Preface

The demand for more sophisticated natural human-computer and human-robot interactions is rapidly increasing as users become more accustomed to conversation-like interactions with AI and NLP systems. Such interactions require not only the robust recognition and generation of expressions through multiple modalities (language, gesture, vision, action, etc.), but also the encoding of situated meaning.

When communications become multimodal, each modality in operation provides an orthogonal angle through which to probe the computational model of the other modalities, including the behaviors and communicative capabilities afforded by each. Multimodal interactions thus require a unified framework and control language through which systems interpret inputs and behaviors and generate informative outputs. This is vital for intelligent and often embodied systems to understand the situation and context that they inhabit, whether in the real world or in a mixed-reality environment shared with humans.

This workshop intends to bring together researchers who aim to capture elements of multimodal interaction such as language, gesture, gaze, and facial expression with formal semantic representations. We provide a space for both theoretical and practical discussion of how linguistic co-modalities support, inform, and align with “meaning” found in the linguistic signal alone. In so doing, the MMSR workshop has several goals:

1. To provide an opportunity for computational semanticists to critically examine existing NLP semantic frameworks for their validity to express multimodal elements;

2. To explore and identify challenges in the semantic representation of co-modalities cross-linguistically and cross-culturally;

3. To gain understanding of domains and tasks where certain semantic frameworks (multimodal or not) are most effective and why.

We would like to thank the authors, reviewers, invited speakers, and IWCS 2021 organizers for making this workshop possible. We look forward to an exciting workshop.

Lucia Donatelli, Nikhil Krishnaswamy, Kenneth Lai, and James Pustejovsky
Organizers:
Lucia Donatelli, Saarland University
Nikhil Krishnaswamy, Colorado State University
Kenneth Lai, Brandeis University
James Pustejovsky, Brandeis University

Program Committee:
Nicholas Asher, Institute de Recherche en Informatique de Toulouse
Claire Bonial, Army Research Lab
Harry Bunt, Tilburg University
Stergios Chatzikyriakidis, University of Gothenburg
Sandy Ciroux, University of Konstanz
Robin Cooper, University of Gothenburg
Simon Dobnik, University of Gothenburg
Maria (Masha) Esipova, University of Oslo
Anette Frank, Heidelberg University
Felix Gervits, Army Research Lab
Jonathan Ginzburg, Université de Paris
Casey Kennington, Boise State University
Stefan Kopp, Bielefeld University
Staffan Larsson, University of Gothenburg
Andy Lücking, Université de Paris, Goethe University Frankfurt
Larry Moss, Indiana University
Francisco Ortega, Colorado State University
Gözde Gül Şahin, Technical University of Darmstadt
Philippe Schlenker, Institut Jean-Nicod - Ecole Normale Supérieure, Paris
Nathan Schneider, Georgetown University
Candy Sidner, Worcester Polytechnic Institute
Juris Škilters, University of Latvia
Benjamin Spector, Institut Jean-Nicod - Ecole Normale Supérieure, Paris
David Traum, University of Southern California
Alexis Wellwood, University of Southern California
Bram Willemsen, KTH Royal Institute of Technology

Invited Speakers:
Chiara Bonsignori, Consiglio Nazionale delle Ricerche
Matthias Scheutz, Tufts University
Virginia Volterra, Consiglio Nazionale delle Ricerche
# Table of Contents

**What is Multimodality?**  
Letitia Parcalabescu, Nils Trost and Anette Frank .................................................. 1

**Are Gestures Worth a Thousand Words? An Analysis of Interviews in the Political Domain**  
Daniela Trotta and Sara Tonelli ................................................................. 11

**Requesting clarifications with speech and gestures**  
Jonathan Ginzburg and Andy Luecking .......................................................... 21

**Seeing past words: Testing the cross-modal capabilities of pretrained V&L models on counting tasks**  
Letitia Parcalabescu, Albert Gatt, Anette Frank and Iacer Calixto ............................. 32

**How Vision Affects Language: Comparing Masked Self-Attention in Uni-Modal and Multi-Modal Transformer**  
Nikolai Ilinykh and Simon Dobnik .............................................................. 45

**EMISSOR: A platform for capturing multimodal interactions as Episodic Memories and Interpretations with Situated Scenario-based Ontological References**  
Selene Baez Santamaria, Thomas Baier, Taewoon Kim, Lea Krause, Jaap Kruijt and Piek Vossen 56

**Annotating anaphoric phenomena in situated dialogue**  
Sharid Loáiciga, Simon Dobnik and David Schlangen .............................................. 78

**Incremental Unit Networks for Multimodal, Fine-grained Information State Representation**  
Casey Kennington and David Schlangen ......................................................... 89

**Teaching Arm and Head Gestures to a Humanoid Robot through Interactive Demonstration and Spoken Instruction**  
Michael Brady and Han Du .................................................................................. 95

**Building a Video-and-Language Dataset with Human Actions for Multimodal Logical Inference**  
Riko Suzuki, Hitomi Yanaka, Koji Mineshima and Daisuke Bekki ................................. 102
Workshop Program

Wednesday, June 16, 2021

16:00–16:15  Introduction

16:15–17:00  Invited Talk: From action to language through gesture
  Virginia Volterra and Chiara Bonsignori

17:05–17:35  Oral Session 1

  What is Multimodality?
  Letitia Parcalabescu, Nils Trost and Anette Frank

  Are Gestures Worth a Thousand Words? An Analysis of Interviews in the Political Domain
  Daniela Trotta and Sara Tonelli

  Requesting clarifications with speech and gestures
  Jonathan Ginzburg and Andy Luecking

  Matthias Scheutz

18:30–19:10  Oral Session 2

  Seeing past words: Testing the cross-modal capabilities of pretrained V&L models on counting tasks
  Letitia Parcalabescu, Albert Gatt, Anette Frank and Iacer Calixto

  How Vision Affects Language: Comparing Masked Self-Attention in Uni-Modal and Multi-Modal Transformer
  Nikolai Ilinykh and Simon Dobnik

  EMISSOR: A platform for capturing multimodal interactions as Episodic Memories and Interpretations with Situated Scenario-based Ontological References
  Selene Baez Santamaria, Thomas Baier, Taewoon Kim, Lea Krause, Jaap Kruijt and Piek Vossen

  Annotating anaphoric phenomena in situated dialogue
  Sharid Loáiciga, Simon Dobnik and David Schlangen
Wednesday, June 16, 2021 (continued)

19:15–19:45 Poster Session

*Incremental Unit Networks for Multimodal, Fine-grained Information State Representation*
Casey Kennington and David Schlangen

*Teaching Arm and Head Gestures to a Humanoid Robot through Interactive Demonstration and Spoken Instruction*
Michael Brady and Han Du

*Building a Video-and-Language Dataset with Human Actions for Multimodal Logical Inference*
Riko Suzuki, Hitomi Yanaka, Koji Mineshima and Daisuke Bekki

19:45–20:00 Closing
Abstract

The last years have shown rapid developments in the field of multimodal machine learning, combining e.g., vision, text or speech. In this position paper we explain how the field uses outdated definitions of *multimodality* that prove unfit for the *machine learning* era. We propose a new *task-relative* definition of *(multi)modality* in the context of multimodal machine learning that focuses on representations and information that are *relevant* for a given machine learning task. With our new definition of multimodality we aim to provide a missing foundation for multimodal research, an important component of language grounding and a crucial milestone towards NLU.

1 Introduction

The holy grail of NLP is natural language understanding (NLU). As previously argued, NLU cannot be achieved by learning from text alone (Bender and Koller, 2020). Instead, an important step towards NLU is grounding language, especially in sight and sounds (Bisk et al., 2020). There is thus great interest in the field of NLP to go beyond the textual modality and to conduct multimodal machine learning (ML) research.

Multimodal ML has made great progress during the last years. Neural architectures are employed in tasks that go beyond single modalities. E.g., language is integrated with vision in Visual Question Answering (Antol et al., 2015), Visual Commonsense Reasoning (Zellers et al., 2019), Visual Dialogue (Das et al., 2017), or Phrase Grounding (Plummer et al., 2015). Audio signal processing has made advances in speech recognition (Nassif et al., 2019) and (visual) speech synthesis (Alam et al., 2020). But ML applications may reach beyond modalities that are familiar to us: Astronomical and medical imaging techniques record wavelengths outside of what we call visible light. Genetic research measures signals alien to human perception, like activity and structure of molecules. Hence, we argue that current definitions of multimodality fall short of capturing the full space of multimodality in the ML era, and – even more seriously – that the field of multimodal ML, including vision and language integration, is lacking a proper definition of *multimodality*.

The point we aim to make in this position paper can be crystallized by asking the following questions regarding the input types shown on the left of Figure 1: *Are the data types shown on the left different from those that appear on the right, or are they instances of the same modality?* This question leads to different answers, depending on whether we take a human-centered as opposed to a machine-centered standpoint. In this position paper we reason that either standpoint is insufficient, therefore we develop a definition of multimodality.

Figure 1: Are these examples instances of the same modality? = the same; ≠ different. Depending on perspective, input data can be judged differently. Human- and machine-centered views would agree for (a) speech and text ≠, (b) images and text ≠. For (c), an image of text and text, the opinions could differ, while for (d), a visible light vs. infrared picture, humans could not even judge the infrared data, since it is not within their sensory capability.
that allows researchers to judge cases as displayed in Figure 1, as involving or not different modalities from a task-specific perspective.

Our contributions are three-fold:

i) We discuss existing definitions of multimodality and argue that they are insufficient to define multimodal tasks in the era of Machine Learning.

ii) We propose a task-relative definition of multimodality and illustrate it for a variety of (multimodal) tasks and data. We argue that a task-relative definition better suits the current age of diversified data, where the field attempts to create multimodally enabled intelligent machines.

iii) By providing a novel definition of multimodality, we offer a foundation for building a roadmap towards multimodality in NLP and beyond.

Multimodality is a new challenge in NLP with research focusing on grounding language into other modalities. But how can we sensibly choose other modalities to ground language in, if we are not clear about what kinds of modalities language can represent itself, and about what constitutes a modality in the first place? For building a roadmap towards Multimodality in NLP, we must establish common ground for what multimodal ML is, what are possible instantiations or specializations of multimodality, and what research questions it gives rise to. Pinning down what makes a task multimodal in terms of information diversity, as we set out to do in this paper, is one important research question to ask. And from deeper understanding of what makes a task multimodal, we may — conversely — derive novel tasks.

2 Multimodality and multimedia

It is difficult to write about multimodality in ML and not to trigger the question “What about multimedia? Isn’t multimodal machine learning actually just multimedia in machine learning?” But the use of the terms “multimodality” and “multimedia” in academic literature is very diverse. Attempts to compare and better capture these terms focus on different aspects and center on human society, human-to-human interaction and learning environments (Anastopoulou et al., 2001; Bezemer and Kress, 2008; Lauer, 2009). We, however, focus on multimodal machine learning where the information receiver and processor is a ML system.

To decide whether multimodality as used so far is the right term for the field of multimodal ML, we thus consult general definitions for “medium” and “modality” in the Oxford Advanced Learner’s Dictionary, rather than the academic literature. For “medium” we find:

(a) “a way of communicating information, etc. to people” comprising text, (moving) images, graphs, sounds.

(b) “a substance that something exists or grows in or that it travels through” being mainly applied in natural sciences.

“Modality” on the other hand is defined as follows:

(A) “the particular way in which something exists, is experienced or is done”

(B) “the kind of senses that the body uses to experience things” being applied in biology.

We argue that the term multimodality should be preferred in the context of ML, since multimodal ML aims to enhance world perception through ML systems (cf. def. (A)). Finding new ways of presenting information to humans (cf. def. (a)) is an endeavour that can benefit from multimodal ML while not being the main focus of multimodal ML research. For the rest of the paper, we will use the term “medium” or “media” in different contexts, referring to the meaning defined in (b).

In the following Section 3, we discuss how multimodality has been interpreted by the multimodal ML field and highlight the shortcomings of the corresponding definitions in Section 4.

3 How multimodality is defined – so far

In the multimodal ML literature and beyond, we find three general ways of defining “modality” or “multimodality”: not at all or etymologically (bypassing the problem), or by way of a human-centered or a machine-centered definition1.

Not at all or etymological Especially recent publications, as in Lu et al., 2020; Tan and Bansal, 2019; Gao et al., 2019, bypass a definition, assuming that the term is generally understood. Others offer an etymological definition: multimodal research involves not one, but multiple modalities

---

1For the scope of this paper, we disregard the statistical sense of “multimodality”, which describes a distribution with more than one peak. Such distributions can occur with any kind of data, unimodal or multimodal in the sense of “modality” we use for this paper.
(Zhang et al., 2020). Clearly, this definition leaves the notion of modality itself unexplicated.

**Human-centered** Popular definitions of multimodality rely on the human perceptual experience, as found in Baltrusaitis et al. (2019); Lyons (2016); Ngiam et al. (2011); Kress (2010). From this literature, we chose the following illustrative example, because the work focuses specifically on multimodality for ML, as is the interest of this paper:

“*Our experience of the world is multimodal – we see objects, hear sounds, feel texture, smell odors, and taste flavors. Modality refers to the way in which something happens or is experienced.*”  
Baltrusaitis et al. (2019)

This view appeals to humans, who are bound to their senses when experiencing the world. It is thus an intuitive explanation of the concept of multimodality, focusing on the propagation channels that human communication is adapted to (e.g., vision, sound).

