

Symbol Grounding Association in Multimodal Sequences with Missing Elements

Federico Raue

FEDERICO.RAUE@DFKI.DE

*Computer Science Department, TU Kaiserslautern,
Gottlieb-Daimler Str. 1, 67663 Kaiserslautern, Germany*

Andreas Dengel

ANDREAS.DENGEL@DFKI.DE

*Smart Data and Knowledge Services,
German Research Center for Artificial Intelligence (DFKI),
Trippstadter Str. 122, 67663 Kaiserslautern, Germany*

Thomas M. Breuel

TMB@CS.UNI-KL.DE

*Computer Science Department, TU Kaiserslautern,
Gottlieb-Daimler Str, 1, 67663 Kaiserslautern, Germany*

Marcus Liwicki

LIWICKI@CS.UNI-KL.DE

*MindGarage, TU Kaiserslautern,
Davenportpl. 11,6 7663 Kaiserslautern, Germany*

Abstract

In this paper, we extend a symbolic association framework for being able to handle missing elements in multimodal sequences. The general scope of the work is the symbolic associations of object-word mappings as it happens in language development in infants. In other words, two different representations of the same abstract concepts can associate in both directions. This scenario has been long interested in Artificial Intelligence, Psychology, and Neuroscience. In this work, we extend a recent approach for multimodal sequences (visual and audio) to also cope with missing elements in one or both modalities. Our method uses two parallel Long Short-Term Memories (LSTMs) with a learning rule based on EM-algorithm. It aligns both LSTM outputs via Dynamic Time Warping (DTW). We propose to include an extra step for the combination with the max operation for exploiting the common elements between both sequences. The motivation behind is that the combination acts as a condition selector for choosing the best representation from both LSTMs. We evaluated the proposed extension in the following scenarios: missing elements in one modality (visual or audio) and missing elements in both modalities (visual and sound). The performance of our extension reaches better results than the original model and similar results to individual LSTM trained in each modality.

1. Introduction

A striking feature of the human brain is to link abstract concepts and sensory input signals, such as visual and audio. As a result of this multimodal association, an abstract concept that has several representations (i.e., visual and audio) maps from one modality to another modality, and vice versa. For example, the abstract concept “ball” from the sentence

“John plays with a ball” can be associated with several instances of different spherical shapes (visual input) and sound waves (audio input). Several fields, such as Neuroscience, Psychology, and Artificial Intelligence, are interested in determining all factors that are involved in binding semantic concepts and the physical world. This scenario is known as *Symbol Grounding Problem* (Harnad, 1990) and is still an open problem (Steels, 2008).

With this in mind, infants start learning the binding between abstract concepts and the real world in a multimodal scenario. Gershkoff-Stowe and Smith (2004) found the initial set of words in infants is mainly nouns, such as dad, mom, cat, and dog. In contrast, the lack of stimulus can limit the language development (Andersen, Dunlea, & Kekelis, 1993; Spencer, 2000), i.e., deafness, blindness. Asano et al. (2015) found two different patterns in the brain activity of infants depending on the semantic correctness between a visual and an audio stimulus. In simpler terms, the brain activity is pattern ‘A’ if the visual and audio signals represent the same semantic concept. Otherwise, the pattern is ‘B’.

Related work has been proposed, in which one model combines the *Symbol Grounding Problem* and the association learning. Yu and Ballard (2004) explored a framework that learns the association between objects and their spoken names in day-to-day tasks. Nakamura et al. (2011) introduced a multimodal categorization applied to robotics. Their framework exploited the relation of concepts in different modalities (visual, audio and haptic) using a Multimodal latent Dirichlet allocation.

Previous approaches have focused on associating isolated elements. This work proposes another approach, where multimodal sequences are the input of the association task. Text line classification in OCR (Breuel, Ul-Hasan, Al-Azawi, & Shafait, 2013) and image segmentations (Byeon, Breuel, Raue, & Liwicki, 2015) are two successful examples that handle images as sequences. In this paper, the association task is between two multimodal sequences that present semantic concepts in one or two sequences. Moreover, we are interested in semantic concept sequences represented by information from two or more modalities with the constraint that some elements are not part of all modalities; for instance, an element may be unique in one modality. For instance, one modality sequence (text lines of digits) is represented by ‘2 4 6’, and the other modality (spoken words) is represented by ‘two five six’; where ‘4’ is unique to the text modality, and ‘five’ is unique to the spoken modality. The missing elements can be interpreted as a scenario where some words are not linked directly to a visual representation, such as verbs are connected to actions instead of a specific object.

In this work, we investigate the multimodal association of weakly labeled sequences based on the alignment between two *latent spaces*. In more detail, some elements of the sequence in one modality are not present in the other modality. Note that presented model is an extension of the work introduced by Raue et al. (2015) where both modalities represent the same semantic sequence (no missing elements). Similarly to their model, two Long Short-Term Memories (LSTMs) are the main components of the presented model where the output vectors are aligned on the time axis using Dynamic Time Warping (DTW) (Berndt & Clifford, 1994). Our contributions in this paper are the following

- We propose a novel model for the cognitive multimodal association task (without claiming that being cognitively plausible). Moreover, our model handles multimodal sequences where the semantic concepts can be in one or both modalities. Also, a *max*

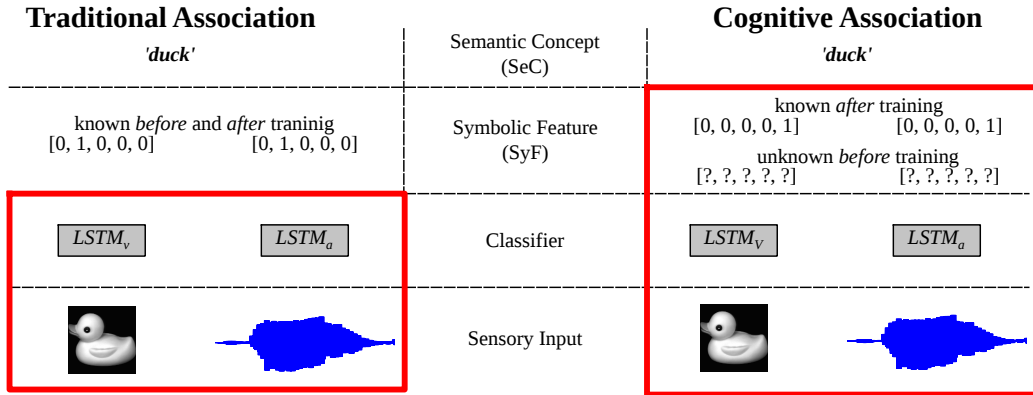


Figure 1: Comparison of components between the traditional setup and our setup for associating two multimodal signals. Note that our task has an extra learnable component (relation between semantic concepts and their representations), whereas the traditional scenario is already predefined (red box). Moreover, the final goal is to agree on the same coding scheme for each modality.

operation in the time-axis is novel in the architecture, and the motivation is to exploit the cross-modality of the shared semantic concepts.

- We evaluate the presented model in two scenarios. In the first scenario, the missing semantic concepts can be in any modality. In the second scenario, the semantic concepts are missing only in one modality. For example, the visual sequence ‘1 2 3 4 5 6’ and the audio sequence ‘two four’. The visual and audio modalities share the semantic concepts *two and four*. In contrast, the semantic concepts *one, three, five, and six* are presented only in the visual sequence. In both cases, our model performances better than the model proposed by Raue et al. (2015).

