turkish, Chuvash, and Tatar) and one Berber language (Kabyle). All of the languages with the exception of Tatar and Chuvash are written with the Latin alphabet. Both Tatar and Chuvash are written with the Cyrillic alphabet. Each language's writing system has a different number of characters, and all have characters which are not found in the English alphabet. All languages are written left to right.

In terms of morphological typology, they range from fusional (in the case of the Indo-European languages and Kabyle) to agglutinative (in the case of the Turkic languages). The languages also display a range of interesting morphophonological phenomena, such as initial consonant mutations in the Celtic languages — where the first consonant of a word changes depending on the previous morphosyntactic context — and vowel harmony in the Turkic languages — where vowels in an affix agree with the last

[§]For an overview of transfer learning in ASR, see (Deng & Li, 2013).

vowel in the stem. These phenomena are both kinds of long-range dependencies.

5. Experimental Set-up

5.1. Data

All the data used in this paper come from Mozilla's Common Voice initiative.[¶] The Common Voice data is crowd-sourced via a web app, where users read a visually presented sentence off their screen. The recording is then verified via a voting system in which other users mark a {transcript, utterance} pair as being correct or incorrect. The text is particularly messy, containing digits (i.e. [0-9]) as well as punctuation. The absolute values of the results presented here are therefore lower than state-of-the-art for some languages, because we are interested more in the relative effects of transfer learning across languages than SOTA itself. The text processing was identical across all languages to ensure as much comparability as possible. The audio itself is particularly noisy, often donated from smartphones in everyday noise environments. The language donation efforts are often spear-headed by a small group of individuals. As such, the early datasets are biased towards a small number of speakers, as can be seen in Table 1.

We made dataset splits (as can be seen in Table 1) such that one speaker's recordings are only present in one data split. This allows us to make a fair evaluation of speaker generalization, but as a result some training sets have very few speakers, making this an even more challenging scenario. The splits per language were made as close as possible to 80% train, 10% dev, and 10% test.

		Dataset Size					
		Audio Clips			Unique Speakers		
Language	**Code**	Dev	Test	Train	Dev	Test	Train
Slovenian	sl	110	213	728	1	12	3
Irish	ga	181	138	1001	4	12	6
Chuvash	cv	96	77	1023	4	12	5
Breton	br	163	170	1079	3	15	7
Turkish	tr	407	374	3771	32	89	32
Italian	it	627	734	5019	29	136	37
Welsh	cy	1235	1201	9547	51	153	75
Tatar	tt	1811	1164	11187	9	64	3
Catalan	ca	5460	5037	38995	286	777	313
French	fr	5083	4835	40907	237	837	249
Kabyle	kab	5452	4643	43223	31	169	63
German	de	7982	7897	65745	247	1029	318

Table 1. Number of audio clips and unique speakers per language per dataset split.

Results from this dataset are particularly interesting in that (1) the text and audio are challenging, (2) the range of languages is wider than most any openly available speech corpus, and (3) the amount of data per language is from very small (less than training 1,000 clips for Slovenian) to relatively large (over 65,000 clips for German), as can be seen clearly in Figure 1.

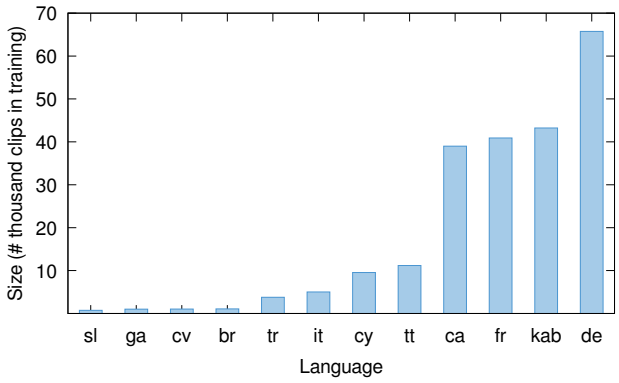


Figure 1. Number of audio clips per train split for each language. Audio clips for any individual speaker are only found in one split of the data (i.e. dev / test / train).