Using this definition, one can agree for Figure 1.a that speech (hearing) and text (seeing) are different modalities. But decisions are less clear for images and text as in Figure 1.(b,c), as humans perceive both of them with their visual apparatus. Hence, as for written and depicted language, the human-centered definition contradicts the common conception in the community, that vision and language are different modalities, as in Lu et al. (2019); Su et al. (2019).

**Machine-centered** Another accepted perspective for defining multimodality is a machine-centered one, that focuses on the state in which information is transferred or encoded before being processed by a ML system:

“In the representation learning area, the word ‘modality’ refers to a particular way or mechanism of encoding information.”  
Guo et al., 2019

This definition is practical, focuses on the technical aspects of data representation, and captures how different types of inputs usually require specific programming solutions. For example, neural architectures typically use CNNs to encode images (exploiting 2d patches) and LSTMs to encode text (modeling sequential reading), exploiting the respective architecture’s inductive bias. From this viewpoint, the machine-centered definition naturally regards images vs. text as different modalities (cf. Figure 1). However, recent developments in neural architectures are challenging this view, since multimodal transformers are processing both images and text with transformer blocks (Lu et al., 2020; Tan and Bansal, 2019; Su et al., 2019).

4 Why we need a better definition

In this section, we indicate shortcomings of the existing definitions and motivate why time is ripe for a new definition of multimodality.

4.1 Human-centered

The human-centered definition (Section 3) is rooted in research on how information is communicated to humans through, e.g., GUIs (Bernsen, 2008; Jovanovic, 2015; Lyons, 2016). However, this definition does not cover the large gap between the rich sensorial capabilities of humans as opposed to machines, and leaves open in which ways the signals that can be perceived by the variety of human senses will be converted to specific types of inputs that can be given to the ML system. Moreover, while we think that the human biology and psychology are and should be a valuable inspiration for ML research, we contend that ML should look far beyond the human, into other organisms.

We will discuss three aspects of the gap between humans and machines: (a) the types of inputs their sensors can detect, (b) the range of these inputs, and (c) the importance of the processor – a human or a machine – that processes the inputs, playing the role of the (multimodal) agent that determines how inputs are perceived.

(a) Input types Human senses are not prescriptive for the sensorial apparatus of machines. Humans, on one side, are limited to the senses nature has gifted them with: sight, hearing, touch, etc. Machines, on the other hand, are only limited by the creativity of physicists or engineers: machines read tapes by measuring magnetization, or DVDs by detecting microscopic engravings. What modalities would the human-centered definition assign to these signals? Also, machines can surpass the human sensorial capabilities, by performing more exact measurements of e.g., humidity or pressure. This may not seem too relevant when restricting multimodality to the context of NLU, as language

---

Cf. section 4.1 for considerations about other human-centered formulations.

e.g., a self-driving car imitating a nematode’s nervous system, as in Lechner et al., 2020
was developed to fit the human experience of the world. In other fields however, ML systems are used on signals that are completely outside of the perception of humans, e.g., predicting gene regulation based on chromatin accessibility and transcription in biology (Minnoye et al., 2020).

(b) Input range Biological sensory detection systems are restricted to specific ranges of signals, and thus impose unnecessary limitations when applied to machines (Bernsen, 2008). Machines are limited only by the borders of human engineering\textsuperscript{4} and use manifold materials and physical phenomena that biological organisms adapted to their specific environment do not. For example, humans can detect and interpret only a tiny part of the electromagnetic spectrum (380 to 700 nanometers, Starr et al., 2010) and call it visible light. But machines can detect and (if programmed) process the whole electromagnetic spectrum, far beyond the visible light. Humans cannot perceive ultra-violet light – and hence this modality is non-existent to them because they cannot experience it. This again mainly impacts fields apparently remote to NLP, e.g., medicine, where imaging techniques are employed that measure wavelengths far outside of the perceptive range of humans. But we desire that future systems can combine their experience of e.g., both the visible light and other wavelengths with natural language (to aid medical diagnosis, for example).

(c) The processor: a human or an ML system? Humans have the innate ability of seeing objects, hearing sounds and delivering some interpretation of these signals. Machines, by contrast, can be clever sensors and count photons on a semiconductor or measure air pressure oscillations – yet without special programming they cannot interpret or derive information from the inputs. Detected physical quantities are then mapped to a voltage sequence interpreted as 0 if the voltage does not surpass a threshold, 1 if it does. Since ultimately, behind all data encodings, there are just 0s and 1s waiting to be interpreted by a program, we argue that multimodal ML research should focus on these programs, and that a definition of multimodality should answer the question: What are the challenges that a program needs to address when it is exposed to a new modality, rather than more unimodal data? In humans, evolution has already addressed this question, by specializing sensory organs and brain areas to the processing and interpretation of various input types (Schlosser, 2018). Multimodal ML is not there yet.

Language – humanity’s stroke of genius It is generally accepted in the multimodal ML literature that vision is one modality and language the other, typically given in form of text (Kafle et al., 2019). But humans hear speech i.e., spoken language, read written language with their visual apparatus, see signed languages, or feel Braille. The upside of the human-centered definition is that it captures the plurality of media that support the transmission of language and accordingly, it can assign different modalities to such different manifestations of language.

However, we are concerned with multimodality in ML and there are important edge cases for which the human-centered definition is inadequate: Does a screenshot of a text editor that displays the content of a .txt file containing a sentence $s$ represent a different modality than the data encoding in the text file? For a human, it is the same visual modality, since in both cases $s$ is visually perceived. But a machine needs very different programming in order to extract the same information about sentence $s$ from an image vs. the ASCII encoding of $s$ stored in a .txt file.

Having raised this criticism of the human-centered view on multimodality, it seems like the machine-centered view, focusing on the encoding of information, can offer a more viable interpretation of multimodality.

4.2 Machine-centered

Data representation While data representations are a challenge (Bengio et al., 2013), we argue that representations themselves should not be the defining trait for multimodal learning. For example, if we follow the machine-centered definition, an undirected unlabeled graph and its adjacency matrix have to be considered different modalities because they are different ways of representing a network. Similarly, PNG and JPEG are possible encodings of the same image and hence would be considered different modalities. This interpretation seems unintuitive for tasks like image recognition, where the image format does not play a big role and is usually homogenized in pre-processing.

Still, there are applications where the 4th dimension of PNGs is useful for encoding transparency.
or depth maps, by using the additional channel that PNG has compared to JPEG. Here we stand with a puzzle: Should data representation matter or not?

**When does representation impact information?**

Speech and handwriting are propagated through different media (e.g. air for speech, ink trail on paper for handwriting) and are represented differently in a computer – time series of amplitudes for speech, images or textual transcriptions for handwriting. Does this make them different modalities? Both speech and writing can be propagated through the same digital medium (fiber-optic cable) and if speech has been transcribed to text in pre-processing, they can have the same (ASCII) encoding. Does this make them the same modality?

When converting between media and representations, information can get lost. But without knowing the multimodal ML task and the information it requires, we cannot decide whether the loss was noteworthy or not. Below, we will argue that crucial factors for defining multimodality are not only an efficient encoding of information, but also the ML task itself.

5 A task-relative definition

We argue that a definition of multimodality for multimodal ML should relate to the task an ML system needs to solve, since it is the task that determines what information is relevant and in which representation this information can be efficiently stored. The human- and machine-centered definitions try to capture the essence of multimodality in a task-agnostic manner, relating it to categories of human experience, media, representations, and encodings. As shown in Section 4, these definitions turn out to be insufficient in view of the plurality of physical sensors, tasks and data types of the present.

Instead, our task-relative definition aims to answer the question: Under which conditions does a multimodal setting reveal crucial insight for the given task – and do we need multiple modalities? In our view, (i) different inputs can contribute specific information, but (ii) what is relevant information can only be determined in relation to the task at hand; and only by taking the task into account (iii) we can determine the status of the inputs as (possibly complementary) modalities.

5.1 Task-relative definition of multimodality

We propose the following task-relative definition of multimodality in ML that relates representation,

![](Figure_2.png)

**Figure 2:** Our definition of multimodality determines the modalities of input channels by considering i) how each input channel is represented, ii) whether the information units each input carries is complementary to each other iii) in relation to the ML task.

information and task as depicted in Figure 2.

A machine learning task is multimodal when inputs or outputs are represented differently or are composed of distinct types of atomic units of information.

Note that our definition covers both input and output to a system. In the following, we will primarily discuss and exemplify uni- and multimodal inputs. The same arguments and examples apply symmetrically for outputs.

Since the definition focuses on multimodal ML, when considering inputs, the representation of interest is the direct input to the learning system; any prior representational steps that are not seen, or filtered by pre-processing, are irrelevant for our definition. The reverse applies to outputs and post-processing.

Atomic units of information Information is one of the three dimensions we use to decide multimodality. We work with the information-theoretic interpretation of information that measures the decrease of uncertainty for a receiver (Shannon, 1948). In our case, the receiver is the ML system – whose objective is to solve a certain task. Hence we ask: What information does the input carry that is relevant for (or decreases uncertainty of) the task solution? As a defining criterion on whether two input channels contribute information from different modalities, we consider what types of atomic units of information they provide: they are of different types, if they cannot be captured by a 1-to-1 mapping between their domains. In this case, we speak of multiple modalities.

In other words, if we find that i) after pre-processing, inputs $x_1$ and $x_2$ are represented the same, we examine whether they live in the same modality or not by asking whether $x_1$ and $x_2$ are
formed of different atomic units: If ii) we can’t establish a bijective mapping between the domains of $x_1$ and $x_2$, then $x_1$ and $x_2$ must be composed of different atomic units, and even if i) was found to be the case, they represent different modalities.

Similarly, if iii) after pre-processing, $x_1$ and $x_2$ are represented differently, the task is multimodal.

The bijectivity criterion is lenient towards a task-dependent error for real-world applicability. Regardless of multimodal characteristics, low data quality and compression can cause loss of task-relevant information.

Information and its atomic units  Atomic units of information are not to be confused with information itself. In analogy, atomic units of information are to information what meters (unit of measuring space) are to objects of a certain size. Types of atomic units differ between each other like the meter (space) differs from the second (time). More concretely, atomic units of information apply to the data domain; information itself applies at the data sample level and can be accumulated by sampling more data (e.g. adding images of cats to a set of dog images). While it is impossible to have an 1-to-1 mapping between cats and dogs (information adds something previously unknown), our definition proposes bijective mappings between information units to identify modality-specific but data sample-invariant properties, e.g. edges and texture for images, time directionality for video, (abstract) concepts in text, nodes in graphs, etc.

We thus speak of a new modality when it contributes information that cannot be delivered by larger (but not infinite) amounts of unimodal data. Note that the same information captured in one modality may be encoded in a different modality, however, not necessarily with the same efficiency:

We can, in infinite time, describe every minute detail of a landscape unimodally through language. But it is clearly more efficient to capture the details of a landscape in a different modality, e.g., a photograph. In general, any kind of information can be reduced to a string of 1s and 0s, yet, depending on the information source and the given task, another representation might be more convenient.

Task-specificity  The task is the second crucial characteristic of multimodality because it determines (i) what information from the input is necessary to solve the task as discussed above, and (ii) it determines two crucial components: input and output and how they relate. Regarding inputs and their possible (multi)modality, we critically focus on the nature of the data that forms the direct input to the ML system, disregarding any data representations that the input may take during any pre-processing stages. Hence, we draw a sharp line between input encodings and representations that are part of the learning process, and that will be continuously refined – as opposed to input formats that are external to the learning process.

For our task-dependent definition of multimodality it will therefore not matter whether the original input was speech, but was transcribed to text if, for example, the task at hand does not require information that is specific to spoken language and can be conveyed by a textual representation of spoken input (see Figure 3). Also for image recognition it is not crucial whether images come as PNG or JPEG, and harmonizing the data is a matter of pre-processing.

Atomic units vs. data representation  The term data representation, as we use it here, refers to the encoding of information and data formats. However, atomic units are not bound by their technical
implementation – they constitute the informational content in relation to the task. For example, PNG and JPEG are undeniably different data representations which can, depending on the task, represent (a) two different types of atomic units of information, if the additional dimension in PNG is important for the task (e.g., in view of encoding transparency, or depth), or (b) a single type of atomic units, because a task-specific bijective mapping can be established that does not lose (task-relevant) information.

Not sensor-specific Our definition is especially robust to changes of medium and representation of information via data transfer or storage. We can thus neglect the physical or biological sensors that capture the data, and the encoding, transmission or storage of the data until it reaches the processor. This property makes our task-relative definition a robust definition: the constant change of representation that information may undergo, does not immediately span a new modality.

5.2 Applying the task-relative definition
We now apply our definition of multimodality to various examples – including edge cases – in order to demonstrate its breath, flexibility and robustness.