This paper is organized as follows. We shortly describe Long Short-Term Memory networks that are a recurrent neural network in Section 2. Section 3 explains how LSTM can be trained based on weakly labeled sequences. Section 4 describes the original model for the object-word association. Section 5 presents the novel model for handling missing elements. Section 6 shows a new dataset of multimodal sequences with missing elements. In Section 7, we compare the performance of the proposed extension, the original model, and a single LSTM network trained on one modality in the traditional setup (predefined coding scheme).

1.1 Symbol Grounding and Association Learning in Neural Networks

In this work, we are interested in unifying the Symbol Grounding Problem and the association Learning in a neural network architecture. For example, de Penning et al. (2011) proposed a model that combines neural networks with temporal rules. Consider the cross-modal scenario, in which an isolated object and a spoken word represent the same Semantic

Concept. One option for learning the association is to have two neural networks: one network for the visual channel and another for the audio channel. It is safe to assume that the *raw output vectors* of neural networks carry attributes or discriminant information of input samples. Thus, the *raw output vectors* can be considered as a *numerical, symbolic feature*. With this in mind, we can define the association task inspired by the Symbol Grounding Problem as follows. The *symbolic features* (raw output vectors) and semantic vectors are not binding initially. This decision is usually taken before training and external to the network. In contrast, the presented task requires that the model learn the binding. Figure 1 shows a comparison between the *traditional association task* and the presented task (inspired by the Symbol Grounding Problem). It can be observed that the *traditional association* has already defined two elements (red line - traditional association): a) the association is between two samples and b) the binding between semantic concepts and symbolic features. Those elements are defined semantic concepts and the network output using the same one hot scheme. On the other hand, the training algorithm includes the previous two elements (red line - cognitive association). In this case, the network training incorporates two tasks. First, each neural network learns the binding between semantic concepts and symbolic features. Second, both neural networks are learning to agree to the same binding between symbolic features and semantic concepts.

1.2 Multimodal Tasks in Machine Learning

Machine Learning has been applied successfully to several scenarios where the architecture exploits the multimodal relation between input samples. In the following, we want to indicate the differences between previous multimodal tasks and our work.

Multimodal Feature Fusion The task is to combine features of different modalities for creating a better feature. In this manner, the generated feature exploits the best qualities of each modality. Deep Boltzmann Machines have been used to combine different modalities in unsupervised environments (Srivastava & Salakhutdinov, 2012; Sohn, Shang, & Lee, 2014). Iqbal and Silver (2016) proposed an architecture that combined three modalities, where the previous approaches combine only two modalities.

Image Captioning The task is to generate a textual description given images as input. In other words, this task can be seen as a machine translation from images to captions. One of the approaches to solving image captioning is a combination of Convolutional Neural Networks (CNN) and LSTM, where CNN encodes images and LSTM generates the textual descriptions (Vinyals, Toshev, Bengio, & Erhan, 2014; Karpathy, Joulin, & Fei-Fei, 2014).

Weakly Labeled Association This task is related to learn the association between two parallel sequences that represent the same order of semantic concepts. One essential requirement is that both sequences are weakly labeled. The two scenarios (image captioning and feature fusion) require that words must be already segmented (more details in Section 3). Raue et al. (2015) proposed a model that has two parallel LSTM networks that exploit the multimodal latent space produced by LSTM networks. Both LSTMs align with each other for training. The goal is that one LSTM of one modality learns with the latent space produced by the other modality (more details in Section 5).

2. Long Short-Term Memory (LSTM)

Long Short-Term Memory (LSTM) is a recurrent neural network that handles the vanishing gradient problem in long sequences (Hochreiter & Schmidhuber, 1997; Hochreiter, 1998). LSTM solves the vanishing gradient problem based on a set of gates that manages the flow information in three aspects: input, reset, and output. More formally, this architecture is defined by

$$\mathbf{i}_t = \sigma(\mathbf{W}_{xi}\mathbf{x}_t + \mathbf{W}_{hi}\mathbf{h}_{t-1} + \mathbf{b}_i) \quad (1)$$

$$\mathbf{f}_t = \sigma(\mathbf{W}_{xf}\mathbf{x}_t + \mathbf{W}_{hf}\mathbf{h}_{t-1} + \mathbf{b}_f) \quad (2)$$

$$\mathbf{o}_t = \sigma(\mathbf{W}_{xo}\mathbf{x}_t + \mathbf{W}_{ho}\mathbf{h}_{t-1} + \mathbf{b}_o) \quad (3)$$

$$\mathbf{g}_t = \tanh(\mathbf{W}_{xc}\mathbf{x}_t + \mathbf{W}_{hc}\mathbf{h}_{t-1} + \mathbf{b}_c) \quad (4)$$

$$\mathbf{c}_t = \mathbf{f}_t\mathbf{c}_{t-1} + \mathbf{i}_t\mathbf{g}_t \quad (5)$$

$$\mathbf{h}_t = \mathbf{o}_t \tanh(\mathbf{c}_t) \quad (6)$$

$$\mathbf{z}_t = \text{softmax}(\mathbf{W}_{hz}\mathbf{h}_t + \mathbf{b}_z) \quad (7)$$

where $\mathbf{x}_t \in \mathbb{R}^n$ is the input vector at time step t , $\mathbf{z}_t \in \mathbb{R}^c$ is the output vector at time step t , and \mathbf{W}_* and \mathbf{b}_* are the weight matrices and bias, respectively.

LSTM is trained following similar steps of any other gradient-based method. Moreover, one of the most common approaches for training has been proposed by Werbos (1990). He defined an algorithm, called *Backpropagation Through Time (BPTT)*, which updates the network parameters from the last time step to the first time step. In other words, the algorithm describes the loss function as follows

$$E(t) = \mathbf{y}_t - \mathbf{z}_t, \quad t = T, \dots, 1 \quad (8)$$

$$\frac{\partial}{\partial \mathbf{W}_*} E = \sum_{t=1}^T \frac{\partial}{\partial \mathbf{W}_*} E(t) \quad (9)$$

where $\mathbf{y}_t \in \mathbb{R}^c$ is the target vector, θ is the network parameters, $\partial E(t)/\partial \mathbf{W}_*$ is the derivative of the loss function w.r.t. to the network parameters.

So far, we have explained only one LSTM. Additionally, another approach combines two LSTMs in the following manner. One LSTM runs from 1 to T, and another LSTM runs from T to 1. This setup is called *Bidirectional LSTM*, and the motivation is to exploit the surrounded context of a specific position (i.e., before and after).

3. Weak Label Training for LSTM

LSTM has been successfully applied to several scenarios, such as image captioning (Karpathy et al., 2014), machine translation (Sutskever, Vinyals, & Le, 2014), and texture classification (Byeon et al., 2015). All previous examples require segmented data. For example, words in sentences show a type of segmentation based on spaces (image captioning and

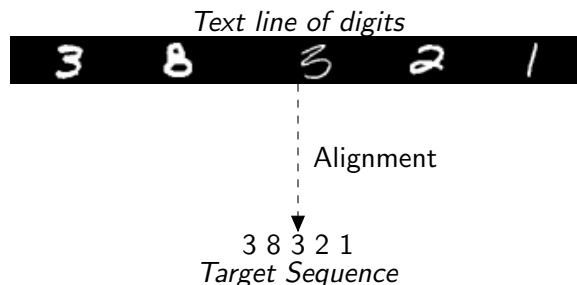


Figure 2: Example of alignment between an input sequence and weakly labeled target sequence. Note that the input sequence does not require to have a bounding box for each character.

machine translation tasks). In this previous example, the segmentation of words is a relatively easy task. In contrast, the segmentation in Speech Recognition and Optical Character Recognition (OCR) requires a vast human effort. For example, consider the human effort of annotating bounding boxes for each character on this page.