5.2. Model Architecture

All reported results were obtained with Mozilla's DeepSpeech toolkit[‖] — a free and open-source implementation of a variation of Baidu's first DeepSpeech paper (Hannun et al., 2014a). This architecture is an end-to-end sequence-to-sequence model trained via stochastic gradient descent with a CTC loss function. The model is six layers deep: three fully connected layers followed by a unidirectional LSTM layer followed by two more fully connected layers (c.f. Figure 2). All hidden layers have a dimensionality of 2048 and a clipped ReLU activation. The output layer has as many dimensions as characters in the alphabet of the target language (including any desired punctuation as well as the blank symbol used for CTC). The input layer accepts a vector of 19 spliced frames (9 past frames + 1 present frame + 9 future frames) with 26 MFCC features each (i.e. a single, 494-dimensional vector).

5.3. Training Hyperparameters

All models were trained with the following hyperparameters on a single GPU. We use a batch-size of 24 for train and 48 for development, a dropout rate of 20%, and a learning rate of 0.0001 with the ADAM optimizer.[**] The new, target-language layers were initialized via Xavier initialization (Glorot & Bengio, 2010). Early stopping was determined when the validation loss had either (1) increased over a window of 5 validation iterations, or (2) the 5th loss in a

[¶] http://voice.mozilla.org

[‖] https://github.com/mozilla/DeepSpeech

[**] For a complete list of ADAM hyperparameters: GitHub URL redacted for review.

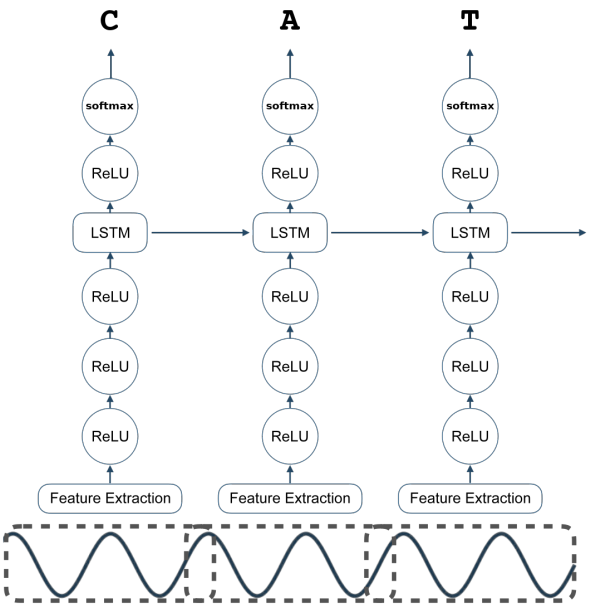


Figure 2. Architecture of Mozilla's DeepSpeech ASR system. A six-layer unidirectional CTC model, with one LSTM layer.

window had not improved more than a mean loss threshold of 0.5 *and* the window of 5 validation losses showed a standard deviation of < 0.5.

5.4. Language Model

A language model is used in beam-search decoding, as outlined in Hannun et al. (2014b). For each language we trained a trigram backoff language model from the raw text of Wikipedia using `kenlm` (Heafield, 2011). Singleton 2-grams and 3-grams were pruned to keep the compiled trie and binary ARPA files under 2G each.

6. Results

The following results will be presented as follows: First, we discuss results from experiments in which we *do not* update the parameters copied from a trained English model. These experiments we refer to in prose, tables, and figures as **frozen** parameter transfer. Second, we discuss results from experiments in which we *do* update parameters copied from an English model. We refer to these experiments as **fine-tuned** parameter transfer.

A priori, frozen transfer is interesting in that parameter estimation (per epoch) is faster than fine-tuning. Fewer calculations are required at every iteration of backpropagation with frozen transfer, because the gradient does not need to be calculated for the copied layers. From a model-interpretability point of view, frozen transfer offers us information as to what the different layers of the source model may be encoding. Results from frozen transfer will allow us to compare our findings to (Yosinski et al., 2014), investigating co-adaptation between layers as well as feature specificity to the source domain towards the output layer.