5.2.1 Images vs. text
By our definition, tasks working with images (stacks of intensity matrices) and text in ASCII format are multimodal, because they consist of different atomic units of information: An image can be truthfully described by multiple textual descriptions, and similarly, text translates equivocally to images (e.g., pictures of different hand writings). The decision becomes less clear-cut, if we are given two inputs: natural images and images of text. Both inputs are intensity matrices and therefore unimodal. However, if the task does not consider the differences in handwriting style and applies Optical Character Recognition on the images of text to obtain e.g. an ASCII text representation, the images of text turn into text in pre-processing. These, alongside the images, make the setting multimodal. In the next paragraph we will show examples of both of these approaches – using images of text unimodally, or using OCR to convert the images of text to the two modalities of images and text.

Examples concerning images of text For us as humans, it is tempting to see an image with text and think of it immediately as multimodal. But we experience text through our visual apparatus. For ML systems, images are always stacks of pixel matrices. Some images contain pixels that can be read by humans as text, others only contain pixels that depict an object – but the unit of information stays the same. To disentangle the issue of how images, text, and images of text relate to multimodality, we give two examples of multimodal models – winning models of the Hateful Meme Challenge dataset and CLIP.

The Hateful Meme Challenge dataset (Kiela et al., 2020) consists of memes – natural images with text rendered into the image as depicted in Figure 4. The task is to classify whether the memes are hateful or not. For now, winning systems (Zhu, 2020; Zhong, 2020) are using both the meme image and a string representation of the meme’s text besides the image. Thus, they solve the task multimodally, because text in string format can not be mapped 1-to-1 to the meme text, which could vary in font, color, or size. But ideally, models would be able to extract all relevant information only from the meme image – like humans do – without the help of an additional string representation of the meme text, therefore solving the task unimodally.

CLIP (Radford et al., 2021) is a multimodal model with impressive zero-shot applications in image recognition. It is trained to predict similar-

\[\text{Figure 4: Two memes from the Hateful Meme Challenge (Kiela et al., 2020). The rendered text can change the meaning of the picture and – from a human perspective – constitutes the multimodal aspect of the problem. But an ML system, tasked to extract meaning from both the rendered text and the depiction of the desert, has to start from raw pixels in both cases.}\]
ity scores between image-text pairs. Some of the images in the training data also show characters, words or sentences, and because CLIP was trained with these images and their corresponding OCR data, CLIP has learned to correctly identify what the pixels depicting text represent (see Figure 5). CLIP is multimodal because its inputs are an image and text, for which it has to predict the similarity; whether or not the image shows text is irrelevant.

5.2.2 Images of infrared vs. visible light

Infrared images and visible light images are represented uniformly (a stacked grid of photon counts of different energy intervals). Additionally, we can define a bijective mapping between the two, e.g. by adding/subtracting a fixed frequency and thereby shifting the infrared image into a visible light image (as used for night vision). Therefore, by our definition, they represent the same modality. In this way, our definition enables us to define modalities for information not directly perceptible by humans because of sensory limitations. For example, the photons of infrared light do not carry the correct amount of energy to be perceptible by our eyes, as it is the case for most of the electromagnetic spectrum.

5.2.3 Language

Finally, our definition captures two key traits of the multimodal nature of language: (i) coming in many forms (speech, handwriting, signed language, ASCII-code), language constantly switches representations and media. In cases where (ii) after pre-processing different language representations cannot be converted one to the other without losing task-relevant information (e.g., intonation, hesitation, modulation in speech), they become multiple modalities, like speech–text, or handwriting–text, etc.

With our definition, languages like English and Japanese are considered to be unimodal if after pre-processing both are represented in Unicode. If not, handling them both becomes a multimodal task. This behavior relates the essence of multimodal ML and multilingual NLP, in terms of their complementarity: There are concepts in some languages that cannot be efficiently translated to other languages; much like humans cannot conceive how a bee sees ultra-violet light (Chittka and Wells, 2004).

Figure 5: In this example, Goh et al. (2021) show that CLIP can detect a high similarity between the image containing a handwritten note reading “PIZZA” and the textual input “pizza”, i.e. CLIP has learned to read from images like humans can do.

5.3 Relevant aspects of multimodality

Similarly to previous definitions of multimodality, we focus on the representation and content of inputs (or output, for that matter), but we consider them in relation to the task (see Figure 2):

(i) Starting from the task, we capture what information is put to use and relevant for solving it. Hence, despite the presence of e.g. video and language in a dataset, the task may be such that one of them is not relevant and could be fully ignored.

(ii) We focus on the representation of the information that forms the direct input to the ML system after pre-processing. Unlike previous definitions, we reduce the importance of the medium in multimodal ML. The medium may change without modifying the nature of the information that is represented, used and needed for the task.

(iii) Finally and crucially, we determine the uni- vs. multi-modality of the information joined to solve a task, by analyzing its lack of bijectivity, i.e. its complementarity.

Hereby we conclude that it is only in the context of a task and data used to solve it that we can analyze the issue of information relevance and bijectivity/complementarity, i.e. of multimodality. In this sense we argue that there is no multimodal input, output, or data per se; it is only through a task that requires multimodal information sources to be solved that the corresponding inputs or outputs can be truly considered multimodal.

6 Conclusion

In this paper we have shown how human-centered definitions of multimodality are too restrictive for current and future developments of ML. Also,
machine-centered definitions focusing on representations only do not capture the crucial trait of multimodal machine learning. We instead propose a new definition of multimodality that focuses on the relation between representations, information and the given task, and that – through the novel dimension of the task – enables us to make much sharper distinctions compared to current standards, while covering a much wider spectrum of multimodal data.

With this position paper, we (a) offer a working definition on how to use the term multimodality, (b) aim to raise awareness that defining multimodality is harder than expected, and (c) invite the community to discuss these challenges and (why not?) to provide a better definition.

7 Ethical considerations

In the present paper, we portray the opinion that the human experience of the world should not be normative for defining the multimodal characteristic of a machine learning system. Instead, we claim that machine learning should draw inspiration from human biology and psychology, but not limit itself by imitation. Independently of (a) where the design of machine learning systems is inspired from, and (b) their capabilities, which may extend or exceed those of humans, we strongly believe that machine learning should be done for humans’ service, following the ethical considerations developed and accepted by human society.

Acknowledgements

We thank our anonymous reviewers for their insightful questions and comments. This theoretical research and question about how to define multimodality was inspired by fruitful discussions in the Copenhagen meeting of the Multi3Generation COST Action CA18231.

References


Danica Jovanovic. 2015. Key terms in multimodality: Definitions, issues, discussions: Entries on: Important people in multimodality theory; information value; connotation/denotation; perspective.


Jiquan Ngiam, Aditya Khosla, Mingyu Kim, Juhan Nam, Honglak Lee, and Andrew Y Ng. 2011. Multimodal deep learning. In ICML.


Are Gestures Worth a Thousand Words?
An Analysis of Interviews in the Political Domain

Daniela Trotta
Università degli Studi di Salerno
Via Giovanni Paolo II 132,
Fisciano, Italy
dtrotta@unisa.it

Sara Tonelli
Fondazione Bruno Kessler
Via Sommarive 18
Trento, Italy
satonelli@fbk.eu

Abstract

Speaker gestures are semantically co-expressive with speech and serve different pragmatic functions to accompany oral modality. Therefore, gestures are an inseparable part of the language system: they may add clarity to discourse, can be employed to facilitate lexical retrieval and retain a turn in conversations, assist in verbalizing semantic content and facilitate speakers in coming up with the words they intend to say. This aspect is particularly relevant in political discourse, where speakers try to apply communication strategies that are both clear and persuasive using verbal and non-verbal cues.

In this paper we investigate the co-speech gestures of several Italian politicians during face-to-face interviews using a multimodal linguistic approach. We first enrich an existing corpus with a novel annotation layer capturing the function of hand movements. Then, we perform an analysis of the corpus, focusing in particular on the relationship between hand movements and other information layers such as the political party or non-lexical and semi-lexical tags. We observe that the recorded differences pertain more to single politicians than to the party they belong to, and that hand movements tend to occur frequently with semi-lexical phenomena, supporting the lexical retrieval hypothesis.

1 Introduction

A bodily gesture is a visible action of any body part, when it is used as an utterance, or as part of an utterance (Kendon, 2004). If such actions are produced while speaking, we can talk about co-speech gestures. Their occurrence, simultaneous or coconcomitant to speech, has led to different views regarding their role in communication (Wagner et al., 2014).

Some authors (McNeill, 2005; Kendon, 2004) have considered gestures as an integrative, inseparable part of the language system. Indeed gestures may provide important information or significance to the accompanying speech and add clarity to the children’s narrative (Colletta et al., 2015); they can be employed to facilitate lexical retrieval and retain a turn in conversations (Hostetter et al., 2007). From this point of view, gestures facilitate speakers in coming up with the words they intend to say by sustaining the activation of a target word’s semantic feature, long enough for the process of word production to take place (Morsella and Krauss, 2004).

Gestures can also convey semantic meanings. For example, Müller et al. (2013) discuss the principles of meaning creation and the simultaneous and linear structures of gesture forms. In this framework, they propose individual aspects of a “grammar” of gestures and conclude that in gestures we can find the seeds of language or the embodied potential of hand-movements for developing linguistic structures. As pointed out by Lin (2017) the link between speech and gesture can be explained by two gesture-speech characteristics: semantic coherence, i.e. combining gesture with meaningful and related speech, and temporal synchrony, i.e. producing gesture in synchrony with speech (Butcher, 2000). The role of synchronization is particularly relevant for the creation of multimodal resources (Allwood, 2008), because it allows researchers to overcome one of the historical limits of traditional corpora that are in one modality (either written or spoken): presenting data in a single format offers limited opportunities for exploring non-verbal, gestural features of discourse, while they are important aspects to understand intercultural face-to-face interaction (Adolphs and Carter, 2013; Knight, 2011).

Nevertheless, Beattie and Shovelton (1999) have shown that most of the time gestures are produced before the linguistic item to which they are related, defining this phenomenon “temporal asynchrony”.

Proceedings of the 1st Workshop on Multimodal Semantic Representations (MMSR), pages 11–20
June 16, 2021. ©2021 Association for Computational Linguistics
Also Butterworth and Beattie (1978) presented some empirical evidence to prove that temporal asynchrony between gestures and speech was more common in spontaneous speech and that hand gestures were associated with low-frequency unpredictable lexical items, i.e., lexical items that were more difficult for speakers to reach in the course of language production (Goldman-Eisler, 1958; Beattie and Butterworth, 1979). Their conclusion was the following: “Gestures are products of lexical preplanning processes, and seem to indicate that the speaker knows in advance the semantic specification of the words he will utter, and in some cases has to delay if he has to search for a relatively unavailable item” (Butterworth and Beattie, 1978, p. 358).

Research on spoken interaction has suggested that non-verbal communication is currently the least understood and analyzed aspect of communication, despite recognizing its equal importance (Knight, 2011; McNeill, 2016). For this reason, we believe it is very important to carry out studies on gesture-talk interaction and develop multimodal corpora.

Our study focuses in particular on the relationships between the co-occurrence of speech and gesture in Italian in the specific case of political interviews since:

i) television interviews are inherently multimodal and multisemiotic texts, in which meaning is created through the co-presence of visual elements, verbal language, gestures, and other semiotic cues (Vignozzi, 2019);

ii) linguistic studies in the political domain can be of interest also beyond the NLP community, for example in political science and communication studies, and

iii) the Italian political scene has been little studied from this perspective.

In particular, in the following sections we address research questions such as:

1. Are there semantic patterns of gesture-speech relationship?

2. Does political party affiliation influence this relationship?

3. Does the presence of gesturing indicate problems with the retrieval of words during speech?

Our examination of the co-occurrence of speech and gesture will shed light into how the two communication models interact. We also release the corpus of political interviews with the new annotation layer encoding the functions of hand movements at https://github.com/dhfbk/InMezzoraDataset.

2 Political and multimodal corpora in Italian

In recent years, political language has received increasing attention, especially in English, since it is possible to have free access to speech transcriptions from UK and US government portals and personal foundation websites such as the White House portal, William J. Clinton Foundation, Margaret Thatcher Foundation. This has fostered research on political and media communication and persuasion strategies (Guerini et al., 2010; Esposito et al., 2015). As for Italian, which is the language of interest for this study, only few corpora in the political domain are available.

One of the first experiments was the CorpusB (Bolasco et al., 2006) composed by 111 speeches by the former Prime Minister Silvio Berlusconi – for a total of 325,000 tokens – created to study the evolution of Berlusconi’s political language from the moment when he started his political career in 1994 until the last programmatic speech of his third government in 2006. Subsequently, the work of Salvati and Pettorino (2010) analyzed diachronically some of the suprasegmental aspects of Berlusconi’s speeches from 1994 to 2010, including an analysis of the length of logical chains, the number of syllables per chain, the maximum and minimum pitch and frequency of speech, average duration of empty pauses, fluency and tonal range.

Among the most recent corpora made available in Italian, the largest one includes around 3,000 public documents by Alcide De Gasperi (Tonelli et al., 2019) that has been mainly used to study the evolution of political language over time (Menini et al., 2020). All the corpora cited above are monomodal and none of them takes into account gestural traits. Indeed, corpora that include only one modality have a long tradition in the history of linguistics. According to Lin (2017, p. 157) “the construction and use of multimodal corpora is still in its relative infancy. Despite this, work using multimodal corpora has already proven invaluable for answering a variety of linguistic research questions that are otherwise difficult to consider”.