In this work, we are interested in exploiting a training algorithm where LSTM can align an input sequence and target sequence. For example, the input sequence can be a text line (e.g., a sequence of hand-written digits) that represents the number “38321”, and the target sequence is the string “38321”. Figure 2 shows an example of the weakly labeled sequence. It can be observed the length of the input sequence is larger than the target sequence. Graves et al. (2006) proposed *Connectionist Temporal Classification (CTC)*. The authors have included an extra class, called *blank class (b)*. So, the target sequence is re-written with the blank class “b3b8b3b2b1b”. The intuition of blank class is to learn the transition between digits (for example, from 3 to 8) and to handle repeated characters (*ll* from the word “hello”). After extending the target sequence, CTC layer exploits the similarities between LSTM and *Hidden Markov Models (HMM)*. In this case, LSTM uses a forward-backward procedure that is similar to HMM training algorithm. CTC-forward-backward step requires two recursive variables forward (*fw*) and backward (*bw*) for generating the target vector y_t . A *forward-backward* algorithm employs the output vectors of LSTM. The idea is to propagate the forward and backward the probabilities on the target sequence. Finally, the output of the forward-backward algorithm is a target sequence for training LSTM. The final step of the CTC training is to predict the label sequence given an unknown input sequence. This step is called *decoding*, and two methods have been proposed: Best Path Decoding and Prefix Search Decoding. Figure 3 shows an example of LSTM classification. Please refer to the original paper for more

4. Multimodal Symbolic Association

The multimodal scenario is defined by a multimodal sequence (visual and audio). Both channels represent the same sequence of semantic concepts and are defined by

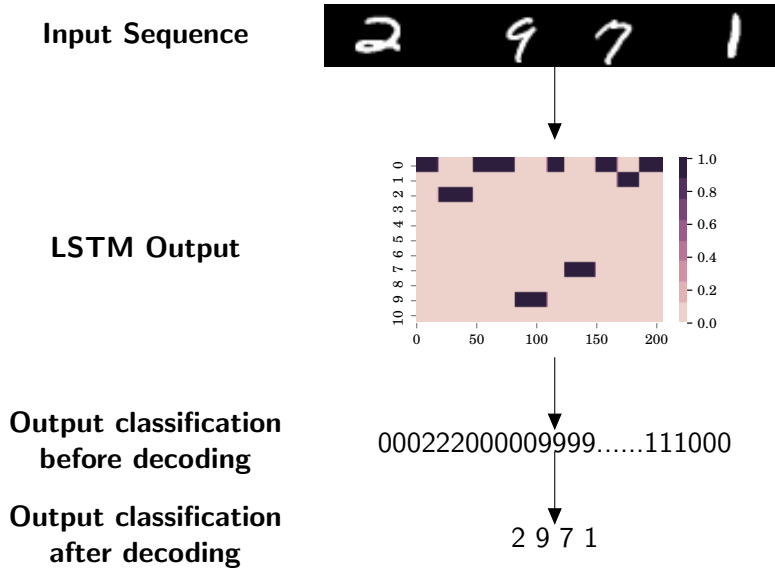


Figure 3: Example of the LSTM classification based on CTC training. The *blank class* is represented by the index zero. Note that the output classification is the same as the input sequence.

$$\mathbf{X}_v = \{\mathbf{x}_{v,1}, \dots, \mathbf{x}_{v,t1}\}, \tag{10}$$

$$\mathbf{X}_a = \{\mathbf{x}_{a,1}, \dots, \mathbf{x}_{a,t2}\}, \tag{11}$$

where $\mathbf{x}_{v,*} \in \mathbb{R}^{n1}$ and $\mathbf{x}_{a,*} \in \mathbb{R}^{n2}$ are the input vectors for the (v)isual and (a)udio modalities (respectively). The semantic concepts are represented by $SeC = \{s_1, \dots, s_c\}$ where c is the vocabulary size. Thus, the ordered sequence for each input sequence is defined by

$$SeC_v = \{s_1, \dots, s_k\} \tag{12}$$

$$SeC_a = \{s_1, \dots, s_k\} \tag{13}$$

where $SeC_v \cap SeC_a = \{s_1, \dots, s_k\}$. The association proposed by Raue et al. (2015) exploits the benefits of CTC training where both sequences align to the same semantic sequence. With this in mind, two bidirectional LSTM networks are defined for each modality $LSTM_v$ and $LSTM_a$. In this manner, the training of $LSTM_v$ uses the latent space produced by $LSTM_a$, and vice versa. The model has two components. The first component is to learn the binding between semantic concepts and symbolic features (Section 4.1). The other component is the alignment between the multimodal space produced by both LSTMs (Section 4.2).

In general, the model works as follows. Initially, $LSTM_v$ and $LSTM_a$ receive as an input the visual and audio components of the multimodal sequence (respectively) Consequently, each network produces the output vectors of both modalities after the last time step

$$\mathbf{Z}_v = \{\mathbf{z}_{v,1}, \dots, \mathbf{z}_{v,t1}\}, \quad (14)$$

$$\mathbf{Z}_a = \{\mathbf{z}_{a,1}, \dots, \mathbf{z}_{a,t2}\}, \quad (15)$$

where $\mathbf{z}_{v,*}$ and $\mathbf{z}_{a,*} \in \mathbb{R}^c$ are the output vectors. At this point, each output sequence contributes for finding the most likely binding between semantic concepts and symbolic features. Additionally, the indexes 4 or 10 (via one-hot coding vector) can represent the semantic concept *duck*. This decision is made internally by the model. Therefore, two sets of *concept vectors* are introduced to each modality γ_i^v and $\gamma_i^a \in \mathbb{R}^c$, where $i = 1, \dots, c$. The motivation behind is to implement a *winning-take-all* rule where all semantic concepts and symbolic features have different binding relations between them (more details in Section 4.1). After determining the most likely binding, LSTM output of each modality and the binding are fed to CTC forward-backward step (*c.f.* Section 3). Consequently, the output of each modality are

$$\mathbf{Y}_v = \{\mathbf{y}_{v,1}, \dots, \mathbf{y}_{v,t1}\}, \quad (16)$$

$$\mathbf{Y}_a = \{\mathbf{y}_{a,1}, \dots, \mathbf{y}_{a,t2}\}, \quad (17)$$

So far, the described steps are applied independently to each modality. For learning the association and exploiting the multimodal latent space, both output vectors from CTC are aligned between them in the *time-axis* by *Dynamic Time Warping (DTW)* (Berndt & Clifford, 1994). Hence, training on one modality uses the latent space of the other modality, and vice versa. Figure 5 illustrates the training algorithm with an example.

4.1 Statistical Constraint for Semantic Binding

In this association scenario, one crucial constraint is related to the binding between semantic concepts and symbolic features, which is not defined before training. As mentioned, the binding between the semantic concepts and the output vectors is learned based a novel set of *concept vectors*. To be clear, note that two or more concepts cannot have the same vectorial representations. This component employs an EM-style algorithm. For explanation purposes, it is described considering only one LSTM and one set of *concept vectors* $\gamma_1, \dots, \gamma_c \in \mathbb{R}^c$. However, it can be applied to two LSTM networks independently.