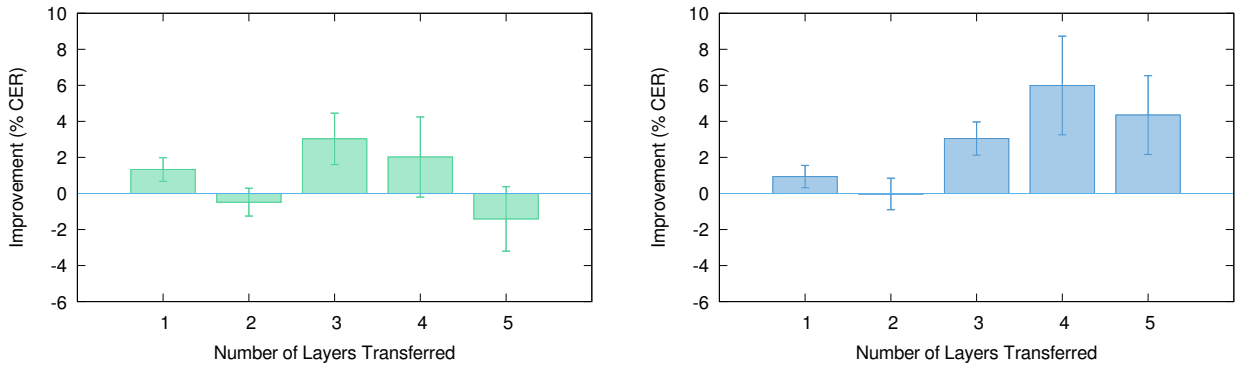
End-to-end speech recognition is by definition a mapping of audio directly onto characters. While in practice researchers often report results which incorporate some kind of language model during decoding, during parameter estimation all end-to-end models are trained to map audio onto characters. As such, the model should learn to perform complicated hierarchical tasks which are handled by independently trained sub-modules in traditional speech recognition. In traditional, "hybrid" speech recognition, various submodules were used to (1) identify context-dependent units (triphones) within continuous speech, (2) map those units onto higher-order sub-word parts (phonemes), and (3) map those sub-word parts onto written words in the target language. We might expect that end-to-end models are also capable of encoding these various higher-ordered representations, but we have very little idea as to where these representations are stored. Furthermore, some of these learned functions are more useful for multilingual transfer than others. If we can make an educated guess as to which layers of end-to-end models encode useful information, then transfer learning experiments may become much more efficient. Frozen transfer learning can give us a glimpse as to where these functions may have been learned in the source model.

Fine-tuned transfer learning is interesting in that it has been shown to perform out-perform frozen transfer on tasks like image recognition (Yosinski et al., 2014). As such, while fine-tuning will usually perform better in terms of accuracy, frozen transfer takes fewer compute resources to train, and may lends us insight as to where the models has learned more or less transferable features.

6.1. Frozen Parameter Transfer

For each target language we present six experiments from frozen parameter transfer in Table 2. We experimented by "slicing" the original model at different depths. The first experiment serves as our absolute baseline, where we train the entire six-layer CTC model from scratch without any transfer from English (c.f. column headed by 'None'). All transferred layers are contiguous, starting from the input layer. For example, the column with header '2' displays results where the first two layers (i.e. input layer + first hidden layer) are copied from English, and then four new layers are added on top. In all cases, the layers from English remain frozen during backprop, and new layers are updated with the gradient from the target language data. Early stopping was determined by a held-out validation set.

With regards to improvement over the baseline model, we find that for all of the twelve languages investigated here



(a) Frozen Transfer (b) Fine-Tuned Transfer

Figure 3. Mean and standard deviation for CER improvement for different # layers transferred from English.

	Character Error Rate					
	Number of Layers Copied from English					
Lang.	None	1	2	3	4	5
sl	23.35	23.93	25.30	18.87	**17.53**	26.24
ga	31.83	29.08	36.14	**27.22**	29.07	32.27
cv	48.10	46.13	47.83	38.00	**35.23**	42.88
br	21.47	19.17	20.76	18.33	**17.72**	21.03
tr	34.66	**32.98**	35.47	33.00	33.66	36.71
it	40.91	39.20	41.55	**38.16**	39.40	43.21
cy	34.15	32.46	33.93	**31.57**	35.26	36.56
tt	32.61	29.20	30.52	**27.37**	28.28	31.28
ca	38.01	**36.44**	38.70	36.51	42.26	47.96
fr	43.33	**43.30**	43.47	43.37	43.75	43.79
kab	25.76	25.57	25.97	**25.45**	27.77	29.28
de	43.76	44.48	44.08	43.70	43.77	**43.69**

Table 2. Frozen Transfer Learning Character-error rates (CER) for each language, in addition to a baseline trained from scratch on the target language data. Bolded values display best model per language. Shading indicates relative performance per language, with darker indicating better models.

at least one transfer learning experiment outperformed the model trained from scratch. After this general finding, interpreting these results from a first glance is difficult, but if we dig in a little, some trends do emerge. The results in Table 2 only show the best result in bold, but if we average the improvements over the baseline for all languages, we find a clearer visualization of these results in Figure 3(a). The bars show percent CER improvement averaged over all languages, and the tick marks display the standard deviation of this improvement.