This is also confirmed by the fact that – to date –
exist 286 multimodal resources certified for all languages by the LRE map but only one is in Italian, i.e. IMAGACT a corpus-based ontology of action concepts, derived from English and Italian spontaneous speech (Moneglia et al., 2014; Bartolini et al., 2014). So both from the political and the multimodal point of view, this language is not well represented.

In an attempt to fill this gap, we first developed the PoliModal corpus (Trotta et al., 2019, 2020), containing the transcripts of 56 TV face-to-face interviews of 14 hours, taken from the Italian political talk show “In mezz’ora in più” broadcast between 2017 and 2018, for a total of 100,870 tokens. The annotation has been done using XML as markup language and following the TEI standard for Speech Transcripts in terms of hesitation phenomena:

(a) **Pause**: this tag is used to mark a pause either between or within utterances;

(b) **Semi-Lexical**: this tag is used to label interjections (i.e. ‘eh’, ‘ehm’ etc.), or more generally words that convey the meaning of an entire sentence, constituting a complete linguistic act demonstrated by their paraphrasability;

(c) **FalseStart**: this tag shows the speaker’s abandonment of an already produced word or sequence of words, with or without repetition of previously used linguistic material;

(d) **Repetition**: with this tag are marked cases of repetition of words or portions of sentences in order to give coherence and cohesion to the speech or self-repetition as a control mechanism of the speech programming;

(e) **Truncation**: truncation indicates the deletion of a phoneme or a syllable in the final part of a word.

The corpus includes also the annotation of facial displays, hand gesture and body posture, which we carried out using the MUMIN coding scheme annotation (Allwood et al., 2007) and ANVIL (Kipp, 2001) a tool for the automatic annotation of audiovisual material containing multimodal dialogue. This corpus is considered as a starting point for our study.

3 **Annotation of Hand Movements**

Starting from PoliModal corpus described in Section 2, we manually add a new level of annotation that takes into account the semantic functions covered by one of the gestures already tagged in the corpus: hand movements. This is because the gestural movements of the hands and arms, i.e. spontaneous communicative movements that accompany speech (McNeill, 2005), are probably the most studied co-speech gestures (Wagner et al., 2014). Based on the seminal works by Kendon (1972, 1980) about the relationship between body motion and speech on the one hand, and about gesticulation and speech in the process of utterance on the other, they are usually separated into several gestural phases: initial/rest position, preparation, (pre-stroke) hold, stroke, (post-stroke) hold, retraction/recovery and rest position (Kita, 1990; Kendon, 2004; Bressem and Ladewig, 2011; Ladewig and Bressem, 2013). Note that all gestures are not necessarily constituted by these phases and that some phases may also be duplicated.

In PoliModal the **hand movement trajectory** tag indicates only the start and end of the movement in terms of time and the trajectory of the gesture, in particular **up, down, sideways, complex**. In order to keep track also of the semantic function covered by the tag, we manually added an additional information layer to those already present – following the classification proposed by Lin (2017) adapting Colletta et al. (2015) and Kendon (1972)2 – which attributes five functions to hand movements:

- **Reinforcing**: the information brought by the gesture is equal to the linguistic information it is in relation with. For example, one of the interviewees emphasizes the sacrifices to which Italians have been subjected in the last fifteen years, including “il 3% del rapporto

---

1 A mechanism to monitor the use and creation of language resources by collecting information on both existing and newly-created resources, freely available at http://lremap.elra.info/

2 We specify that one of the first classifications of gestures was proposed by Ekman and Friesen (1969) that classified kinesic behavior into four broad categories: (1) emblems (“are those nonverbal acts which have a direct verbal translation, or dictionary definition, usually consisting of a word or two, or perhaps a phrase” (Ekman and Friesen, 1969, p. 63)), (2) illustrators (“they are movements which are directly tied to speech, serving to illustrate what is being said verbally” (Ekman and Friesen, 1969, p. 68)), (3) affect displays (“can be related to verbal behaviour in a number of ways. They can repeat, qualify or contradict a verbally stated affect, or be a separate, unrelated channel of communication” (Ekman and Friesen, 1969, p. 77)), and (4) regulators (“These are acts which maintain and regulate the back-and-forth nature of speaking and listening between two or more interactants” (Ekman and Friesen, 1969, p. 82)).
deficit/PIL \( \text{(en. the 3\% deficit/PIL ratio)} \). In saying this he makes the sign of the number three with the fingers of his right hand.

- **Integrating**: the information provided by the gesture does not add supplementary information to the verbal message, but makes the abstract concepts more precise. A frequent example in our annotation is when a politician, in order to contrast two items such as left and right parties, points one of his hands toward the right and the other toward the left.

- **Supplementary**: the information brought by gestures adds new information not coded in the linguistic content. For example, in one of the interviews, the interviewee comments on the amount of members of Parliament elected from another party saying “...non so quanti parlamentari porterà in Parlamento” (en. ...I don’t know how many MPs they will bring to Parliament”) and in the meantime he opens his arms as if to imply a large number.

- **Complementary**: the information provided by the gesture brings a necessary complement to the incomplete linguistic information provided by the verbal message. The gesture usually disambiguates the message, for example, in our annotation it is common to find cases where deictic adverbs such as *qui* (en. here) are accompanied by the corresponding pointing gesture.

- **Contradictory**: the information provided by the gesture contradicts the linguistic information provided by the verbal message. This kind of gesture was not found in our annotation.

- **Other**: within this category we include all the gestures that annotators were not able to classify with the above mentioned semantic labels.

Our annotation follows the selection criterion highlighted by Allwood et al. (2007), claiming that annotators are expected to select gestures to be annotated only if they have a communicative function. Following this principle, each annotator looked at the portion of the video in which the hand movements were occurring and depending on the meaning that he/she thought the gesture had in that particular context of utterance, attributed the corresponding semantic function.

However, as Yoshioka (2008) points out gestures can be functionally ambiguous and thus have multiple semantic functions simultaneously. According to Tsui (1994), the source of these multiple functions often lies in the sequential environment of the conversation in which the utterance occurs. To simplify the task, annotators are therefore asked to assign a single semantic function to the gestures under investigation, choosing the function that they consider prevalent in the context of use. When the gesture-speech relationship appears too vague, it is good practice to conduct interviews with speakers to confirm the interpretation of gesture meanings. In fact, as suggested by Kochman et al. (2014), through multiple methods of data analysis, such as triangulation, we can test whether interpretations of the results were consistent and internally coherent.

In our case, since such checking is not possible, we try to ensure a high-quality and consistent annotation by computing inter-annotator agreement. Specifically we perform a double annotation of the semantic functions listed above on three of the interviews considered (Matteo Renzi, Luigi Di Maio, Matteo Salvini) for a total of about 2 hours of interviews. Both annotators (one male and one female) are expert linguists. Macro-averaged F1 computed on exact matches amounts to 0.83, which corresponds to an almost perfect agreement. This result confirms that the task is well-defined and that the corresponding annotation guidelines are clear.

Figure 1 shows an example annotation with the new information layer specified with the tag ‘semantic function’. For each observed gesture, the PoliModal corpus already contained: i) the start and end point in the video in terms of milliseconds; ii) the type of gesture observed; iii) the movement trajectory. We add to this the semantic function covered by the gesture in the context.

4 Description of gesture-speech annotation

A summary of the hand gesture annotations in the corpus is reported in Table 1 and 2. In the first one, the number of annotated tags is reported for each politician, while in the second table the values are aggregated by political party. The parties include PD (left-center), Five-Star Movement (center-populist), Lega (right-populist), Casa Pound (right), Popolo delle Libertà (center-right). The ‘Contradic-
<table>
<thead>
<tr>
<th>Interviewee</th>
<th>Integrat.</th>
<th>Reinforc.</th>
<th>Supplement.</th>
<th>Complement.</th>
<th>Other</th>
</tr>
</thead>
<tbody>
<tr>
<td>Matteo Renzi</td>
<td>32</td>
<td>9</td>
<td>2</td>
<td>23</td>
<td>1</td>
</tr>
<tr>
<td>Luigi Di Maio</td>
<td>6</td>
<td>0</td>
<td>1</td>
<td>9</td>
<td>1</td>
</tr>
<tr>
<td>Matteo Salvini1</td>
<td>16</td>
<td>6</td>
<td>3</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>Matteo Salvini2</td>
<td>17</td>
<td>10</td>
<td>0</td>
<td>14</td>
<td>5</td>
</tr>
<tr>
<td>Walter Veltroni</td>
<td>8</td>
<td>3</td>
<td>0</td>
<td>8</td>
<td>4</td>
</tr>
<tr>
<td>Simone Di Stefano</td>
<td>5</td>
<td>0</td>
<td>2</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>Pierluigi Bersani</td>
<td>13</td>
<td>4</td>
<td>0</td>
<td>12</td>
<td>2</td>
</tr>
<tr>
<td>Angelino Alfano</td>
<td>21</td>
<td>11</td>
<td>1</td>
<td>16</td>
<td>8</td>
</tr>
<tr>
<td>Giulio Tremonti</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Matteo Orfini</td>
<td>7</td>
<td>0</td>
<td>0</td>
<td>10</td>
<td>3</td>
</tr>
<tr>
<td>Pier Carlo Padoan</td>
<td>16</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>15</td>
</tr>
<tr>
<td>Carlo Calenda</td>
<td>41</td>
<td>1</td>
<td>0</td>
<td>35</td>
<td>26</td>
</tr>
<tr>
<td>Alessandro Di Battista</td>
<td>29</td>
<td>1</td>
<td>0</td>
<td>20</td>
<td>0</td>
</tr>
<tr>
<td>Total</td>
<td>214</td>
<td>46</td>
<td>10</td>
<td>159</td>
<td>66</td>
</tr>
</tbody>
</table>

Table 1: Frequency of the type of gestures annotated for each interviewee.

<table>
<thead>
<tr>
<th>Political party</th>
<th>Integrat.</th>
<th>Reinforc.</th>
<th>Supplement.</th>
<th>Complement.</th>
<th>Other</th>
</tr>
</thead>
<tbody>
<tr>
<td>PD</td>
<td>117</td>
<td>17</td>
<td>2</td>
<td>91</td>
<td>51</td>
</tr>
<tr>
<td>M5S</td>
<td>35</td>
<td>1</td>
<td>1</td>
<td>29</td>
<td>1</td>
</tr>
<tr>
<td>Lega</td>
<td>36</td>
<td>17</td>
<td>4</td>
<td>20</td>
<td>6</td>
</tr>
<tr>
<td>Casa Pound</td>
<td>5</td>
<td>0</td>
<td>2</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>Il Popolo delle Libertà</td>
<td>21</td>
<td>11</td>
<td>1</td>
<td>16</td>
<td>8</td>
</tr>
<tr>
<td>Total</td>
<td>214</td>
<td>46</td>
<td>10</td>
<td>159</td>
<td>66</td>
</tr>
</tbody>
</table>

Table 2: Frequency of the type of gestures for each political party

Figure 1: Example of the new annotation level in xml

Over the years, studies have shown that the production of gestures is influenced by the syntax of the language itself and by the socio-cultural context of the language. As explained in a 2015 study by (Colletta et al., 2015) – focused on co-speech gesture production in children’s narratives – language syntax influences gesture production. For example – as known – some languages require an explicit subject (i.e. English, French, etc.), whereas others (i.e. Italian, Spanish, etc.) are null-subject languages. This characteristic requires distinct marking of referential continuity in the textual use of language, with less need to repeat anaphora in the latter case (Hickmann, 2002). Another key factor influencing the communication is culture as a set of
values and norms that help shape the social behavior of individuals who belong to a cultural group as well as social interaction between them. Very well known is the study in (Kendon, 2004), showing that Italians use a great number of gestures when communicating. So – since some socio-cultural factors seem to influence the production of gestures – we wonder whether political party affiliation is among them. Then as a next step, we investigate whether there is a significant relationship between the political party of affiliation of interviewees in the dataset and the type of gestures used, and thus whether the political party of affiliation affects the choice of gesture categories. The political parties considered are the ones reported in Table 2. We compute a one-way ANOVA with independent samples. The results obtained suggest that the null hypothesis cannot be rejected since the significance value obtained is \( p = 0.11 \). Therefore we can conclude that the party affiliation does not play any role in the combination of gestures and speech.

A qualitative analysis of the single interviews shows interesting differences in attitude and communication style, which pertain to single politicians rather than to party positions. Matteo Renzi, for example, uses gestures very frequently to accompany his speech. We report an example of ‘Integration’ below:

Matteo Renzi: “Quello che sta accadendo invece in queste settimane, in questi mesi, conferma che c’è una grande distanza tra la politica dei palazzi e la politica della quotidianità [integrating].”

(Eng. “Instead what is happening in these weeks, in these months, confirms that there is a great distance between the politics of the Palaces and the politics of everyday life.”)