The *E-step* predicts the mapping between semantic concepts in the sequence and the symbolic representation given the LSTM output and the *concept vectors*. The first step is to combine the weighting vectors with the output sequence, which is defined by

$$\hat{\mathbf{z}}_i = \frac{1}{T} \sum_{t=1}^T \text{power}(\mathbf{z}_t, \gamma_i), \quad i = 1, \dots, c \quad (18)$$

where \mathbf{z}_t is the LSTM output vector at time t , γ_i is the concept vector, T is the number of time steps of the sequence, and $\text{power}(\mathbf{z}_t, \gamma_i)$ is the *element-wise* power operation between the output vector \mathbf{z}_t at time step t and the concept vector γ_i . Then, a matrix is assembled by concatenating $\hat{\mathbf{z}}_1, \dots, \hat{\mathbf{z}}_c$. As a result, the assembled matrix represents the relation between semantic concepts (column) and the symbolic features (row).

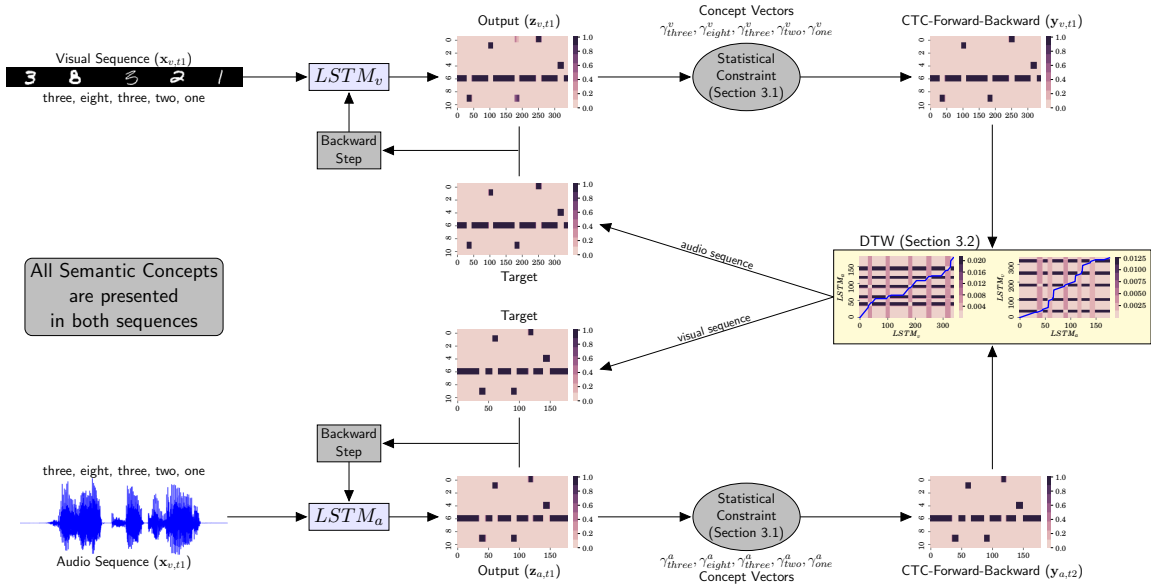


Figure 4: Association model based on two parallel bidirectional LSTM networks, which is proposed by Raue et al. (2015). Note that each semantic concept is presented in both channels. It can be observed that DTW module aligns the CTC layer produces by $LSTM_a$ to the CTC layer produces by $LSTM_v$. As a result, the target vector for training $LSTM_v$ is purely obtained from $LSTM_a$. The training of $LSTM_a$ based on $LSTM_v$ follows a similar process.

$$\hat{\mathbf{Z}} = [\hat{\mathbf{z}}_1, \dots, \hat{\mathbf{z}}_c] \quad (19)$$

$$\mathbf{e}_1, \dots, \mathbf{e}_c = g(\hat{\mathbf{Z}}) \quad (20)$$

where $g(\hat{\mathbf{Z}})$ is a row-column elimination, and $\mathbf{e}_1, \dots, \mathbf{e}_c \in \mathbb{R}^c$ are column vectors of a permutation of the identity matrix. For simplicity, the column vector \mathbf{e}_i can represent j -th identity vector where i and j can or cannot be the same. In other words, the column vector \mathbf{e}_4 can represent the 1-st identity vector (e.g., $\mathbf{e}_4 = [1 \ 0 \ 0 \ \dots]^T$). The row-column elimination procedure ranks all values in the matrix. Next, the position (col, row), where the maximum value is found and determines the row -th identity column vector \mathbf{e}_{col} . For example, the maximum value is found at (2, 5), and its correspondence vector is $\mathbf{e}_2 = [0 \ 0 \ 0 \ 0 \ 1 \ 0 \ \dots]^T$. All values of the previous column and the previous row are set to zero. This column-row elimination is applied c times. Hence, the vectors $\mathbf{e}_1, \dots, \mathbf{e}_c$ are the mapping between semantic concepts (columns) and their symbolic feature (rows).

The M -step updates the concept vectors given the LSTM output and the statistical distribution target. Hence, the cost function is defined by

$$cost_i = \left(\hat{\mathbf{z}}_i - \frac{1}{c} \mathbf{e}_i \right)^2, \quad i = 1, \dots, c \quad (21)$$

$$\gamma_i = \gamma_i - \alpha * \nabla_{\gamma_i} cost_i, \quad i = 1, \dots, c \quad (22)$$

where e_i is a column vector of the identity matrix that represents the semantic concept, α is the learning rate, and $\nabla_{\gamma_i} cost_i$ is the derivative w.r.t γ_i .

4.2 Dynamic Time Warping (DTW)

In this module, the goal is to combine both LSTMs in the latent space. This combination is possible because the multimodal sequence represents the same sequence of semantic concepts and the monotonic behavior of LSTM. The alignment is between CTC output of both modalities. Moreover, Berndt and Clifford (1994) proposed DTW for aligning two signals. Similarly, DTW can also utilize LSTM output sequences. DTW requires two steps for alignment of two signals. The first step is to calculate a cost matrix with the following relation

$$DTW[i, j] = dist[i, j] + \min \begin{cases} DTW[i - 1, j - 1] \\ DTW[i - 1, j] \\ DTW[i, j - 1] \end{cases} \quad (23)$$

where $dist[i, j]$ is the *Euclidean* distance between output vectors at timestep i of $LSTM_v$ and at timestep j of $LSTM_a$. The Second step is the alignment path between both LSTMs. In this case, the path is a set of tuples that maps one time step of one LSTM to another time step of the other LSTM. In other words, there is a function $f(s1, s2) : s1 \rightarrow s2$, where $s1$ is the source sequence and $s2$ is the target sequence. Afterwards, the loss function of one modality can use the other modality as a target, and vice versa. Hence, the losses are defined by

$$E_v(t) = f_{a \rightarrow v} \left(\mathbf{y}_{a,t'} \right)_t - z_{v,t}, \quad \text{where } t = 1, \dots, t1 \quad (24)$$

$$E_a(t) = f_{v \rightarrow a} \left(\mathbf{y}_{v,t'} \right)_t - z_{a,t}, \quad \text{where } t = 1, \dots, t2 \quad (25)$$

where $f_{a \rightarrow v}$ and $f_{v \rightarrow a}$ are the alignment path from audio to visual and from visual to audio (respectively), $z_{v,t}$ and $z_{a,t}$ are LSTM output vector at time t , $\mathbf{y}_{v,t'}$ and $\mathbf{y}_{a,t'}$ are the CTC-forward-backward steps. Note that t' is the time step in the source modality and time step t is the time step in the target modality.