We find that for frozen transfer learning from Mozilla's trained DeepSpeech model, two of the five transfer scenarios (i.e. transfer of layer [1] or layers [1-3]) show reliable improvement over a baseline trained from scratch. Two other transfer scenarios tend to lead to interference (i.e. layers [1-2] and layers [1-5]). Transfer of the first four layers (crucially including the LSTM layer) can lead to either improvement or interference.

Similar to the findings of (Yosinski et al., 2014), we suspect that transfer of the first and second layer fails due to co-adaptation between the second and third layers. We reach this conclusion based on the fact that transfer of the proceeding layer (i.e. layer 1), as well as transfer up to the following layer (i.e. layers [1-3]) both show improvement. If transfer at layer [2] failed due to over-specificity, then we should not find that transfer at layer [3] shows reliable improvement. Furthermore, we find that on average transfer up to layer [5] of DeepSpeech shows interference on average. We conclude that interference of transfer at this layer is due to specificity to the source domain (i.e. English).

We find that for languages with the largest training sets (Kabyle, French, Catalan, and German), the relative improvement from frozen transfer was smaller compared to languages with less data. The best-case improvement in CER for each language is presented (along with the results from fine-tuning experiments) in Figure 4.

6.2. Fine-Tuned Parameter Transfer

Now we present results from fine-tuning transfer learning experiments. In these experiments, the copied layers from English are updated via gradient descent (i.e. fine-tuned) according to the training data from the target language. Results from all fine-tuned experiments are presented in Table 3.

The first result that stands out (in contrast to Table 2) is that for almost all of the twelve languages, the best transfer scenario is when we copy the first four layers from a trained English model. Upon closer inspection, we find that for the three languages which are the exception to this rule (Irish, French, and German), the difference in improvement between the best result and the fourth layer is small (i.e. less than half of a percent in CER).

Lang.	Character Error Rate: Number of Layers Copied from English None	1	2	3	4	5
sl	23.35	21.65	26.44	19.09	**15.35**	17.96
ga	31.83	31.01	32.2	27.5	25.42	**24.98**
cv	48.1	47.1	44.58	42.75	**27.21**	31.94
br	21.47	19.16	20.01	18.06	**15.99**	18.42
tr	34.66	34.12	34.83	31.79	**27.55**	29.74
it	40.91	42.65	42.82	36.89	**33.63**	35.10
cy	34.15	31.91	33.63	30.13	**28.75**	30.38
tt	32.61	31.43	30.80	27.79	**26.42**	28.63
ca	38.01	35.21	39.02	35.26	**33.83**	36.41
fr	43.33	43.26	43.51	43.24	43.20	**43.19**
kab	25.76	25.5	26.83	25.25	**24.92**	25.28
de	43.76	43.69	43.62	**43.60**	43.76	43.69

Table 3. Fine-Tuned Transfer Learning Character-error rates (CER) for each language, in addition to a baseline trained from scratch on the target language data. Bolded values display best model per language. Shading indicates relative performance per language, with darker indicating better models.

Furthermore, looking at the averaged results (including standard deviation), we find that it is very much the case that the fourth layer usually leads to biggest improvements (c.f. Figure 3(b)). As with the frozen parameter transfer experiments, we find that even with fine-tuning the largest languages show smaller relative improvements compared to languages with less data (c.f. Figure 4). Frozen parameter transfer is typically considered appropriate when the target dataset is very small, and as such fine-tuning is prone to overfit a model with a large number of parameters (Yosinski et al., 2014), but our experiments show that fine-tuning always leads to an improvement over frozen transfer when using early stopping.

Summarizing findings from the importance of model depth for transfer (c.f. Figure 3), we find that (1) transfer of any segment of the first three (fully-connected) layers leads to near identical performance between frozen vs. fine-tuned, (2) transfer of the fourth (LSTM) layer leads to large stable gains when fine-tuned, but unreliable gains when frozen, and (3) transfer of the fifth (fully-connected) layer leads to reliable gains with fine-tuning, and interference when frozen.

6.3. Importance of Data Size for Transfer

We find a correlation between the size of the training dataset and the effectiveness of transfer learning (c.f. Figure 4). Generally speaking, the more data we have, the less transfer learning helps. However, we find that with fine-tuned transfer learning in particular, the performance from the largest datasets tends to the performance of the baseline. This is a very desirable feature, meaning that on average, transfer learning with fine-tuning will either improve or match the performance of a baseline, but performance should not drop significantly.