Renzi underlines that the distance between politics made by elites, detached from the real problems of the country (“politics of the Palaces”), and “politics of everyday life”, that is, attentive to reality and to citizens, is increasingly evident. A gesture is used to stress this difference: the speaker’s open right hand points away from his torso to the screen where the photo refers to the photo by pointing with his left hand

Concerning the Reinforcing type of gesture-speech relationship, it is mainly used to reiterate a concept already expressed linguistically, and it is not very used, probably because it may seem redundant. Angelino Alfano turns out to be the interviewee who makes most use of this type of gesture. In this example, Alfano, talking about the consensus obtained by one of his political opponent Matteo Salvini, claims that this consensus was obtained at his expense. So, in saying “contro di me” (against me), the open hands are close to his bust.

Angelino Alfano: “Quindi la sfida di Salvini, avendo aggregato consenso – contro di me per altro [reinforcing] – sull’immigrazione, è incanalarlo su un regime di legislazione democratica.”

(Eng. “So Salvini’s challenge, by aggregating consensus – against me by the way – on immigration, is to channel it on a regime of democratic legislation.”)

As mentioned above, Supplementary gestures are used with a very low frequency. One of the few examples in the corpus is present in Simone di Stefano’s interview, where he is asked to clarify the alleged relations of the party with a convicted member of the Mafia. The interviewee tries to provide an explanation, but the interviewer continues to put him under pressure. At this point the interviewee lowers his gaze and moves his open right hand away from his torso while saying “but I don’t want to avoid [your question]”, as if to implicitly ask the journalist to stop her suppositions and let him explain his position.

Supplementary gestures bring a necessary complement to the incomplete linguistic information provided by the verbal message. They are frequently used by the respondents in the corpus under analysis, in most cases to disambiguate the message or simply some linguistic elements. This indicates the speaker’s intention to be as clear as possible. For example, at the beginning of the interview with Carlo Calenda, he is shown a photo that portrays him wearing a worker’s helmet. The interviewee refers to the photo by pointing with his left hand away from his torso to the screen where the photo is displayed, making it easier for viewers to understand what he was referring to:

Carlo Calenda: “Benché gli operai non si sentiranno, come posso dire, contenti dopo aver visto la mia foto con quel caschetto [complementary]
(Eng. “Although the workers won’t feel, how can I say, happy after seeing the picture of me in that helmet where I looked like a total idiot.”)

As noted above, a residual category has been added to the tags. The Other category includes all the gestures that annotators were not able to classify with the above mentioned semantic labels. This problem was found most frequently in the interviews with Pier Carlo Padoan and Carlo Calenda. These gestures are different from the others because they show a batonic value, that is, they are used to mark the rhythm of the enunciation, for example by tapping a finger on the table.

5 Lexical Retrieval hypothesis

Many studies have suggested that gestures, especially representational gestures (Krauss and Hadar, 1999) play a direct role in speech production by priming the lexical retrieval of words. This view has been termed the Lexical Retrieval hypothesis.

The hypothesis is based on research arguing that (1) gesturing occurs during hesitation pauses or in pauses before words indicating problems with lexical retrieval (Dittmann and Llewellyn, 1969; Butterworth and Beattie, 1978), and (2) that the inability to gesture can cause verbal disfluencies (Dobrogaev, 1929). In addition – as (Krauss, 1998) pointed out – speakers were more dysfluent overall in constrained-speech conditions than in natural conditions. Since the corpus used as the object of study presents a level of annotation that takes into account some hesitation pauses and verbal disfluencies, we decided to verify this hypothesis in the political domain, where speakers usually have to control well their communication and be persuasive.

We compute weighted mutual information (Guiasu, 1977) between hand movements and each of the speech disfluencies reported in Table 3. This measure is calculated to show existing mutual dependencies between co-occurring tags. We consider only the interviews in the PoliModal corpus that have a minimal length of 50 turns, so to have a good amount of annotations to consider. We report in Table 3 the tag incidence per 100 turns for each interview considered.

Among the politicians included in this dataset, the one that most accompanies his speech with hand gestures is Matteo Salvini (Lega) considering both interviews, followed by Carlo Calenda (PD) and Angelino Alfano (Il Popolo della Libertà). Their belonging to different political parties suggests that the use of hand movements is more an individual trait than a feature characterising specific political positions.

Weighted mutual information (WMI) is computed between hand movements and tags reported in Table 3. The values obtained are shown in the heatmap reported in Figure 2, with lighter colors corresponding to higher WMI values.

Figure 2: WMI values between hand movements and tags reported on the x-axis for each interviewee on the y-axis.

Overall, hand movements tend to have a higher association with semi-lexical traits and pauses, which would confirm the assumptions of Lexical Retrieval hypothesis according to which gesturing occurs during hesitation pauses or in pauses before words indicating problems with lexical retrieval (Dittmann and Llewellyn, 1969; Butterworth and Beattie, 1978).

This effect is however not present for some politicians, such as Di Battista and Alfano, while it is evident for some others such as Bersani and Salvini. Therefore, our findings are not generally applicable to all interviewees in our corpus. Fig. 2 shows also...
evident differences in gesturing behaviour among the considered politicians. For instance, although Carlo Calenda and Angelino Alfano present a high incidence of hand movements, they do not seem to be associated with specific tags. Matteo Renzi, instead, shows a gesturing behaviour that is unique compared to all the other interviewees, with hand gestures that are almost always used in association with other speech phenomena.

In the interviews, we observe also the presence of negative values for WMI obtained in relation to false-starts (-0.11), repetitions (-0.1 and -0.6) and truncations (-0.8), suggesting that hand movements are less likely to be accompanied by such linguistic phenomena.

### 6 Conclusions

In this work, we investigate co-speech gestures of several Italian politicians during face-to-face interviews. To this purpose, we enrich an existing corpus with labels describing the semantic type of the different hand movements. Concerning gesture-speech relationship, the results obtained suggest that hand movements are mainly used with an integrative and complementary functions. So, the information provided by such gestures adds precision and emphasis to spoken information. We also show that party affiliation does not significantly influence the gesture-speech relationship. Finally we test the Lexical Retrieval Hypothesis by computing the association between hand movements produced by each interviewee and speech disfluencies using weighted mutual information. Results show that hand movements tend to co-occur with full pauses (i.e. repetition) and empty pauses (i.e. pause) and more frequently with interjections (i.e. semi-lexical), suggesting that gesticulating may represent an attempt at lexical retrieval.

In the future we plan to conduct further analyses aimed at understanding whether such gestures co-occur with specific types of words (e.g. copulative verbs, predicative verbs, etc.) and whether other linguistic or socio-linguistic variables such as language complexity or age influence the use of hand movements and their semantic functions.

### References


Geoffrey Beattie and Heather Shovelton. 1999. Do iconic hand gestures really contribute anything to the


Daniela Trotta, Sara Tonelli, Alessio Palmero Aprosio, and Elia Annibale. 2019. Annotation and analysis of the polimodal corpus of political interviews. In Sixth Italian Conference on Computational Linguistics (CLiC-it 2019).


Requesting clarifications with speech and gestures

Jonathan Ginzburg
Université de Paris
Laboratoire de Linguistique Formelle
Institut Universitaire de France
yonatan.ginzburg@u-paris.fr

Andy Lücking
Université de Paris
Laboratoire de Linguistique Formelle
Goethe-Universität Frankfurt
luecking@em.uni-frankfurt.de

Abstract
In multimodal natural language interaction both speech and non-speech gestures are involved in the basic mechanism of grounding and repair. We discuss a couple of multimodal clarification requests and argue that gestures, as well as speech expressions, underlie comparable parallelism constraints. In order to make this precise, we slightly extend the formal dialogue framework KoS to cover also gestural counterparts of verbal locutionary propositions.

1 Introduction
Detailed taxonomies of verbal Clarification Requests (CRs) already exist (Purver et al., 2003; Rodriguez and Schlangen, 2004) and accounting for these motivate theories of grounding and clarification interaction such as (Schlangen, 2004; Purver, 2006; Ginzburg, 2012), which provide wide coverage thereof. Although there exists some corpus-based and experimental work on multimodal repair (Healey et al., 2015; Seo and Koshik, 2010; Hough et al., 2015), detailed taxonomies are yet to be developed, nor formal accounts thereof.

In this paper we consider how to account for the multimodal versions of one of the commonest types of clarification request dubbed reprise fragments by Purver et al. (2003). Clarification requests play an important role in semantic methodology (Purver and Ginzburg, 2004) and in the construction of dialogue systems (Purver et al., 2011). Ginzburg and Cooper (2004) argue in detail that these exhibit significant syntactic and phonological parallelism with their source, as exemplified in (1a); concretely their claim is that the intended content reading (‘what do you mean by …’) requires segmental identity with the source. A similar condition mutatis mutandis seems to be the case for gestural ones (2): (2a,b) involve clarifying a body movement (the former from example (1), Healey et al., 2014, 26, emphasis added), the latter two concern laughter, either with respect to content or in the latter case clarifying the manner of laughter ((2e) is from Fig. 1 of Healey et al., 2014, 26):

(1) a. (i) A: Do you fear him? B: Fear? (= What do you mean by ‘fear’ or Are you asking if I fear him) / #Afraid? (ii) A: Are you afraid of him? B: Afraid? (= What do you mean by “afraid”? or Are you asking if I am afraid of him) / #Fear?

b. A: Are you afraid of him? B: Afraid? (= What do you mean by “afraid”?)

(2) a. B: You have to move your legs like this [moves right hand up and down in a wave-like manner]. A: [moves right hand up and down in a wave-like manner, raises eye-brows]

b. … and that movement really cracks your back

c. What’s that? You do that and someone pulls?

d. A: I hear you’re busy ⟨laughter ⟩ [= little giggle]. B: ⟨laughter ⟩ ? (= low arousal laughter with rising contour). (attested example)

e. Was it kind of like [H:o?] =
[H:hhh]

Clarification requests also occur on larger time scales, as is evinced in Figs. 1 to 3. The example is taken from the Speech and Gesture Alignment

1We use the letters ‘A’ and ‘B’ to denote the participants. Paraphrases of reprise fragments are introduced by an equation symbol, emphasis is indicated by italics, impossible or infelicitous clarifications are marked by ‘#’.
corpus SaGA (Lücking et al., 2010), which is a multimodal corpus of route direction dialogues. The example is about a section of a route where the route follower has to enter a park and walk around a pond, but not completely, just to three quarters. The route section is described by the route giver in Fig. 1. It is put to clarification by the addressee (route follower) in Fig. 2. Abstracting over perspective, the moving around movement is more or less kept constant, but modelling the pond is changed from a gesture hold to a drawing on the back of the hand. The route giver subsequently corrects the clarification by a path drawing on the addressee’s back of hand in Fig. 3.

We show how to extend existing notions of conversational context and representation of speech multimodally to account for such cases. The basic extensions to the formal framework introduced in the following section are (i) multi-tier partiturs for capturing signals on different channels, (ii) a classification of gesture events on the tiers, and (iii) an anaphoric multimodal clarification rule requesting feedback concerning a previous multimodal fragment.

2 Background

Our account is formulated within Type Theory with Records (TTR, Cooper, 2005; Cooper and Ginzburg, 2015). TTR is a formal semantics framework based in the proof-theoretic, intuitionistic mathematics of Martin-Löf (1984). The reason for using a formal framework is that it enables researchers working on semantic phenomena in a scientific, precise manner. This is possible since the interpretation of types and structures used can be fixed in models—for such a denotational interpretation of TTR see Cooper (2021). Although traditionally mainly applied to the compositional semantics of sentences, semanticists working on dialogue soon developed conversation-oriented extensions (just think of the content of particles such as Hi! or Yes or highly normative patterns such as question–answering.) However, classifying (multimodal) natural language utterances is not always a binary affair (think, e.g., of vagueness). To this end, there are probabilistic interpretations of TTR (Cooper et al., 2015). Although we could render our discussion in probabilistic terms, we refrain from doing so since this paper is not concerned with probabilistic phenomena as such and this keeps representations simpler. TTR integrates logical techniques such as the lambda calculus and the expressiveness of feature-structure like objects (namely records and record types). A typing judgement $a : T$ is true iff object $a$ is of type $T$. Types constructed from $n$-ary predicates ($n > 0$) are dependent on the values assigned to the labels that appear as arguments. Thus, if $a_1 : T_1$, $a_2 : T_2(a_1)$, ..., $a_n : T(a_1,a_2,\ldots,a_{n-1})$, then the record on the left in (3) is of the record type on the right in (3):

$$
\begin{align*}
[l_1 = a_1] & \quad [l_2 : T_1] \\
\vdots & \quad \vdots \\
[l_n = a_n] & \quad [l_n : T(l_1,l_2,\ldots,l_{n-1})]
\end{align*}
$$

The notation $[l = a : T]$ represents a manifest field (Coquand et al., 2003). It is a notational convention for a singleton type $T_a$, where for any $b, b : T_a$ iff $b = a$.