5. Handling Missing Elements

The previous section described a multimodal association model that exploits weakly labeled samples based on CTC training. The initial assumption is that each channel in the multimodal sequence represents the same ordered sequence of semantic concepts. However, the initial assumption is a limitation. In this paper we present, an extension that handles missing elements in the sequences. The goal is to exploit semantic concepts that are presented in both modalities. Additionally, the multimodal combination can be boosted using a *max operation* that combines the best of each modality, whereas the previous model only

exploits one modality. More formally, the association task can be rewritten as follows

$$\mathbf{X}_v = \{\mathbf{x}_{v,1}, \dots, \mathbf{x}_{v,t1}\}, \quad (26)$$

$$\mathbf{X}_a = \{\mathbf{x}_{a,1}, \dots, \mathbf{x}_{a,t2}\}, \quad (27)$$

$$SeC_v = \{s_1, \dots, s_{k1}\}, \quad (28)$$

$$SeC_a = \{s_1, \dots, s_{k2}\} \quad (29)$$

where $SeC_v \cap SeC_a \neq \{\emptyset\}$ and $|SeC_v \cap SeC_a| \leq \min(|SeC_v|, |SeC_a|)$.

The coupling of both LSTMs occurs in DTW module. Similar to Section 4, this model follows the same procedure until DTW step. As a reminder, the output of DTW is an alignment function from one modality to the other modality. In other words, a function $f(s1, s2) : s1 \rightarrow s2$ that maps the closest signal to $s2$ (target) based on the values of $s1$ (source).

The two main differences with the model proposed by Raue et al. (2015) relies on two parts. The first part is to combine only semantic concepts that are shared between both channels because the probability distribution produced by LSTMs should be similar (even if different input feature space produces them). Furthermore, the max operation is a common approach for combining two vectors. The second part is related to the semantic concepts that are presented only in one channel. In this case, it is not required to use another modality. Thus, this step can be formulated as follows

$$\hat{\mathbf{y}}_{a \rightarrow v, t} = \begin{cases} \max(y_{a,t'}^k, y_{v,t}^k) & \text{semantic concept } k \text{ is presented in both modalities} \\ y_{v,t}^k & \text{otherwise} \end{cases} \quad (30)$$

$$\hat{\mathbf{y}}_{v \rightarrow a, t} = \begin{cases} \max(y_{v,t'}^k, y_{a,t}^k) & \text{semantic concept } k \text{ is presented in both modalities} \\ y_{a,t}^k & \text{otherwise} \end{cases} \quad (31)$$

where $y_{v,t}^k$ and $y_{a,t}^k$ are the probabilities of semantic concept k at time step t , $y_{v,t'}^k$ and $y_{a,t'}^k$ are the latent space produced by CTC layer for each modality, and function *max* is the *element-wise* maximum operation. Note that the goal is to assemble the vectors $\hat{\mathbf{y}}_{a \rightarrow v, t}$ and $\hat{\mathbf{y}}_{v \rightarrow a, t}$, which combine shared and non-shared semantic concepts using the scalar values $y_{v,t}^k, y_{a,t}^k, y_{v,t'}^k$, and $y_{a,t'}^k$. Afterwards, the target vectors are used based on this operation. The Equations 24 and 25 are updated

$$E_v(t) = \hat{\mathbf{y}}_{a \rightarrow v, t} - \mathbf{z}_{v,t}, \text{ where } t = 1, \dots, t1 \quad (32)$$

$$E_a(t) = \hat{\mathbf{y}}_{v \rightarrow a, t} - \mathbf{z}_{a,t}, \text{ where } t = 1, \dots, t2 \quad (33)$$

6. Experimental Design

The goal of our paper is to evaluate the association between different channels of a multimodal sequence. One significant constraint is the weakly annotation of each channel. With this in mind, we generated a multimodal dataset, which is described in Section 6.1. Additionally, a description of input features and model parameters is mentioned in Section 6.2

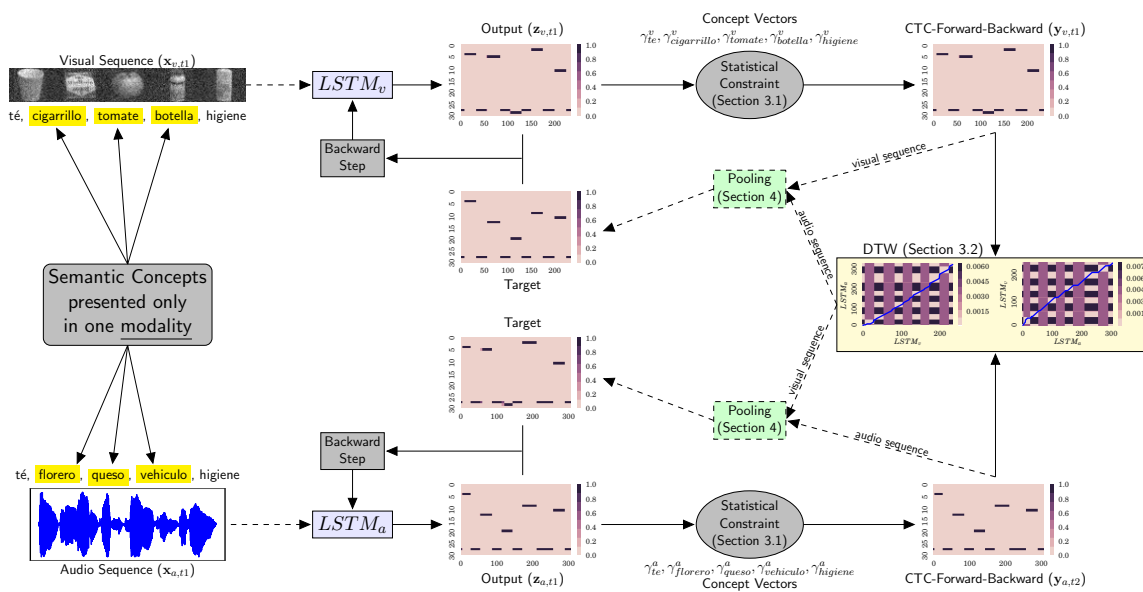


Figure 5: The new model can handle semantic concepts that are presented in one or two modalities. In this work, we include a module that combines the audio and visual information based on the presence of semantic concepts. Moreover, both channels are combined if they have the same semantic concept. Otherwise, there is no combination. Also, the *max operation* improves the combination of the channels.

6.1 Datasets

We generated several multimodal datasets where the elements of the sequence are missing in one or both modalities, but the relative order between the elements is the same. For example, a visual semantic concept sequence is a text line of digits “2 4 7”, and an audio semantic concept sequence can be represented by “two seven”. In this case, we assumed a simplified scenario of symbol grounding, where the continuity of semantic concepts is different on each modality. The visual component is a horizontal arrangement of isolated objects, and the audio component is spoken semantic concepts of some elements of the visual component, and vice versa. We want to point out that the visual component is similar to a panorama view. The procedure for generating the multimodal datasets is explained.

6.1.1 GENERATING SEMANTIC SEQUENCES

Two scenarios are considered for generating the semantic sequences for each modality: missing elements in both modalities and one modality. For the first scenario, we generated a sequence of ten semantic concepts. Later, we randomly remove between zero and five elements from each sequence. As a result, two different sequences for two different modalities were obtained with few common elements between them. For the second scenario, we follow a similar procedure. In that case, one modality has a sequence with ten semantic

concepts, and the other modality has missing elements. For example, $|S_a| = 10$ and $|S_v| = S_a$ —missing elements. In addition, our vocabulary has 30 semantic concepts in Spanish: *oso, bote, botella, bol, caja, carro, gato, queso, cigarrillo, gaseosa, bebida, pato, cara, comida, hamburguesa, higiene, liquido, loción, cebolla, pimentón, pera, redondo, sanduche, cuchara, té, teléfono, tomate, florero, vehículo, madera.*

6.1.2 VISUAL COMPONENT

We used a subset of 30 objects from COIL-100 (Nene, Nayar, & Murase, 1996) that is a standard dataset of 100 isolated objects. Each isolated object has 72 views at different angles with a black background. After selecting the object for the sequence, each object was converted to grayscale and re-scaled to 32 x 32 pixels. The visual components are composed by horizontally stacking isolated objects. Moreover, a random noise was added (as a background) to the final image to increase the difficulty of segmenting it. While the training set contains images of odd angles, the testing set has images of even angles.