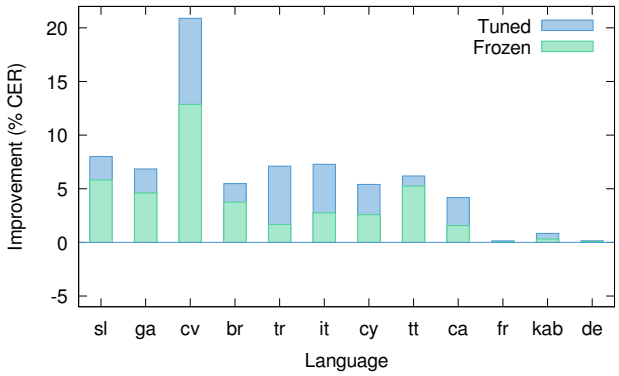


Figure 4. Largest improvement from Transfer Learning relative to the baseline, for each language. Languages are ordered from left → right in ascending order of size of training dataset. These improvements represent the bolded values in Table 2 and Table 3.

7. Model Interpretability: Investigating Embeddings

Our transfer learning seem to indicate that the first three layers of the source model encode good features for language transfer. That is, these features transfer well from English to any of the 12 languages investigated here. However, we can only conjecture at this point as to what those three layers may be encoding. Likewise, we have observed that the fourth (LSTM) layer transfers well with fine-tuning, suggesting that the layer has learned a combination of English-specific and language-generic representations. However, we can only conjecture at this point. Even for the fifth layer, we can only assume that frozen transfer fails due to specificity to English. The follow-up experiments presented in this section aim to provide evidence for or against these intuitions.

For eleven out of twelve languages, frozen transfer of the first three layers shows improved CER over a baseline (French being the exception with a 0.04 decrease in CER). This finding holds for even the larger languages (i.e. Catalan, Kabyle, German). These first three layers are purely feed-forward, and as such they should not be able to encode sequential higher-order information from English (e.g. spelling rules). This may be a clue as to why the first three layers transfer so well, regardless of the target language. Furthermore, we find that when transferring these three layers, CER improvement is essentially identical between frozen or fine-tuned transfer (i.e. less than 0.017 CER difference between average improvement of fine-tuned vs. frozen transfer). This means that information from the target language data is adding nothing extra to the estimation of these layers.

We do find that the LSTM layer can be useful in frozen transfer, but not always. For seven of the eight smallest languages, frozen transfer of the LSTM layer showed improvement over the baseline (i.e. Slovenian, Irish Gaelic, Chuvash, Breton, Turkish, Italian, Tatar, but **not** Welsh). For the four largest languages, we find that frozen transfer up to the LSTM layer either interferes or adds no real improvement over the baseline. This seems to indicate that the LSTM layer encodes some information useful for transfer (i.e. language-agnostic sequential information). Given that the fourth layer is the only recursive layer in Mozilla's variant of DeepSpeech, this layer must encode all sequential information, both language-agnostic and language-specific. An example of language-specific sequential information is spelling rules — they must be learned for every language. An important kind of language-generic sequential information is co-articulation — adjacent phonemes effect each other's acoustic signal based on the physiology of the human vocal tract. All kinds of sequential information are learned in this one layer.

Given our experimental observations and linguistic hypotheses on language-agnostic and language-specific features, we chose to evaluate the usefulness of embeddings at different layers on two new tasks: classification of audio as speech vs. non-speech, and classification of audio as English vs. German. If the features encoded at a certain layer are useful in performing these tasks, then the model has learned something about human speech which is generic (speech vs. noise) or language-specific (English vs. German).

As in all the experiments above, we use the v0.3.0 release of Mozilla's English DeepSpeech as a source model, but instead of appending more 2048x2048 hidden layers, we merely add a logistic regression to the output, trained over three epochs with Cross-Entropy loss (all other hyperparameters are identical to above transfer experiments). This is shown in Figure 5 and with the loss function as defined in Equation 1. This approach is equivalent to using DeepSpeech as a feature extractor, and estimating parameters of a logistic regression over those features.

$$\mathcal{L} = Y \cdot -\log(P) + (1 - Y) \cdot -\log(1 - P) \quad (1)$$

$$P = \sigma((H_n \cdot W_p) + B_p) \quad (2)$$

where

(P) = model prediction
(Y) = ground-truth target label
(H_n) = activations of model at layer n
(W_p) = weights of regression layer
(B_p) = biases of regression layer

