Augmenting Sparse Corpora
for Enhanced Sign Language Recognition and Generation

Heike Brock¹, Juliette Rengot², Kazuhiro Nakadai¹

¹Honda Research Institute Japan Co., Ltd., ²Ecole Nationale des Ponts et Chaussées
8-1 Honcho, Wako-shi, Saitama 351-0188, Japan, 6 et 8 avenue Blaise Pascal, 77455 Champs-sur-Marne, France
{h.brock, j.rengot, nakadai} @jp.honda-ri.com

Abstract

The collection of signed utterances for recognition and generation of Sign Language (SL) is a costly and labor-intensive task. As a result, SL corpora are usually considerably smaller than their spoken language or image data counterparts. This is problematic, since the accuracy and applicability of a neural network depends largely on the quality and amount of its underlying training data. Common data augmentation strategies to increase the number of available training data are usually not applicable to the spatially and temporally constrained motion sequences of a SL corpus. In this paper, we therefore discuss possible data manipulation methods on the base of a collection of motion-captured SL sentence expressions. Evaluation of differently trained network architectures shows a significant reduction of overfitting by inclusion of the augmented data. Simultaneously, the accuracy of both sign recognition and generation was improved, indicating that the proposed data augmentation methods are beneficial for constrained and sparse data sets.

Keywords: sign language, machine learning corpus, data augmentation, motion capture data, avatar animation

1. Introduction

In contrast to resources on spoken language, signed machine learning data can hardly be obtained from scratch. Instead, signed conversations need to be recorded first using movement sensing devices such as video or depth cameras, and their content be annotated with the help of experienced or fluent signers. For this reason, corpora for Sign Language (SL) research are seldom as extensive as their spoken counterparts. Corpora for specific tasks such as sign recognition or avatar animation require even more specialized capture environments and settings, consuming a high amount of both financial and human resources. This leads to small and sparse SL corpora that are seldom able to depict all linguistic features under a natural vocabulary scope: common corpora either focus on specific aspects (e.g., facial expressions for sign avatar synthesis (Gibet et al., 2016); Ebling and Haenerfauth, 2015) or specific language domains (e.g., weather reports (Koller et al., 2015; Umeda et al., 2016)) and usually contain a very small subset of lexical items or separated signs only (Pigou et al., 2014; Ong and Ranganath, 2005).

Especially for the application of modern Machine Translation (MT) methods that base on pure data-driven artificial networks, common SL corpora appear insufficient to learn meaningful data representations. To develop sign recognition and synthesis systems that are robust towards outliers and universally applicable without prior domain knowledge, future data sets need to considerably increase their amount of training data. Here, it appears reasonable to modify existing data in such a way that new data is created that contains the significant characteristics of the signed expressions and their variant lexical items, while simultaneously maintaining their linguistic shape and semantic meaning. However, this requirement excludes the use of common simple data augmentation strategies such as mirroring and flipping or cropping, and new strategies for data augmentation have to be found to develop highly accurate communication technologies in the future.

A number of data augmentation strategies that could be used to increase the number of available training data without data loss are given in this work. Based on a corpus of motion captured sentence expressions in Japanese Sign Language (JSL), we evaluate the benefit of the presented data augmentation methods on the learning of a deep recurrent neural network architecture. Results indicate that problems characteristic for small and sparse data sets can be reduced, suggesting the benefit of the presented strategies for future SL translation interfaces.

2. Machine Learning Corpus Augmentation

Data augmentation is a common machine learning strategy to improve accuracy of classifiers and predictors. Its deployment bases on the presumption that machine learning models do not generalize well when they are trained on data sets that do not contain sufficient variation within the data. Previous research utilizing image training data has shown that data augmentation can act as a regularizer and prevent overfitting of the neural networks (Simard et al., 2003; Ciresan et al., 2010): the more data a machine learning model has access to, the more useful information can be extracted from the original data set and hence the more effective it can be. In line with these results, the most accurate image classification or segmentation networks presented within the last years all utilize techniques to artificially synthesize new samples from existing ones, such as the Imagenet (Krizhevsky et al., 2012) or its succeeding, even deeper network variations (Szegedy et al., 2015; He et al., 2016). Consequently, data augmentation should be particularly useful when the number of available training data is small and new training data cannot be acquired easily, as it is common for sign language corpora. This holds especially true when deep neural networks – that rely largely on the number of available sample data for network learning – are applied.
2.1. Data Augmentation and SL