6.1.3 AUDIO COMPONENT

We recorded each semantic concept two times from twelve different subjects who are Spanish native speakers (five female and seven male speakers) from different countries of Center and South America. Afterwards, concatenating isolated semantic concepts generates the audio sequences. The training set contains eleven voices for training, whereas the testing set contains two.

6.1.4 TRAINING AND TESTING MULTIMODAL DATASETS

We have generated three different multimodal configurations for evaluating our model. The first setup has missing elements in both modalities. While the second setup has ten semantic concepts in the visual component, the audio modality has a fixed number of missing elements. Thus, we can evaluate the impact of the missing elements. This setup covers from zero to five missing semantic concepts. The third dataset has a similar idea concerning the second dataset. In this case, we are testing an audio sequence with ten semantic concepts, but the visual component has a fixed number of missing elements. Additionally, 1,000 multimodal sequences have been generated for each subject in each setup. We follow a 5-fold cross-validation scheme where for each run eleven subjects are selected for training and the remaining two subjects are used for testing. One example with both elements missing from both modalities is shown in Figure 6.

6.2 Input Features and LSTM Setup

We did not apply any pre-processing step for the visual component. In contrast, the audio component was converted to Mel-Frequency Cepstral Coefficient (MFCC) using HTK toolkit¹. The audio representation is a vector of 123 components: a Fourier filter-bank with 40 coefficients (plus energy), including the first and second derivatives. All audio and visual components were normalized to have zero mean and standard deviation one.

1. <http://htk.eng.cam.ac.uk>

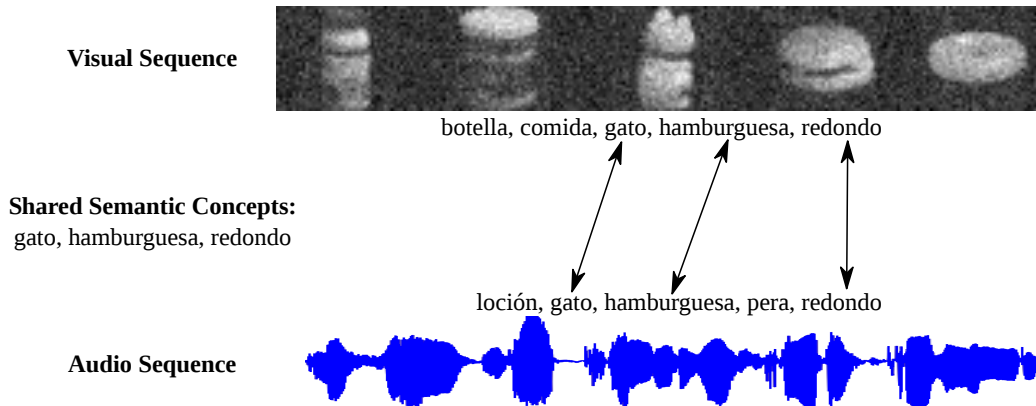


Figure 6: Example of the multimodal dataset. It can be observed that only three elements are the same on both modalities.

Also, the presented extension was compared against the original model introduced by Raue et al. (2015). Additionally, we compared the extension against LSTM with CTC layer and a predefined coding scheme. The parameters of the visual LSTM were: 40 memory cells, learning rate 0.0001, and momentum 0.9. On the other hand, the audio LSTM had 100 memory cells, and the learning rate and momentum are the same as in the visual LSTM. Furthermore, the learning rate in the statistical constraint was set to 0.001. The parameters are selected based on the best performance of LSTM that is trained independently on each modality.

7. Results and Discussion

As mentioned previously, the assumption of the original model was to represent the same semantic concept sequence in both modalities. In other words, a one-to-one relationship exists between modalities. In contrast, in this work, our assumption is more challenging because the semantic concept in one modality can be or cannot be present in the other modality. We evaluate the multimodal association task using *Association Accuracy (AAcc)*, which is defined by the following equation

$$AAcc = \frac{\sum_{i=1}^N LCS(output_{a,i}, output_{v,i}, gt_{a,i}, gt_{v,i})}{\sum_{i=1}^N LCS(gt_{a,i}, gt_{v,i})}, \quad (34)$$

where LCS is the length of the longest common sequence, $output_{a,i}$ and $output_{v,i}$ are the output classification of each modality, $gt_{a,i}$ and $gt_{v,i}$ are the ground-truth labels of each modality, and N is the number of elements in the dataset. Therefore, we evaluate the association between the common elements. Our model not only learns the association but also learns to classify each modality. With this in mind, we also reported the Label Error Rate (LER) as a performance metric, which is defined by

Table 1: Association Accuracy (%) and Label Error Rate (%) from the multimodal dataset that has missing elements in both modalities. It can be seen that the original model performs worse than the proposed combination. Furthermore, the presented extension reached similar results to LSTM (trained for the easier classification task, and not for association) and under some conditions reaches better results (audio component).

Model	Association Accuracy (%)	Label Error Rate (%)	
		visual	audio
LSTM + CTC (baseline)	70.68 ± 6.12	0.14 ± 0.14	35.84 ± 5.35
Original Model (Raue et al. (2015))	19.59 ± 8.90	7.00 ± 2.42	79.01 ± 9.51
Our work	71.52 ± 11.85	0.97 ± 1.58	33.09 ± 10.38

$$LER = \frac{1}{N} \sum_{i=1}^N \frac{ED(output_i, gt_i)}{|gt_i|} \quad (35)$$

where $output_i$ is the output classification, gt_i is the ground-truth, and $ED(output_i, gt_i)$ is the edit distance between the output classification and the ground-truth. As a reminder, the training set has 11,000 multimodal sequences, whereas the testing set has 2,000 multimodal sequences. In this work, we have reported the average results.

Table 1 summarizes the performance of LSTM trained with a predefined coding scheme, the original model, and the presented extension. Those results are divided into two parts as follows. First, the proposed extension handles missing elements in multimodal sequences better than the original model. It can be inferred that the max operation keeps the strongest of the common semantic concepts between modalities. Note that the representations update the weights in the backward step. Second, the proposed extension reaches similar results to the standard LSTM. In this case, LSTM was trained in each modality independently. As a reminder, we mentioned two setups for classification tasks: the traditional setup and the setup used in this work. We want to point out that the visual LSTM boost the performance of audio sequences compared to LSTM. As a result, our model reaches lower Label Error Rate in the audio sequences than the standard LSTM trained only in audio sequence.