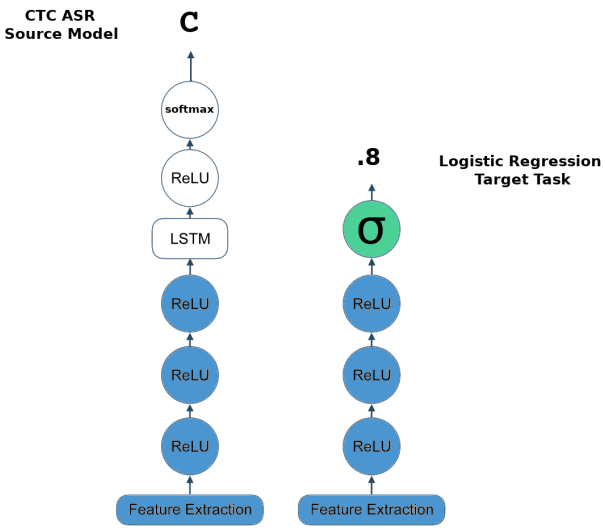


Figure 5. An example of Frozen Transfer from trained English CTC model to logistic regression tasks. In this example, three fully-connected layers are transfered from the original model. Parameters shown in blue are copied and held frozen, parameters shown in green are estimated via Cross-Entropy Loss.

7.1. Speech vs. Noise Classification

For the first task, we choose classification of audio as speech or non-speech. We train and test regression on embeddings of data from 13 Common Voice languages (i.e. the 12 mentioned languages plus English itself). Using multilingual data should reveal the degree to which these features are specific to English, or language-agnostic. We expect any language-agnostic layers to perform this task better than tasks which are purely specific to English. In order to train the regression, we first create a dataset of speech data by combining Common Voice samples for thirteen languages, and for non-speech data we use UrbanSound8K (Salamon et al., 2014), a corpus of environmental city noises.

For the speech data, we make train and test splits of 5005, 442 audio clips respectively (i.e. 385, 34 clips per language). For the non-speech data, we make train and test splits of 5000, 435. We then slice the source model at varying depths, keeping the source layers frozen, and add a feed-forward layer with a single output and logistic activation. This new network layer is then trained with Cross-Entropy loss ($\mathcal{L}$) on the new task. Results are displayed in Table 4.

For the classification task of speech vs. non-speech - we find that the first layer does not encode enough information to perform the task. A regression trained over embeddings from this first layer achieve an accuracy of 51.01% (i.e. essentially a coin-toss) on a held out test set. However, starting

Classification Accuracy					
Number of Layers Copied from English					
1	2	3	4	5	6
51.01	93.68	92.82	**95.30**	94.55	93.53

Table 4. Speech vs. Non-Speech Audio Classification Accuracy (%) of a logistic regression trained on top of frozen layers of English DeepSpeech model. The bolded value shows the model with the highest accuracy. Shading corresponds to relative performance on classification, with darker being more accurate.

at layer [2], all depths contain enough information to classify the samples with $> 92\%$ accuracy. The best accuracy is seen when slicing the model at the LSTM (fourth) layer, likely due to being more effective at capturing long-term patterns, as each sample in the dataset is either all speech or all non-speech. Furthermore, in the first task we find a slight drop in accuracy moving from the fourth layer to the fifth, and likewise we see a drop from the fifth to the sixth. This seems to indicate language-specificity (i.e. highly-tuned to English) at the output layers. Both language-agnostic and language-specific encodings will be useful for speech vs. noise classification, but we are also interested as to which layers encode purely language-agnostic information. To investigate this, we devised a very simple, language identification task.

7.2. Language Identification

The following section presents results from a language identification task trained on the embeddings of the various layers of a trained English model. Specifically, we train a logisitic regression to classify audio as containing English or German language. We create a training dataset by taking an equal number of English and German samples from Common Voice, and then train at varying depths following the same procedure detailed above.

By setting one of the two languages to English, and by extracting embedded features from a trained English DeepSpeech as a source model, we can identify which languages are English-specific. If the layers do not encode English-specific information, then a logistic regression built on top of those features will perform this task poorly. If the layer activates differently for English than for another language, then we can conclude that the layer in question has encoded English-specific information. For the other language, we choose German, given that it is the most closely related to English of the Common Voice languages. As such, distinguishing English and German should be a relatively harder task than distinguishing English from any of the other languages. We make splits of train and test data of 5000,500 audio clips for each language (German and English). The classifier was added on top of the DeepSpeech layers and trained over three epochs with gradient descent from a Cross-Entropy loss function. The results are presented in Table 5.