Merge types correspond to unification in feature-structure formalisms. A merge ‘&’ is exemplified in (4):

$$
\begin{align*}
a. \quad A &= \begin{bmatrix} l_1 : T_1 \\ l_2 : T_2(l_1) \end{bmatrix} \\
b. \quad A \wedge B &= \begin{bmatrix} l_1 : T_1 \\ l_2 : T_2(l_1) \\ l_3 : T_3 \end{bmatrix}
\end{align*}
$$

Drawing on work of Fernando (2007, 2011), TTR comes with a string theory of events. For three events $e_1$, $e_2$ and $e_3$, the string $e_1 e_2 e_3$ represents a course of events, namely the succession of $e_1$, $e_2$ and $e_3$, in that order. The notation $e_1 e_2 e_3$ is an abbreviation for a time-indexed record:

$$
\begin{bmatrix}
t_0 = e_1 \\ t_1 = e_2 \\ t_3 = e_2
\end{bmatrix}, \text{ where time indices } t_i \text{ are in } \mathbb{N}.
$$

If $e_1 : T_1$, $e_2 : T_2$ and $e_3 : T_3$, then $e_1 e_2 e_3 : T_1 T_2 T_3$—the type constructor ‘&’ builds string types out of types. In order to exploit feature structure expressiveness in string types, a string of record types can be build by the same means, but is notationally enclosed in brackets.

---

2The semantic status of natural language processing (NLP) remains unclear, to say the least, as recently pointed out by Bender and Koller (2020). However, theoretical work such as the one developed here can of course trigger NLP applications, where, e.g., theoretically derived labels are automatically annotated on large(r) scale data.

3With some repercussions for some versions on its own (Larsson, 2020).
Figure 1: ‘Du fährst ’um den Teich herum’ ([You drive] around the pond): Index finger and thumb of left hand form a circle and right hand with stretched index finger is moved to three quarters around left hand.

Figure 2: ‘Hier ist der Teich [Frame 1]. Ich komm’ auf den zu [Frames 2–3]. Und was heißt “rechts ab”? [Frame 4]’ (Here is the pond [Frame 1]. I approach it [Frames 2–3]. And what do you mean ‘turn right?’ [Frame 4]): A circular index finger drawing gesture indicates the pond [Frame 1]. The index finger is first moved towards and then around the virtual pond [Frames 2–3]. A straight movement towards the wrist indicates turning right [Frame 4].

Figure 3: ‘Du fährst noch weiter rum.’ (You drive around even more.): Stretched index finger is moved around the virtual pond.
Making use of TTR, the simplest model of context, going back to Montague (1974) is one which specifies the existence of a speaker, addressing an addressee at a particular time. This can be captured in terms of the type in (6).

\[
\begin{align*}
\text{spkr} &: \text{Ind} \\
\text{addr} &: \text{Ind} \\
\text{u-time} &: \text{Time} \\
\text{c-utt} &: \text{addr(spkr,addr,u-time)}
\end{align*}
\]

However, over recent decades it has become clearer how much more pervasive reference to context in interaction is. The visual situation is a key component in interaction from birth (see Tomasello, 1999, Chap. 3). Expectations due to illocutionary acts—one act (querying, assertion, greeting) giving rise to anticipation of an appropriate response (answer, acceptance, counter-greeting), also known as adjacency pairs (Schegloff, 2007). Extended interaction gives rise to shared assumptions or presuppositions (Stalnaker, 1978), whereas epistemic differences that remain to be resolved across participants—questions under discussion are a key notion in explaining coherence and various anaphoric processes (Ginzburg, 2012; Roberts, 1996). These considerations among several additional significant ones lead to positing a significantly richer structure to represent each participant’s view of publicized context, the dialogue game-board (DGB), whose basic make up is given in (7), following the recent version of the dialogue semantic framework called KoS including mood described by Ginzburg et al. (2020b):

\[
\begin{align*}
\text{DGBType} & := \\
\text{spkr} &: \text{Ind} \\
\text{addr} &: \text{Ind} \\
\text{utt-time} &: \text{Time} \\
\text{c-utt} &: \text{addressing(spkr,addr,utt-time)} \\
\text{facts} &: \text{Set(Prop)} \\
\text{vis-sit} &: \text{[foa: Ind ∨ Sir]: RecType} \\
\text{pending} &: \text{List(LocProp)} \\
\text{moves} &: \text{List(IllocProp)} \\
\text{qud} &: \text{poset(} \text{Question}) \\
\text{mood} &: \text{Evaluated}
\end{align*}
\]

Here facts represents the shared assumptions of the interlocutors—identified with a set of propositions. Vis-sit represents the visual situation of an agent, including his or her focus of attention (foa), which can be an object (Ind), or a situation or event (Sit). The remaining fields concern locutionary and illocutionary interaction: Dialogue moves that are in the process of being grounded or under clarification are the elements of the pending list; already grounded moves are moved to the moves list. Within moves the first element has a special status given its use to capture adjacency pair coherence and it is referred to as LatestMove. The current question under discussion is tracked in the qud field, whose data type is a partially ordered set (poset). Mood tracks public displays of emotion, crucial for inter alia laughter and smiling (Ginzburg et al., 2020b).

The evolution of context in interaction is described in terms of conversational rules, mappings between two cognitive states, the precondition(s) and the effects. Some examples of such rules are given in (8):

(8)a. Ask QUD-incrementation: given a question \(q\) and ASK(A,B,q) being the LatestMove, one can update QUD with \(q\) as MaxQUD.

\[
\begin{align*}
\text{pre} &: q, \text{Question} \\
\text{effects} &: \text{QUD} = \{q, \text{pre.QUD}\}; \text{poset(} \text{Question})
\end{align*}
\]

b. Assert QUD-incrementation: a straightforward analogue for assertion of (8a): given a proposition \(p\) and ASSERT(A,B,p) being the LatestMove, one can update QUD with \(p\) as MaxQUD.

\[
\begin{align*}
\text{pre} &: p, \text{Prop} \\
\text{effects} &: \text{QUD} = \{p, \text{pre.QUD}\}; \text{poset(} \text{Question})
\end{align*}
\]

c. QSPEC: this rule characterizes the contextual background of reactive queries and assertions—if \(q\) is MaxQUD, then subsequent to this either conversational participant may make a move constrained to be \(q\)-specific (i.e., either About or Influencing \(q\)).

\[
\begin{align*}
\text{pre} &: \text{QUD} = \{q, q\}; \text{poset(} \text{Question}) \\
\text{effects} &: \text{LatestMove} = \text{R}(\text{spkr, addr, r}) \land \text{IllocRel} \\
& \quad \land \text{c}1 \land \text{Qspecific(r, q)}
\end{align*}
\]

As emphasized by Clark (1996) and by work in Conversation Analysis (CA; Schegloff et al., 1977) grounding and clarification interaction are
important structuring processes in interaction. In Ginzburg (2012) these are modelled as a process triggered by awareness of an utterance event $u$ and the attempt to instantiate the fields of an utterance type $T_u$ emergent from parsing and resolving $u$. The pair of $u$ and $T_u$ is referred to as *locutionary proposition LocProp*. This is a special kind of (Austrian) proposition—records of type
\[
\begin{cases}
\text{sit} : \text{Rec} \\
\text{sit-type} : \text{RecType}
\end{cases}
\]
(Austin, 1950; Barwise and Etchemendy, 1987)\(^4\) where *sit* is an utterance event and *sit-type* the type of a grammatical sign. This allows *inter alia* access to the individual constituents of an utterance. Purver (2004) and Ginzburg (2012) show how to account for the main classes for instance ‘Whether Bo left’, ‘Who left’, and ‘Which student modulo their domain, the questions involve similar answers:

\[
\text{Clarification question is a possible follow up utterance type.}
\]

Thus, the schema ‘if $u$ is an utterance spoken by $A$ and $u0$ is a constituent of $u$, allow responses that are co-propositional\(^5\) with the clarification question $CQ(u0)$ into QUD’, where ‘$CQ'(u0)$’ is one of the three types of clarification question (repetition, confirmation, intended content) specified with respect to $u0$.

In order to utilize the information state update semantics of KoS for analysing multimodal discourse, we add extra structure to the utterance events by incorporating tiers. Tiers can be likened to different instruments on a musical score: a partitur.\(^6\) We represent partiturs as strings of multimodal communication events, which is a temporally ordered sequence of types. One can think of strings in term of a flip-book: a dynamic event is cut into slices, and each slice is modeled as a record type. Such string types (Fernando, 2007; Cooper, 2021) are notated in round brackets:

\[
\text{partitur} := \left[ \begin{array}{l}
\text{\(e\text{\_}speech\)} : \text{Phon} \\
\text{\(e\text{\_}gesture\)} : \text{Trajectory} \\
\text{\(e\text{\_}gaze\)} = \text{\(\text{vis\text{-}sit} : \text{RecType}\)} \\
\text{\(e\text{\_}head\)} : \text{headMove} \\
\text{\(e\text{\_}face\)} : \text{faceExpr}
\end{array} \right]
\]

The progressive unfolding of sub-events on the various tiers in time gives rise to incremental production and perception. Formally, this is indicated by the Kleene plus ($\oplus$): the string type in (11) classifies events which consists of a sequence of multimodal communication signals. Hence, partiturs provides a formal means for describing cross-tier interaction.

\(\text{on this view, a proposition } p = \left[ \begin{array}{l}
\text{sit} = s \\
\text{sit-type} = T
\end{array} \right] \text{ is true iff}
\]

\(s : T\) —the situation $s$ is of the type $T$.

\(\text{on a descriptive level, partiturs are akin to XML-encoded messages in the Behavior Markup Language (BML; Vilhjálmsdóttir et al., 2007). But while BML is designed to define the generation of multimodal behavior in virtual agents, partiturs provide a platform for compositional multimodal chart parsing.}\)

---

\(4\)On this view, a proposition $p =$ \(\left[ \begin{array}{l}
\text{sit} = s \\
\text{sit-type} = T
\end{array} \right] \text{ is true iff}
\]

\(s : T\) —the situation $s$ is of the type $T$.

\(5\)Here *CoPropositionality* for two questions means that, modulo their domain, the questions involve similar answers: for instance ‘Whether Bo left’, ‘Who left’, and ‘Which student left’ (assuming Bo is a student.) are all co-propositional.

\(6\)On a descriptive level, partiturs are akin to XML-encoded messages in the Behavior Markup Language (BML; Vilhjálmsdóttir et al., 2007). But while BML is designed to define the generation of multimodal behavior in virtual agents, partiturs provide a platform for compositional multimodal chart parsing.
In order to model one sort of multimodal integration we make use of the account of speech-gesture of Lücking (2013), respectively its TTR reformulation (Lücking, 2016). Speech-gesture integration on this account is modelled in terms of a speech-gesture ensemble (Kendon, 2004), where a gesture (G-DTR) from tier e_gesture attaches to a phonetically marked affiliate (AFF; Schegloff, 1984) from speech (S-DTR, tier e_speech). Thus, multimodal integration of this sort is constrained by both temporal alignment and phonetic-kinematic interface (cf. also Alahverdzhieva et al., 2017). Semantic integration is formally governed by a imagistic feature called conceptual vector meaning (“CVM”). CVM draws on abstract motion perception from psychophysics (Johansson, 1973) and can in semantics formally spelled out in terms of vector-based representations of shapes, movements, orientations, or object axes within the vector space algebra of Zwarts (2003). The basic integration scheme is given in (12):

\[
\text{sg-ensemble} = \begin{cases} 
\text{phon}:s\text{-dtr.phon} : \text{Phon} \\
\text{cat}=s\text{-dtr.cat} : \text{SynCat} \\
\text{cont}=g\text{-dtr.traj} \land s\text{-dtr.cont.cvmm} 
\end{cases}
\]

The underlying rationale of (12) is that both a gesture movement and a CVM value are a trajectory that is mathematically described as a sequence of vectors in three dimensions ($\mathbb{R}^3$; or $\mathbb{R}^4$ if the temporal dimension is explicitly built in). Drawing on work in gesture annotation, gestures are represented in terms of their kinematic features, giving rise to a ‘phonetic’ gesture representation. For example, moving the wrist rightwards, back (i.e., towards the body of the gesturer), and leftwards in a rectangular manner (‘line’) —

\[
\text{path} : \text{line} \\
\text{wrist=mr} \quad \text{mb=ml} : \text{Move}
\]

— a cornered, horseshoe-shaped trajectory ‘[]’ is displayed. Via a translation procedure from gesture representations onto vector representations, the abstract trajectory in (13) is obtained (Lücking, 2016).

\[
\begin{align*}
\text{aff} &= \left[ \text{phon} : [\text{accent} : \text{marked}] : \text{sign} \right] \\
\text{traj} &= \left[ \begin{align*}
&\text{pt} : [\text{u} \perp \text{v} \perp \text{w}] \\
&\text{u(0)} \neq \text{w(1)} \\
&\text{sh} : \{\text{rectangular, open}\} 
\end{align*} \right] : \text{Vec}
\end{align*}
\]

Spatial predicates also carry trajectory information as part of their CVM feature. The vector sequence from (13) is part of the lexical entry of the adjective u-shaped (it modifies a nominal, whose content is an individual).

Since the gesture’s trajectory and the adjective’s CVM value are compatible, both can merge into a sg-ensemble.7 Abstracting away from concrete movements to abstract vector representations seem to provide a format that is appropriate for gestural parallelism constraints, as will be discussed in Sec. 4.