Images are the most commonly used machine learning data type for manipulation and artificial sample synthesis. This is because the content of an image can already be sufficiently altered by simple modifications such as adding minor perturbations (e.g., noise, blur or contrast) or minor transformations (e.g., mirroring, rotation, shearing or zooming) (Asperti and Mastronardo, 2017). A recent, more advanced possibility to change data content is the modification of image style with the help of Generative Adversarial Networks (Perez and Wang, 2017). Common to all those manipulations however is that they cannot be applied to time-series data without altering the semantic content of the underlying data: already simple changes like mirroring can render a signed expression meaningless from a linguistic point of view.

To date, few studies discuss the effect of data warping on machine learning corpora of time-serial data such as handwriting strokes (Wong et al., 2016) or data acquired by an accelerometer (Munoz-Organero and Ruiz-Blazquez, 2017). However, these data streams are of relatively small dimensionality as compared to full body signing movements, and potential data augmentation strategies have to be investigated separately to validate their benefit with a respective SL corpus.

3. Experimentation

In this work, we evaluate the benefit of multiple motion sequence augmentation methods on the performance of a Recurrent Neural Network (RNN) for the recognition of signed sentence content. This RNN constitutes a simple implementation of the sequence to sequence model (Seq2Seq) (Sutskever et al., 2014) for English-French translation with a set of Long-Short Term Memory (LSTM) cells and is similar to a simple encoder-decoder pipeline (Cho et al., 2014). One of the main advantages of this network architecture is that no prior data segmentation is required and the signing sequences can be used as is as network input. The network was trained using a specialized corpus of signed sentence expressions in JSL captured with an optical motion capture system. Here, the idea was to utilize a corpus of data streams that are highly detailed and accurate, so that they could not only be used for the recognition of signed content but also for the generation of virtual signing avatars in the future.

3.1. Corpus and Network Details

379 sentence structures were signed in 2 to 3 different speeds by one fluent signer (Child of Deaf Adults) and simultaneously recorded utilizing a dense Vicon optical motion capture system of 42 cameras (Figure 1). Position and rotations of all relevant body and finger joints as well as facial motion capture were acquired to build a dense set of signing movements able to represent all relevant aspects of a signed expression.

In total, 740 sentences with a vocabulary of 195 words and their corresponding gloss annotations were recorded. Within this corpus, 69 groups of 4 to 6 sentences with similar vocabulary were composed to ensure the repetitive occurrence of the chosen word content. Every group of sen-

Figure 1: A set of 48 optical cameras was used to record the signed motion sentences. Marker were densely placed on body, finger and face of the sign speaker to obtain a dense collection of sign motion data.

ences furthermore contained basic grammatical structures of JSL including non-manual signs and context information (Brock and Nakadai, 2018). In concrete, these were directional and syntactic information such as affirmation, negation and interrogation, adjective inflection (annotated as (CP2) for the comparative and (CP3) for the superlative), logical content separation, as well as compound verbs built from space and size classifiers (annotated e.g. as CL(P) for a location indication or CL(2ppl) for a two-person indication).

One sentence pattern of each group was randomly chosen for evaluation (154 sentences) and the rest for training (586 sentences). For every data augmentation strategy tested, the number of available training data was then increased by synthesizing new data streams from the original motion capture training data. Next, a new network was learned and the network performance for recognition of the test sentences determined. According to the specification of the Seq2Seq model, the model was trained taking as input the JSL motion sentences and as output one-hot encoded vectors of the respective sentence expression in gloss annotation. To recognize words within the sentence, an additional layer then performed a softmax function on the separate encoding parts (representing classes) of each output sequence and chose the word with the higher probability each time. Lastly, network outputs were post-processed by removing repeated output words.