Classification Accuracy					
Number of Layers Copied from English					
1	2	3	4	5	6
66.51	66.38	52.77	**86.21**	74.97	85.00

Table 5. English vs. German Audio Classification Accuracy (%) of a logistic regression trained on top of frozen layers of English DeepSpeech model. The bolded value shows the model with the highest accuracy. Shading corresponds to relative performance on classification, with darker being more accurate.

We find that the features of the top three layers are better than the bottom three layers on this task. When slicing at the LSTM layer, there's a significant improvement in performance with an 86.21% classification accuracy. This finding means that the model is indeed more language-specific (i.e. English-specific) at the top layers. A particular interesting finding from Table 5 is that feature embeddings from layers [1-3] are essentially language-agnostic. These features are useless for distinguishing English and German. However, we know from our findings from classification of speech and background noise (c.f. Table 4) that the embeddings at layer [3] do encode information about human speech. As such, the features learned at layer [3] are specific to language, but not to any one language. This helps make some sense out of what we find consistent improvements when transfering these features (c.f. layer [3] in Figure 3(a) and Figure 3(b)).

8. Discussion

In this paper we present results from end-to-end ASR transfer learning experiments using a trained English CTC model as a source model and 12 other languages as target languages. We observe a stable, cross-lingual tendency that the first three layers of the trained model are crucial for successful transfer. Our follow-up experiments on the interpretability of the first three layers reveal that they robustly encode the difference between human speech and background noise, but they do not encode information useful for distinguishing languages. In this sense, these first three layers encode language-general information. Furthermore, we find that transfer learning with fine-tuning can exploit and adapt useful bias from a fourth, LSTM layer, which is otherwise too specific to English. Embeddings from this LSTM layer are both useful for speech vs. non-speech classification as well as language discrimination.

For low-resource languages, transfer learning promises quick bootstrapping, avoiding the need of linguistic resources. Furthermore, supporting experiments which investigate the interpretability of end-to-end ASR models are crucial for advancing knowledge on how transfer learning works, and how it can be better used in speech recognition.

Acknowledgments

Removed for review.

References

Amodei, D., Ananthanarayanan, S., Anubhai, R., Bai, J., Battenberg, E., Case, C., Casper, J., Catanzaro, B., Cheng, Q., Chen, G., et al. Deep Speech 2: End-to-end speech recognition in English and Mandarin. In *International Conference on Machine Learning*, pp. 173–182, 2016.

Bahdanau, D., Chorowski, J., Serdyuk, D., Brakel, P., and Bengio, Y. End-to-end attention-based large vocabulary speech recognition. In *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*, pp. 4945–4949. IEEE, 2016.

Bataev, V., Korenevsky, M., Medennikov, I., and Zatvornitskiy, A. Exploring end-to-end techniques for low-resource speech recognition. In *International Conference on Speech and Computer*, pp. 32–41. Springer, 2018.

Battenberg, E., Chen, J., Child, R., Coates, A., Li, Y. G. Y., Liu, H., Satheesh, S., Sriram, A., and Zhu, Z. Exploring neural transducers for end-to-end speech recognition. In *Automatic Speech Recognition and Understanding Workshop (ASRU), 2017 IEEE*, pp. 206–213. IEEE, 2017.

Chan, W., Jaitly, N., Le, Q., and Vinyals, O. Listen, attend and spell: A neural network for large vocabulary conversational speech recognition. In *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*, pp. 4960–4964. IEEE, 2016.

Collobert, R., Puhrsch, C., and Synnaeve, G. Wav2letter: an end-to-end convnet-based speech recognition system. *arXiv preprint arXiv:1609.03193*, 2016.

Cui, J., Kingsbury, B., Ramabhadran, B., Sethy, A., Audhkhasi, K., Cui, X., Kislal, E., Mangu, L., Nussbaum-Thom, M., Picheny, M., et al. Multilingual representations for low resource speech recognition and keyword search. In *Automatic Speech Recognition and Understanding (ASRU), 2015 IEEE Workshop on*, pp. 259–266. IEEE, 2015.