An example involving the ‘u-shape’ gesture is used by Lücking and Ginzburg (2020): the house [has a RECtangular] [] shape. The noun phrase the house has a rectangular shape is accompanied by a rectangular shape gesture which temporally overlaps the bracketed portion of speech. This tier-crossing utterance is incrementally processed by a multimodal chart parser (Earley, 1970; Johnston et al., 1997; Ginzburg et al., 2020a; Alahverdzhieva et al., 2017). The string chart in (15) represents the state after having processed the house has and the gesture’s preparation phase. Due to this input, a VP rule (e9) and a gesture integration rule (e10) have been triggered, but are still pending:

\[
\begin{align*}
\text{phon} : \langle \text{u-shaped} \rangle \\
\text{mod} : \left[ \begin{align*}
\text{cat} : \{\text{head} : \text{noun}\} \\
\text{cont} : \{\text{Ind}\} \\
\text{cvmm} &= \left[ \begin{align*}
&\text{pt} : [\text{u} \perp \text{v} \perp \text{w}] \\
&\text{u(0)} \neq \text{w(1)} \\
&\text{sh} : \{\text{rectangular, open}\} 
\end{align*} \right] : \text{Vec}
\end{align*} \right]
\]

\[
\begin{align*}
\text{shape} : &\text{shape}(\text{mod.cat.cont}, \text{cvmm}) \\
\text{cvm} &= \text{RecType}
\end{align*}
\]

---

7The example illustrates the gist of one form of multimodal integration. Much needs to be said, of course, for instance, on timing, affiliation, and more complicated ways of semantic integration—further details can be found in the references provided here.
edges. Such edges are defined in multichart parsers which have been developed exactly for the purpose of processing multimodal input (Johnston et al., 1997; Alahverdzhieva et al., 2017). We note tier-crossing bindings on the level of utterance events (where an utterance comprises speech and gesture) in terms of the reserved label $e_{sync}$—such combined representations are object of at least one class of gestural clarifications.

## 4 Gestural Clarification: the case of reprise fragments

In this section we show how to modify an existing account of speech reprise fragments with minimal additions, though important empirical questions about the unity of this type of clarification request remain.

The analysis proposed by Ginzburg (2012) for this class of reprise fragments involves two components:

1. A construction $utt-ana-ph$ that enables deixis to the repaired constituent under the constraint of segmental phonological parallelism. This is captured by identifying the phonological type of the clarification seeking utterance with that of the repaired constituent $rc.sit-type.phon$; whereas the content is identified with the speech event of the repaired constituent $rc.sit$. This makes crucial use of the fact that locutionary propositions store both type and token information:

$$\text{(17) } \text{utt-ana-ph} = \begin{bmatrix}
\text{dgb-params} : \text{rc} : \text{LocProp} \\
\text{phontype} = \text{rc.sit-type.phon} : \text{Type} \\
\text{phon} : \text{phontype} \\
\text{cat} : \text{syncat} \\
\text{cont} = \text{rc.sit} : \text{Rec}
\end{bmatrix}$$

2. evocation of the clarification question ‘what do you mean by $u$’ accommodated via the clarification context update rule (9).

These two components get reified into a somewhat more general construction $qud-anaph-int-cl$:

$$\text{This construction, which arguably occurs already at the one word stage (Clark and Bernicot, 2008), is needed for other ‘quotative’ utterances such as}$$

- A: Bo is coming. B: Who do you mean ‘Bo’?
- D: I have a Geordie accident. J: ‘accident’ that’s funny.
its content is identified with max-qud, whereas its sole constituent is a phrase of type utt-ana-ph:

(18) \[ \text{utt-anaph-int-cl} = \begin{cases} \text{dgb-params : } & \text{MAX-QUD : Question} \\ \text{cont} = & \text{max-qud : Question} \\ \text{hd-dtr : utt-anaph-ph} \end{cases} \]

This is exemplified in (19):

(19) a. Input utterance: A: Did Bo leave?
   b. Context assuming the reference of ‘Bo’ cannot be fully resolved: MAX-QUD: \(?x.\text{mean}(A,x,’bo’)\) (Who, is A referring to as ‘Bo’);
   c. Content of Bo? = MAX-QUDuestion (=Who, is A referring to as ‘Bo’?)

Scaling up (18) multimodally involves two moves:

1. generalizing phonological segmental parallelism to multimodal parallelism
2. positing a lexical entry for frowns

With respect to the former task we need to generalize the condition \(\text{phontype} = \text{rc.sit-type.phon}^\text{in (17)}\) so that it can apply to gestures, laughs and their combinations with speech. The most obvious generalization would be to require type identity with respect to form on all tiers. However, this will not work because in all cases small but important divergences actually need to apply. In the case of speech the identity is segmental identity, but not with respect to the speech contour (where the reprise is typically LH), whereas in the case of gesture reprises the face is required to involve a frown (in the FACS system Ekman and Friesen, 1978 a combination of \(A(ction)U(nits)\) 1 and 4 (Hager, 1985)). Indeed it seems like a repetition which will not work because in all cases small but important divergences actually need to apply. In the case of speech the identity is segmental identity, but not with respect to the speech contour (where the reprise is typically LH), whereas in the case of gesture reprises the face is required to involve a frown (in the FACS system Ekman and Friesen, 1978 a combination of \(A(ction)U(nits)\) 1 and 4 (Hager, 1985)).

In both cases, then, one needs to leave a channel free, presumably to express interrogative force. Hence, the most straightforward way to achieve this generalized parallelism condition is simply to specify the facial form as identity modulo specification of AUs 1 and 4 and the speech form as identity modulo intonation. An additional question is whether or not multimodal reprises require all channels to be reactivated, as exemplified in (21). We hypothesize that only the complete reprise can communicate a ‘what do you mean’ content, whereas the other reprises are understood as confirmations. However, clearly this requires experimental investigation.

(21) A: I don’t care + shrug. B: You don’t care + shrug + frown?/ You don’t care?/Shrug + frown

For now we will postulate a generalized utt-anaph type, building on (16)

(22) \[ \text{mm-utt-anaph-ph} = \begin{cases} \text{dgb-params : } & \text{rc : MMProp} \\ \text{formtype : Type} \\ \text{c1 : quasi-identical(rc.syn.form-type)} \\ \text{sync} : \text{formtype} \\ \text{cont} = \text{rc:mm-event}^\text{c1 : addressing(spkr,addr,t) : Prop} \\ \text{prop} : \text{Rec} \end{cases} \]

Why can a frown give rise to a clarification question in this context? We assume, following Ginzburg et al. (2020b), who in turn build on proposals of Scherer (1992); Wierzbicka (2000), that frowns communicate the emergence of a problem in interaction, more specifically involve the frownable giving rise to a question, which can indeed be spoken.9

(23) \[ \text{face : frownbrowtype} \]

\[ \begin{cases} \text{dgb-params : } & \text{spkr : Ind} \\ \text{addr : Ind} \\ \text{t : Time} \\ \text{c1 : addressing(spkr,addr,t)} \\ \text{q : Question} \\ \text{p : Prop} \end{cases} \]

9This is backed by entries on Eyebrow Raise and, even stronger, Eyebrow Cock in the Nonverbal Body Dictionary, which are described as signalling surprise, excitement, or general disbelief (http://bodylanguageproject.com/nonverbal-dictionary/, accessed April 27, 2021). Eyebrows are also used as question markers in sign languages (e.g. Baker et al., 2016, 132). There different kind of eyebrow movement are correlated with different types of sentences (e.g., yes-no vs. wh; see Freitas et al., 2014, 183, Tab. 3 for a particular clear overview of eyebrow use in Brazilian sign language questioning).
How to package this to attain a construction akin to (18)? There seem to be two options: assume that there is a single reprise fragment construction with certain components that are optional. On this line all instances spoken and purely gestural involve frowning with an utterance anaphora constituent involving phonological or gestural parallelism. The other option is to assume two subtypes of such a construction, a spoken one which involves an LH tone sequence, and a gestural where the interrogative force is driven by the frown. Choosing between these options requires a detailed experimental study, which we leave for future work. For concreteness we offer in (24) a sketch of the former strategy:

$$\begin{bmatrix}
\text{phon} : \text{LH} \\
\text{face} : \text{frownbrowtype} \\
\text{dgb-params} : [\text{MAX-QUD} : \text{Question}] \\
\text{cont-max-quad} : \text{Question} \\
\text{hd-dtr} : \text{mn-att-anaph-ph}
\end{bmatrix}$$

A precise semantic analysis along these lines of the discourse functions of gestures in multimodal interaction is attained (for a related work on the so-called what are you talking about face see Francis, 2020). Such analyses are needed in order to understand and model tier-crossing coherence in natural language processing, in both artificial and human agents. CRs are a key interactional competence in this respect.

5 Conclusions

Clarifications requests are an important dialogical resource for seeking mutual understanding and driving conversational interactions. However, in face-to-face dialogue CRs extend to the full range of verbal and non-verbal signals. We provided some data illustrating the phenomena at stake and introduced the basic ingredients to develop multimodal clarifications for linguistic theories.

This work fills in particular two explanatory gaps left by current multimodal studies, namely (i) projecting (non-emblematic) gestures to illocutionary acts, and (ii) connecting gestures to the basic dialogue dynamics of grounding and repair.

Acknowledgments

This work is partially supported by a public grant overseen by the French National Research Agency (ANR) as part of the program “Investissements d’Avenir” (reference: ANR-10-LABX-0083). It contributes to the IdEx Université de Paris – ANR-18-IDEX-0001

References


Seeing past words: Testing the cross-modal capabilities of pretrained V&L models on counting tasks

Letitia Parcalabescu\textsuperscript{1} Albert Gatt\textsuperscript{2} Anette Frank\textsuperscript{1} Iacer Calixto\textsuperscript{3,4}

\textsuperscript{1}Heidelberg University, Department of Computational Linguistics
\textsuperscript{2}University of Malta, Institute of Linguistics and Language Technology
\textsuperscript{3}New York University \textsuperscript{4}ILLC, University of Amsterdam

{parcalabescu,frank}@cl.uni-heidelberg.de
albert.gatt@um.edu.mt, iacer.calixto@nyu.edu

Abstract

We investigate the reasoning ability of pretrained vision and language (V&L) models in two tasks that require multimodal integration: (1) discriminating a correct image-sentence pair from an incorrect one, and (2) counting entities in an image. We evaluate three pretrained V&L models on these tasks: ViLBERT, ViLBERT 12-in-1 and LXMERT, in zero-shot and finetuned settings. Our results show that models solve task (1) very well, as expected, since all models are pretrained on task (1). However, none of the pretrained V&L models is able to adequately solve task (2), our counting probe, and they cannot generalise to out-of-distribution quantities. We propose a number of explanations for these findings: LXMERT (and to some extent ViLBERT 12-in-1) show some evidence of catastrophic forgetting on task (1). Concerning our results on the counting probe, we find evidence that all models are impacted by dataset bias, and also fail to individuate entities in the visual input. While a selling point of pretrained V&L models is their ability to solve complex tasks, our findings suggest that understanding their reasoning and grounding capabilities requires more targeted investigations on specific phenomena.

1 Introduction

Recently, many vision and language (V&L) models that combine images and text have been proposed (Lu et al., 2019; Tan and Bansal, 2019; Li et al., 2019; Chen et al., 2020; Li et al., 2020; Su et al., 2020). These models follow the pretrain-and-finetune paradigm, i.e. they are pretrained using self-supervision on large amounts of image-caption pairs\textsuperscript{1} and are then finetuned on the task(s) of interest. Such V&L models have obtained state-of-the-art performance across a number of different V&L tasks, e.g. visual question answering (VQA); visual commonsense reasoning; grounding referring expressions; and image retrieval, among others.

Pretrained V&L models use a combination of masked multimodal modelling – i.e., masking out words and object bounding boxes from the input and predicting them – and image-sentence alignment, i.e., predicting whether an image-sentence pair is correctly aligned or not. Such models hold the promise of partially addressing the ‘meaning gap’ in unimodal pretrained language models such as BERT (Devlin et al., 2019) by directly connecting language to visual representations (Bender and Koller, 2020; Bisk et al., 2020).

In this paper, we use foiling to investigate how well pretrained V&L models integrate and reason upon textual and visual representations. The foiling strategy, introduced by Shekhar et al. (2017) in the context of vision and language tasks, relies on replacing an element in a text with another element, such that the replacement results in a mismatch with the image. We propose two tasks which require effective multimodal integration: (1) discriminating a correctly aligned image-sentence pair from an incorrectly aligned one, and (2) counting entities in the image.

V&L models are commonly pretrained on task (1), and should not have many difficulties detecting incorrect image-sentence pairs. Counting, our task (2), nicely puts together visual and textual reasoning. It requires the detection of object instances in the visual input, mapping these instances to categories, as well as properly aligning such instances to references in the textual input. Model architectures have been proposed specifically for counting, which is known to be a hard V&L problem (Zhang et al., 2018; Acharya et al., 2019; Trott et al., 2018; Chattopadhyay et al., 2017). Unlike these specialised approaches, we focus on general-purpose V&L models. Related V&L work has also...
investigated generalised quantifiers (such as most) in a V&L context, but this work has generally exploited synthetic datasets (Sorodoc et al., 2018; Pezzelle and Fernández, 2019; Testoni et al., 2019).