Another outcome of this work is the conformity of the symbolic structure in both modalities, even with missing elements. In this case, the model does not know the relationship between semantic concepts and symbolic features before training. As a result, both LSTM networks learn to agree that the same semantic concepts require the same coding scheme regardless of the modalities. Figure 7 shows examples of the coding scheme agreement. It can be observed that both LSTM networks learn to classify the object-word relation in weakly labeled multimodal sequences. Moreover, the common concepts in both modalities are represented by a similar symbolic feature and located at the right position in the sequence. For example, the semantic concept “madera” (first element at the visual and audio

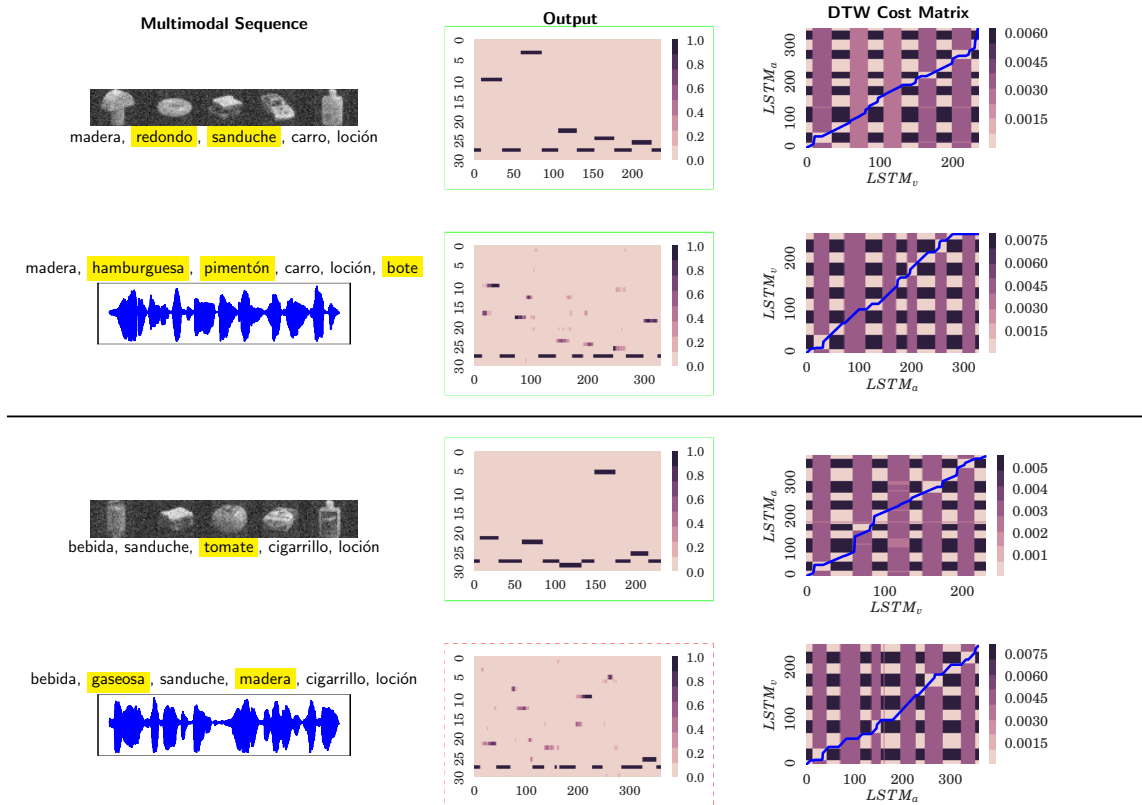


Figure 7: Several examples of the output classification and DTW cost matrices. The first multimodal sequence shows an example, in which both output classifications are correct (solid green square). The second multimodal sequence shows one correct and one incorrect output classification (dashed red square).

components) is represented by the index “10” in both modalities². Note that not only the common elements but also the missing elements are classified correctly. Furthermore, the common semantic concepts are correctly classified even if LSTM does not correctly classify all semantic concepts of the sequence. The second example in Figure 7 shows the semantic concept “loción” is represented by index “25”.

In addition to the considerations we made so far, we were also interested in the robustness of the presented model against the number of missing elements. With this in mind, we generated several datasets where one modality has ten semantic concepts, and the other has only a fixed number of missing elements from the ten semantic concepts. Figure 8 shows the *Association Accuracy* of the original model and the presented model for handling missing elements. First, the original model (solid blue line) decreases its performance when

2. There are some cases that represent one semantic concept with two different coding vectors for each network. However, both networks retrieve correctly the same concept regardless of the different coding scheme.

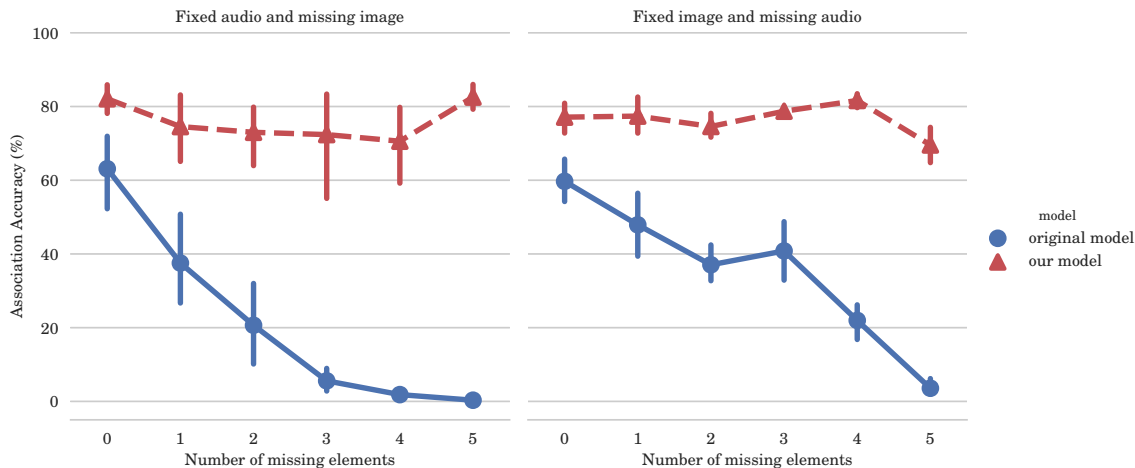


Figure 8: Association of two multimodal setups. In this case, one modality has ten semantic concepts, and the other modality has a fixed number of missing elements. The presented model (triangle) outperforms to the original model (circle) regardless of the modality and number of missing elements.

the number of missing elements increases in both modalities. These results were expected because the original model relies on *the one-to-one relation* between modalities. Second, we recognize that the presented model (dashed red line) shows a better performance compared to the original model (solid blue line) in both modalities. Thus, we may conclude that the presented model does not reduce its performance even if 50% of elements are missing in one of the modalities. Note that the max operation boosts the performance of our model when there are zero missing elements.

Figure 9 shows that Label Error Rate of each modality. One pattern appears at zero missing elements. Our model reaches better performance than the original model because of the audio modality. In this case, the combination of audio and visual latent spaces helps to reduce the error. Moreover, it can be observed that the error of our model does not increase the same rate as the original model. Another setup was also evaluated where the training set has sequences with missing elements whereas the testing set does not have missing ones. The results are similar to Figures 8 and 9. These results are expected because LSTM network has already learned the discriminant information during training and does not require the condition of missing elements in the prediction step.