Throughout the experiment, network parameter such as the number of hidden layers or the size of the LSTM cells were left unchanged to enable the best comparability within the various training results. These parameters were: 1 hidden layer with 256 LSTM cells and six different buckets of length 400, 500, 600, 700, 800 and 900 frames to account for varying sentence length. The recognition network was trained for 2000 epochs for every comparison and the loss function employed was the cross entropy. For the optimization of the LSTM cell parameters, we used the Adagrad optimiser.

It should be noted that motion capture data are seldom used for sign recognition in practice due to the high cost, overhead and effort of data acquisition. Instead, motion captured data sets are more relevant for sign synthesis and avatar animation tasks. For the comparison in this work, we still chose to implement and evaluate a recognition network instead of a generation network, mostly due to the reason that the output of recognition networks are easier to
evaluate in terms of accuracy and training loss. Since the Seq2Seq model can be used to generate and at the same time understand sequences, the identical model could also be used for synthesis of a motion sequence from gloss annotation input in the future.

3.2. Applied Augmentation

Three data augmentation strategies were chosen to increase the number of available training data in this work: noise, reversing sequence streams based on anthropometric specifications and sequence warping according to length of different sequences of similar content. Every approach enabled us to double the respective underlying corpus size, leading to differently sized training corpora for the learning of five independent sign recognition networks. These were a simple baseline network trained on only the motion captured data, one network using additional noised corpus data, one network using additional anthropometric reversed sequence streams, one network including warped variations of the original and reversed streams and one network combining all of the augmentation strategies.

3.2.1. Noise

The simplest way of creating new samples must be adding noise to the existing ones. We assessed the effect of two noise types here: basic Gaussian noise and Perlin noise known to better suit the characteristics of human motion.

**Gaussian Noise** We considered a random variable following a Gaussian distribution $X \sim \mathcal{N}(0, 0.02)$. At each time step, new coordinates were obtained by adding this random variable to the former data:

$$
\begin{align*}
x' &= x + \mathcal{N}(0, 0.02) \\
y' &= y + \mathcal{N}(0, 0.02) \\
z' &= z + \mathcal{N}(0, 0.02) \\
\alpha' &= \alpha + \mathcal{N}(0, 0.02) \\
\beta' &= \beta + \mathcal{N}(0, 0.02) \\
\gamma' &= \gamma + \mathcal{N}(0, 0.02)
\end{align*}
$$

with $x$ ($x'$) representing the former (new) position along the lateral axis, $y$ ($y'$) the former (new) roll position along the dorsal axis, $z$ ($z'$) the former (new) position along the vertical axis, $\alpha$ ($\alpha'$) the former (new) pitch angle, $\beta$ ($\beta'$) the former (new) roll angle and $\gamma$ ($\gamma'$) the former (new) yaw angle. We added this noise to only five significant joints, namely the head, both elbows and both wrists (Guo et al., 2016).

**Perlin Noise** Different than for the Gaussian noise, noise characteristics were chosen in dependence on the considered joint: the more distal the joint, the lower the noise amplitude and the higher the noise frequency. We used amplitude and frequency values demonstrated to be meaningful for use with SL motion capture data (Medonald et al., 2016) to add Perlin noise to hips, waist, upper spine, neck, shoulders, elbows and wrists.

3.2.2. Anthropometric Reversing

In JSL, some words, like the name “Yamamoto” or “book”, need both hands to be expressed whereas other ones, like the name “Sato” or “mother”, are one-handed. For the latter, both the right and the left hand can be used. JSL signers usually have a “dominant hand”: right-handed signers use more often their right hand, and vice-versa. However, it is possible to switch hands if it is more convenient in a particular situation (e.g. when driving or holding something) (Nakamura, 2006), or to emphasize spatial content information within a sentence expression. Therefore, it is critical that our corpus contains both right-handed and left-handed examples of each words. Woefully, the original dataset was strongly imbalanced because data was recorded from only one right-handed signer. That is why additional data representing the same motion as executed by the opposite body half should improve robustness of a training corpus.

To synthesize such movements, we mirrored the data streams of all upper body joints along the vertical axis by swapping right and left values. Moreover, we changed the sign of pitch translations as well as roll and yaw angles. To give a more natural feeling, roll and yaw angles of the head also took their opposite values. Using a virtual avatar displaying the motion capture data (Figure 2), we verified the naturalness and accuracy of the anthropometric reversed sentence expression.