Cui, J., Kingsbury, B., Ramabhadran, B., Saon, G., Sercu, T., Audhkhasi, K., Sethy, A., Nussbaum-Thom, M., and Rosenberg, A. Knowledge distillation across ensembles of multilingual models for low-resource languages. In *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*, pp. 4825–4829. IEEE, 2017.

Deng, L. and Li, X. Machine learning paradigms for speech recognition: An overview. *IEEE Transactions on Audio, Speech, and Language Processing*, 21(5):1060–1089, 2013.

Gales, M. J., Knill, K. M., Ragni, A., and Rath, S. P. Speech recognition and keyword spotting for low-resource languages: Babel project research at cued. In *Spoken Language Technologies for Under-Resourced Languages*, 2014.

Ghoshal, A., Swietojanski, P., and Renals, S. Multilingual training of deep neural networks. In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pp. 7319–7323. IEEE, 2013.

Glorot, X. and Bengio, Y. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pp. 249–256, 2010.

Graves, A., Fernandez, S., Gomez, F., and Schmidhuber, J. Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd International Conference on Machine Learning*, 2006.

Grézl, F., Karafiát, M., and Vesely, K. Adaptation of multilingual stacked bottle-neck neural network structure for new language. In *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, pp. 7654–7658. IEEE, 2014.

Hannun, A. Y., Case, C., Casper, J., Catanzaro, B., Diamos, G., Elsen, E., Prenger, R., Satheesh, S., Sengupta, S., Coates, A., and Ng, A. Y. Deep Speech: Scaling up end-to-end speech recognition. *CoRR*, abs/1412.5567, 2014a. URL http://arxiv.org/abs/1412.5567.

Hannun, A. Y., Maas, A. L., Jurafsky, D., and Ng, A. Y. First-pass large vocabulary continuous speech recognition using bi-directional recurrent DNNs. *arXiv preprint arXiv:1408.2873*, 2014b.

Heafield, K. KenLM: faster and smaller language model queries. In *Proceedings of the EMNLP 2011 Sixth Workshop on Statistical Machine Translation*, pp. 187–197, Edinburgh, Scotland, United Kingdom, July 2011. URL https://kheafield.com/papers/avenue/kenlm.pdf.

Knill, K. M., Gales, M. J., Rath, S. P., Woodland, P. C., Zhang, C., and Zhang, S.-X. Investigation of multilingual deep neural networks for spoken term detection. In *Automatic Speech Recognition and Understanding (ASRU), 2013 IEEE Workshop on*, pp. 138–143. IEEE, 2013.

Oquab, M., Bottou, L., Laptev, I., and Sivic, J. Learning and transferring mid-level image representations using convolutional neural networks. In *Proceedings of the IEEE*

conference on computer vision and pattern recognition, pp. 1717–1724, 2014.

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

Pratap, V., Hannun, A., Xu, Q., Cai, J., Kahn, J., Synnaeve, G., Liptchinsky, V., and Collobert, R. wav2letter++: The fastest open-source speech recognition system. *arXiv preprint arXiv:1812.07625*, 2018.

Rao, K., Sak, H., and Prabhavalkar, R. Exploring architectures, data and units for streaming end-to-end speech recognition with rnn-transducer. In *Automatic Speech Recognition and Understanding Workshop (ASRU), 2017 IEEE*, pp. 193–199. IEEE, 2017.

Rosenberg, A., Audhkhasi, K., Sethy, A., Ramabhadran, B., and Picheny, M. End-to-end speech recognition and keyword search on low-resource languages. In *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*, pp. 5280–5284. IEEE, 2017.

Salamon, J., Jacoby, C., and Bello, J. P. A dataset and taxonomy for urban sound research. In *Proceedings of the 22nd ACM international conference on Multimedia*, pp. 1041–1044. ACM, 2014.

Toshniwal, S., Sainath, T. N., Weiss, R. J., Li, B., Moreno, P., Weinstein, E., and Rao, K. Multilingual speech recognition with a single end-to-end model. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4904–4908. IEEE, 2018.

Vu, N. T., Imseng, D., Povey, D., Motlicek, P., Schultz, T., and Bourlard, H. Multilingual deep neural network based acoustic modeling for rapid language adaptation. In *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, pp. 7639–7643. IEEE, 2014.

Yosinski, J., Clune, J., Bengio, Y., and Lipson, H. How transferable are features in deep neural networks? In *Advances in neural information processing systems*, pp. 3320–3328, 2014.