8. Conclusions

In summary, we have presented a solution inspired by the *symbol grounding problem* for the object-word association problem. Additionally, the model relies on multimodal sequences (visual and audio) where the semantic elements can be presented in one or both modalities. Note that the cognitive association adds a new aspect to neural network architectures

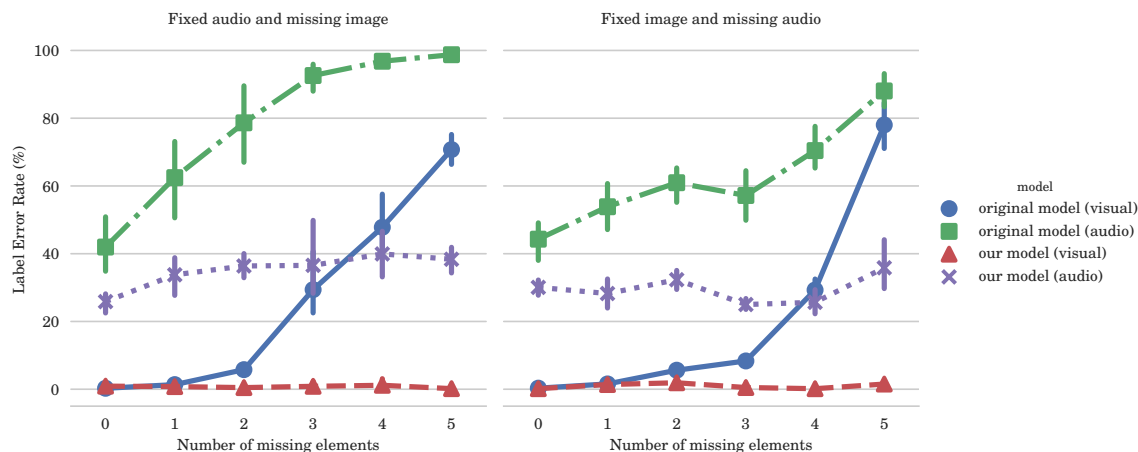


Figure 9: Label Error Rate (%) of each modality for each multimodal setup. It can be observed that the presented model (visual-triangle and audio-X) keeps similar performances across the number of missing elements. In contrast, the original model (visual-circle and audio-square) increases the error w.r.t. the number of missing elements.

because of the *Symbolic Feature* axis. In this case, each LSTM network learn to agree on the same relationship between semantic concepts and symbolic features (represented by the output vectors). However, we believe that one interesting direction is to analyze in more detail if the model is cognitive plausible. Further work is planned for more realistic scenarios where the visual component is not segmentable. Moreover, we are interested in extending the word-association problem between a two-dimensional image and speech. With this in mind, we will incorporate visual attention mechanism in synchronization with speech. In the future, the model will be evaluated to add more modalities, for instance visual, audio, and motor sensors, the three modalities can be aligned between them. Note that each sensorial input collects data of the same action. Two approaches can be considered for the alignment step. One option is to apply DTW for three dimensions (similar to Wöllmer et al., 2009). The other approach is to align between each pair signals and evaluate the most suitable relation. For example, visual-audio, motor-visual, and audio-motor. Finally, the human language development relies on the relationship between abstract concepts and the real world collected by the sensory input. The scenario of the symbol grounding problem might be considered as simple. However, many questions remain still open (Needham, Santos, Magee, Devin, Hogg, & Cohn, 2005; Steels, 2008).

References

- Andersen, E. S., Dunlea, A., & Kekelis, L. (1993). The impact of input: language acquisition in the visually impaired. *First Language*, 13(37), 23–49.
- Asano, M., Imai, M., Kita, S., Kitajo, K., Okada, H., & Thierry, G. (2015). Sound symbolism scaffolds language development in preverbal infants. *cortex*, 63, 196–205.
- Berndt, D. J., & Clifford, J. (1994). Using Dynamic Time Warping to Find Patterns in Time Series., 359–370.
- Breuel, T., Ul-Hasan, A., Al-Azawi, M., & Shafait, F. (2013). High-performance ocr for printed english and fraktur using lstm networks. In *Document Analysis and Recognition (ICDAR), 2013 12th International Conference on*, pp. 683–687.
- Byeon, W., Breuel, T. M., Raue, F., & Liwicki, M. (2015). Scene labeling with lstm recurrent neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3547–3555.
- de Penning, H. L. H., d’Avila Garcez, A. S., Lamb, L. C., & Meyer, J.-J. C. (2011). A neural-symbolic cognitive agent for online learning and reasoning. In *IJCAI Proceedings-International Joint Conference on Artificial Intelligence*, Vol. 22, p. 1653.
- Gershkoff-Stowe, L., & Smith, L. B. (2004). Shape and the first hundred nouns.. *Child development*, 75(4), 1098–114.
- Graves, A., Fernández, S., Gomez, F., & Schmidhuber, J. (2006). Connectionist temporal classification. In *Proceedings of the 23rd international conference on Machine learning - ICML '06*, pp. 369–376, New York, New York, USA. ACM Press.
- Harnad, S. (1990). The symbol grounding problem. *Physica D: Nonlinear Phenomena*, 42(1), 335–346.
- Hochreiter, S. (1998). The Vanishing Gradient Problem During Learning Recurrent Neural Nets and Problem Solutions. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 06(02), 107–116.
- Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Computation*, 9(8), 1735–1780.
- Iqbal, M. S., & Silver, D. L. (2016). A scalable unsupervised deep multimodal learning system.. In *FLAIRS Conference*, pp. 50–55.
- Karpathy, A., Joulin, A., & Fei-Fei, L. F. (2014). Deep fragment embeddings for bidirectional image sentence mapping. In *Advances in Neural Information Processing Systems*, pp. 1889–1897.
- Nakamura, T., Araki, T., Nagai, T., & Iwahashi, N. (2011). Grounding of word meanings in latent dirichlet allocation-based multimodal concepts. *Advanced Robotics*, 25(17), 2189–2206.
- Needham, C. J., Santos, P. E., Magee, D. R., Devin, V., Hogg, D. C., & Cohn, A. G. (2005). Protocols from perceptual observations. *Artificial Intelligence*, 167(1), 103–136.
- Nene, S. A., Nayar, S. K., & Murase, H. (1996). Columbia object image library (coil-100). Tech. rep..

- Raue, F., Byeon, W., Breuel, T., & Liwicki, M. (2015). Symbol Grounding in Multimodal Sequences using Recurrent Neural Network. In *Workshop Cognitive Computation: Integrating Neural and Symbolic Approaches at NIPS 15*.
- Sohn, K., Shang, W., & Lee, H. (2014). Improved multimodal deep learning with variation of information. In *Advances in Neural Information Processing Systems*, pp. 2141–2149.
- Spencer, P. E. (2000). Looking without listening: is audition a prerequisite for normal development of visual attention during infancy?. *Journal of deaf studies and deaf education*, 5(4), 291–302.
- Srivastava, N., & Salakhutdinov, R. R. (2012). Multimodal learning with deep boltzmann machines. In *Advances in neural information processing systems*, pp. 2222–2230.
- Steels, L. (2008). The symbol grounding problem has been solved, so whats next ?. *Symbols, Embodiment and Meaning. Oxford University Press, Oxford, UK*, pp. 223–244.
- Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pp. 3104–3112.
- Vinyals, O., Toshev, A., Bengio, S., & Erhan, D. (2014). Show and tell: A neural image caption generator. *arXiv preprint arXiv:1411.4555*.
- Werbos, P. J. (1990). Backpropagation through time: what it does and how to do it. *Proceedings of the IEEE*, 78(10), 1550–1560.
- Wöllmer, M., Al-Hames, M., Eyben, F., Schuller, B., & Rigoll, G. (2009). A multidimensional dynamic time warping algorithm for efficient multimodal fusion of asynchronous data streams. *Neurocomputing*, 73(1), 366–380.
- Yu, C., & Ballard, D. H. (2004). A multimodal learning interface for grounding spoken language in sensory perceptions. *ACM Transactions on Applied Perception (TAP)*, 1(1), 57–80.