![Figure 2: A screen-shot of the original motion (Right) and of the reversed motion (Left), executed by our avatar.](image-url)

3.2.3. Dynamic Time Warping

For each corpus sentence, at least two repetitions of different signing pace were available, providing the possibility to generate new time variations with Dynamic Time Warping (DTW). First and foremost, it is interesting to note that all recorded data contained static phases at the beginning and at the end of no content information. Therefore, all sentences were first truncated by a simple threshold metric that kept insignificant sequence parts small. The shortest sequence among all captures of identical sentence content was then aligned to the length of the longest sequence, and vice versa.

Let $S_1$ and $S_2$ be two sequences. In order to obtain a sequence of the same length as $S_2$, $S_1$ is stretched ($S_1 < S_2$) or squeezed ($S_1 > S_2$) by uniform scaling. Scaling is not necessary for sequence alignment, but considered beneficial to increase invariance to large variations in global scale (Fu et al., 2008). The formula to scale a time series $Q = (Q_1, Q_2, \ldots, Q_m)$ of length $m$ to produce a new time series $Q_P = (QP_1, QP_2, \ldots, QP_p)$ of length $p$ is defined by:

$$Q_P = \left(\frac{Q_1}{Q_{max}}, \frac{Q_2}{Q_{max}}, \ldots, \frac{Q_m}{Q_{max}}\right)$$

where $Q_{max}$ is the maximum value in the original sequence $Q$. This operation is linear and maintains the relative order of elements in the sequence.
Next, DTW was used to adjust the sequence to small local misalignments. We made use of the implementation of fastDTW (Salvador and Chan, 2007) (Figure 3). For two sequences $S_1 = (x_1, \ldots, x_N)$ and $S_2 = (y_1, \ldots, y_N)$, we obtained the optimal warping path $P$ of length $L$: $P = ((a_1, b_1), \ldots, (a_L, b_L))$. The best alignment (i.e. with minimal total cost) assigned $x_a$ to $y_b$ with $\ell \in [1, L]$. The warped sequences showed some defects and appeared less natural than the original data when transferred onto the virtual avatar. On the one hand, we could observe stutters in the motion data generated by expansion of the shorter sequences. On the other hand, transformation of the longer sequences gave rise to hasty motion. As a consequence, warped sentence expressions might be difficult to understand when displayed using the virtual avatar. In order to make the motion more natural, we added a smoothing function to the warped sequence outputs. Clearly, this solution has drawbacks: gestures lost in accuracy and sharpness. Applying a moving average over 5 or 11 data points seemed to be a good middle-of-the-road solution. We applied this method to both the original and anthropometric reversed data. Thus, we create 2960 additional files for network training.

4. Results

In this section, we present the results obtained with the five different corpora mentioned hereinafore. For comparison, the evaluation set was always composed of the same.

$$QP_j = Q_j \frac{m}{p} \text{ where } 1 \leq j \leq p$$
4.2 Noise

Gaussian noise  The inclusion of training data augmented with Gaussian noise seemed to reduce the overfitting problem as the training loss decreased constantly (Figure 7). However, the test loss still remained quite far from training loss. As compared to Figure 4, test accuracy was brought closer to the training accuracy and now varied between 15% and 35% over the different buckets. Further analysis using the confusion matrix and decoded sentence recognition did not show any improvements and the output could still not be considered relevant or reliable. Semantically (and also morphologically) different lexical items such as for instance given in the sentences 26C_02 and 69A_03 (as listed in Figure 6) could not be distinguished. Moreover, again all pointing gestures were abnormally often detected and predicted labels tended to be selected from a restricted set of vocabulary. We assumed that Gaussian noise-augmented data may either not be different enough from the original data – or too different from the original data, respectively – to significantly improve the learning process. Another possible explanation of those results could be that the five considered joints were not significant enough: in a signed expression, finger joints, facial movements and eye-gaze often carry further meaningful information. We therefore expanded the previous augmentation to all joints, but did not obtain considerably different results. As a conclusion, we did not keep Gaussian noise files in the following.

Figure 6: Three examples of the decoded sentences obtained by evaluating the model trained on non-augmented data.

Figure 7: Loss and accuracy evolution during network training when using the original MoCap data and additional noise-augmented data, here shown for Bucket 3 of length 600. Gaussian noises come to the red, green and cyan curves. Perlin noises produce the black, orange and purple curves.

Perlin noise  Replacing Gaussian noise with Perlin noise decreased the test loss, improved test accuracy and seemed to settle the overfitting problem well: for all the six buckets, the test loss was continuously and smoothly decreasing, and the test accuracy remained close to the training one (Figure 7). We obtained test accuracies between 20% and 55% within the respective buckets.

While studying confusion matrix plots, we noted that signs were still confused to the most frequent lexical items, and especially the referential pointing “pt3”. Considering the procedure of the performed data augmentation, these misclassifications could be explained well with the underlying data structure: as all corpus sentences were uniformly augmented, also their internal word count was equally increased. This means that the absolute count of frequent lexical items expanded as compared to the absolute count of infrequent lexical items. In conclusion, it appears reasonable to apply data augmentation only to those parts of the training data that contain none or few of the most frequent words, or to include and augment single words of low frequency, to better balance the general word distribution in the training corpus.

4.3 Anthropomorphic Reversing

Now, let us have a look at the corpus augmented with the anthropomorphic reversing strategy. Similar as for using Perlin noise, the overfitting problem seemed to be solved well, while test accuracies ranged between 20% and 50% within the respective buckets. Nevertheless, no significant positive changes could be registered in either the confusion matrix nor the decoded sentence recognition output and results shall hence not be further discussed here.

4.4 Dynamic Time Warping

Inclusion of the warped sentences could not further improve the data and gave similar results than the previous (smaller) corpora with respect to network training parameters. Test accuracies varied between 20% and 50% depending on buckets, while decoded sentences were still as irrelevant as before. Supported by the visual data inspection performed with the virtual avatar, we suppose that DTW did not necessarily preserve all meaningful properties of the signing dynamics. Human gestures follow certain laws related to motor control. DTW may introduce data which
violate these kinematics, leading to unnatural signing data streams that might influence recognition.

4.5. Final set

Finally, we trained our network with a corpus including the original data and synthesized files of all data augmentation strategies. In concrete, these were Perlin noise, anthropometric reversing and DTW applied to both the original and reversed sequences. Resulting test accuracies ranged between 20% and 55% depending on buckets, and good overall network training could be achieved (Figure 10). Differences between the new confusion matrix (Figure 11) and the original one (Figure 5) are obvious, but oppose our first expectations: instead of higher accurate recognition, signs were even stronger confused to few certain lexical items. Surprisingly, predominantly detected words were not the frequent pointing gestures “pt1”, “pt2”, and “pt3”, but words that seem to be randomly chosen from the available corpus vocabulary such as “recommend” and “beautiful”. Moreover, discriminant lexical items like “Hawaii”, whose presence within a sentence pattern could be correctly identified within the test data beforehand, could no longer be retrieved from the test data (Figure 12).

5. Discussion

All of the previous data augmentation strategies improve the general model trainability and extend the amount of corpus data for better use in deep recurrent neural networks, as summed up in Table 1. In particular, the main target of reducing network overfitting could be addressed well by adding a larger number of unknown training data. This is promising as it suggests that further extended data corpora could also improve the robustness and accuracy of future translation systems.

In our specific context, the addition of Gaussian noise may be the less efficient method because general test accuracy does not surpass 35%. Highest test accuracies that could be reached were of approximately 55%. Although this number appears small, results are encouraging for the given context of unsegmented and consecutive SL data streams: to date, continuous sentence expressions have rarely been utilized to learn machine translation networks. Best results were reported for continuous hand-shape recognition using a convolutional neural network trained on a considerably larger set of video data, but could also not surpass accuracies of approximately 60% (Koller et al., 2016).

Despite the improved network trainability, data augmentation could not support reliable sentence recognition. We assume this to be mainly due to the unbalanced characteristic of the training corpus: as discussed, SLs contain specific lexical items that are used repetitively within most sentence patterns. Hence, the network architectures quickly fit to these words and preferably chose the lexical items in their recognition output. Specialized data augmentation strategies such as the Synthetic Minority Over-Sampling...