Here, we task the model to judge whether an unambiguous question or statement about the number of entities visible in a natural image is correct.

We use three publicly available, representative V&L models in our investigation: LXMERT\(^2\) (Tan and Bansal, 2019), ViLBERT and ViLBERT 12-in-1\(^3\) (Lu et al., 2019, 2020). ViLBERT and ViLBERT 12-in-1 use the same BERT-based model architecture, which incorporates two separate visual and linguistic streams that interact through multiple co-attention transformer layers. ViLBERT is trained using self-supervised learning on image-caption pairs, while ViLBERT 12-in-1 is further finetuned on 12 different tasks using multi-task learning. LXMERT is also a dual-stream architecture and combines textual and visual transformer-based encoders with cross-modal layers. However, LXMERT is pretrained not only on image-caption pairs but also directly on the visual question answering (VQA) task using multiple VQA datasets.

While all models are trained on image-sentence alignment, only ViLBERT is not directly trained on VQA; hence the model can be probed “zero-shot” on our counting task. LXMERT, by contrast, is pretrained on VQA (including how many questions, the focus in our counting probe). Hence, LXMERT was exposed to examples where answering a question correctly requires the model to detect and categorise instances in an image, and then aligning these to the text. Finally, the tasks ViLBERT 12-in-1 are finetuned on also include VQA, including instances with numerical answers requiring counting abilities. We therefore believe it serves as a solid baseline and should be well equipped to detect foiled probing instances. To our surprise, we find that none of these models perform particularly well in our counting experiments.

Our main contributions are: i) We show that all three models perform image-sentence alignment well, as expected given their pretraining; ii) We build a counting probe, which requires a model to adequately perform cross-modal grounding; iii) We find that ViLBERT, ViLBERT 12-in-1 and LXMERT perform similarly to the random baseline when directly applying the image-sentence alignment head to perform counting without finetuning; iv) We find that all models seem to exploit dataset bias and fail to generalise to out-of-distribution quantities. Even when finetuned, they only partially solve our counting probe.\(^4\)

2 V&L Models

The pretrained V&L models we use are ViLBERT (Lu et al., 2019), ViLBERT 12-in-1 (Lu et al., 2020), and LXMERT (Tan and Bansal, 2019). ViLBERT is pretrained on Google Conceptual Captions (Sharma et al., 2018) on (i) multimodal masked prediction, i.e., masking objects and words and predicting them; and (ii) image-sentence alignment, i.e., determining whether a text corresponds to an image or not. LXMERT uses losses (i) and (ii) and is additionally pretrained on multiple VQA datasets, as well as object labelling. Finally, ViLBERT 12-in-1 starts from a pretrained ViLBERT model checkpoint and is additionally finetuned on 12 different tasks, once again including VQA.

2.1 Evaluation

In both image-sentence alignment and counting probes, models are exposed to either correct or foiled image-text pairs. We evaluate pretrained V&L models on our probes using accuracy (\(acc\)), which is the overall accuracy on all classes; precision (\(p_c\)), which measures how well the models identify the correct examples; and foil precision (\(p_f\)), which measures how well a model identifies foiled instances:

\[
\text{acc} = \frac{P + N}{P + N + \tilde{P} + \tilde{N}},
\]

\[
p_c = \frac{P}{P + \tilde{P}},
\]

\[
p_f = \frac{N}{N + \tilde{N}},
\]

where \(P\) and \(N\) are the number of true positives and true negatives, and \(\tilde{P}\) and \(\tilde{N}\) are the number of false positives and false negatives, respectively.

We also evaluate our models using a pairwise ranking accuracy \(acc_r\) computed using the image-sentence alignment score \(\phi\) that the model assigns to correct and foiled image-text pairs. Given an image \((i)\) paired with a correct \((c)\) versus a foil \((f)\) text, if the score of the positive/correct pair is greater than that of the foiled pair, the prediction is

\(^2\)https://github.com/huggingface/transformers
\(^3\)https://github.com/facebookresearch/vilbert-multi-task
\(^4\)We will release all data necessary to reproduce our experiments, including our counting dataset, upon publication.
considered successful.

\[
\text{acc}_r = \frac{\sum_{(i,c) \in C} \sum_{f \in F} s(i, c, f)}{|C| + |F|},
\]

where \( C \) is the set of correct image-caption pairs \((i, c)\), and \( F \) is the set of foils for the pair \((i, c)\).

### 3 Image-Sentence Alignment Probe

In this set of experiments, we probe whether pretrained V&L models can distinguish correct image-sentence pairs from foiled ones. While all models under consideration have been pretrained on this task, results are not usually reported for pretraining. Our aim is to explicitly establish their capabilities on a fundamental V&L task that we would expect them to perform well at, before probing them on the more challenging counting task.

#### 3.1 Data

To probe our models on the image-sentence alignment task, we construct evaluation sets using 5000 images each from the MSCOCO (Lin et al., 2014) and Google Conceptual Captions (GCC; Sharma et al., 2018) validation splits. MSCOCO images are collected from Flickr and its captions are crowdsourced. GCC’s images are obtained from the web with captions harvested from online alt-text enabled sources, and therefore contain more noise and variability. ViLBERT is pretrained on GCC image-caption pairs; LXMERT is pretrained on five datasets including MSCOCO, but not GCC. For both datasets we select one correct caption for each image, and create foils by pairing the image to one random caption from the remaining 4999 images. All models are tested on the same data.

#### 3.2 Experiments

In these experiments, we probe pretrained V&L models without any additional fine-tuning. Table 1 reports the results of applying the models’ pretrained image-sentence alignment prediction head to image-caption pairs from MSCOCO and GCC. We also highlight which models can be considered “zero-shot” in this setting: GCC is used to pretrain the two ViLBERT models, while MSCOCO is used when pretraining LXMERT.

ViLBERT performs very well on both datasets and achieves 96–97 \( \text{acc} \) overall. It predicts both correct and foiled examples well, as shown by 96–98 \( p_c \) and \( \sim 96 \ p_f \). When using ViLBERT 12-in-1, results on GCC are considerably worse compared to ViLBERT. This is surprising, since ViLBERT 12-in-1 was trained using more tasks and considerably more data than ViLBERT. Finally, LXMERT performs worst overall among all three models.

These results suggest that LXMERT (and to a lesser extent, ViLBERT 12-in-1) may be exhibiting catastrophic forgetting, a well-studied problem in neural networks (Robins, 1995) which has received attention in NLP (Kirkpatrick et al., 2017; Yogatama et al., 2019) as well as in V&L tasks in particular (Greco et al., 2019): LXMERT is finetuned on visual question answering in the last 10 epochs of pretraining, and ViLBERT 12-in-1 is finetuned on 12 different tasks. This finetuning may be responsible for the worse results observed, resulting in a downgrading of performance on the task the models were originally pretrained on.

In summary, all models solve the image-sentence alignment probe well (as expected) but the models show notable differences in performance; we conjecture catastrophic forgetting may be impacting the finetuning procedure of each model differently.

### 4 Counting Probe

In our second task we probe pretrained V&L models on their ability to count, i.e., to correctly predict the number of entities visible in an image, given the image itself and either a corresponding question coupled with a numerical answer, or a declarative statement about the number of entities of a specific kind derived from the question-answer pair (see Figure 1).
4.1 Data

We collect our counting probe data from Visual7W (Zhu et al., 2016), a VQA dataset with diverse question types including how many questions, where a correct answer requires the model to count the number of entities of a certain type in an image.

4.1.1 Data Formats

The data is originally in question-answering format, where the answer is a number. We experiment with two alternative formats.

**Q+A format** We concatenate the original question with the separator token [SEP] and each answer (correct and foil), e.g., the example in Fig. 1 becomes “How many magnets are on the bottom of the fridge? [SEP] 2/3/4/5”.

**Declarative format** ViLBERT is never pretrained on questions and answers with a separator token [SEP]. We therefore create a version of the counting data where we transform the question and answer into a declarative statement using simple templates, described in detail in Appendix A.2. For instance, we create the following statements for the example shown in Fig. 1: “There are 2/3/4/5 magnets on the bottom of the fridge.” Examples which could not be converted were removed.

4.1.2 Data Splits

To avoid leaks, instances extracted from a given Visual7W split are put into the same split in our counting dataset.5

We create three splits for our counting dataset: standard, hard, and interpolated. In the standard split we include all examples of how many questions in the train, dev and test splits in Visual7W, excluding examples that cannot be transformed into a declarative statement with our templates. The distribution of numerals in the standard split is highly skewed and answers such as “1” or “2” are by far the most common (see the outer circles in Fig. 2). We mitigate this by introducing a hard split (see the inner circles in Fig. 2), in which high-frequency classes are capped at \(k = 200\) examples for train, dev and test sets, and any training examples where the answer is a number greater than 20 are removed. Finally, in the interpolated setting we split the original data so that only examples whose answers are even are in the training set, with validation and test sets only containing examples with odd answers.

Data statistics are reported in full in Appendix A.3. We note that both the capping in the hard split, and the interpolation in the interpolated split, result in fewer instances. The hard split is more balanced with regards to the number of classes, whereas quantities in the standard split follow a more natural distribution, where numerals like “one”, “two” or “three” are more common than large quantities or mentions of empty sets (Figure 2). The less skewed distribution in the hard split would be expected to be harder, since we artificially lower the relative frequency of frequent answers.

4.2 Experiments

We conduct a number of experiments where counting capabilities are probed in different ways, via image-sentence alignment (Section 4.2.1), masked language modelling (MLM; Section 4.2.2), and visual question answering (Section 4.2.3).
Table 2: Counting: test results for models without finetuning on our counting dataset, including ViLBERT “zero-shot”. We report overall accuracy \( acc \), precision on correct examples \( p_c \), precision on foiled examples \( p_f \), and pairwise accuracy \( acc_r \). Splits: standard (std.) and hard. Formats: Q+A and declarative (decl).

<table>
<thead>
<tr>
<th>Split</th>
<th>Format</th>
<th>( acc )</th>
<th>( p_c )</th>
<th>( p_f )</th>
<th>( acc_r )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random baseline</td>
<td>50.0</td>
<td>50.0</td>
<td>50.0</td>
<td>50.0</td>
<td></td>
</tr>
<tr>
<td>ViLBERT</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>std. Q+A</td>
<td>37.8</td>
<td>74.3</td>
<td>25.5</td>
<td>49.0</td>
<td></td>
</tr>
<tr>
<td>decl. Q+A</td>
<td>37.6</td>
<td>77.9</td>
<td>24.1</td>
<td>57.0</td>
<td></td>
</tr>
<tr>
<td>hard Q+A</td>
<td>38.6</td>
<td>73.1</td>
<td>27.1</td>
<td>51.9</td>
<td></td>
</tr>
<tr>
<td>decl. Q+A</td>
<td>38.0</td>
<td>75.3</td>
<td>25.5</td>
<td>55.9</td>
<td></td>
</tr>
<tr>
<td>LXMERT</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>std. Q+A</td>
<td>50.5</td>
<td>45.7</td>
<td>55.2</td>
<td>57.2</td>
<td></td>
</tr>
<tr>
<td>decl. Q+A</td>
<td>54.7</td>
<td>51.0</td>
<td>58.5</td>
<td>72.8</td>
<td></td>
</tr>
<tr>
<td>hard Q+A</td>
<td>50.4</td>
<td>47.4</td>
<td>53.4</td>
<td>59.3</td>
<td></td>
</tr>
<tr>
<td>decl. Q+A</td>
<td>52.3</td>
<td>50.5</td>
<td>54.2</td>
<td>64.4</td>
<td></td>
</tr>
<tr>
<td>ViLBERT 12-in-1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>std. Q+A</td>
<td>43.3</td>
<td>80.2</td>
<td>30.9</td>
<td>77.3</td>
<td></td>
</tr>
<tr>
<td>decl. Q+A</td>
<td>62.4</td>
<td>73.7</td>
<td>58.7</td>
<td>75.4</td>
<td></td>
</tr>
<tr>
<td>hard Q+A</td>
<td>46.9</td>
<td>67.1</td>
<td>40.1</td>
<td>70.3</td>
<td></td>
</tr>
<tr>
<td>decl. Q+A</td>
<td>61.3</td>
<td>70.0</td>
<td>58.3</td>
<td>72.6</td>
<td></td>
</tr>
</tbody>
</table>

It is important to note that there is a difference between the three models in terms of their prior exposure to the VQA task in general, and to questions involving counting in particular. Specifically, while ViLBERT was exclusively pretrained on GCC, LXMERT’s pretraining involved VQA in the final ten epochs, and this included the Visual7W training set. In the case of ViLBERT 12-in-1, VQA was also one of the tasks on which it was finetuned, and again this included Visual7W. In the experiments reported below, we distinguish between a no finetuning and a finetuned scenario. In the former, we present results on models which were not directly finetuned on our counting training set, irrespective of whether they were exposed to Visual7W during pretraining (as in the case of LXMERT) or training (as in the case of ViLBERT 12-in-1). In the finetuned scenario, we finetune each model using three different random seeds and report mean and standard deviation for all metrics.

### 4.2.1 Counting as Image-Sentence Alignment

In this setup, we frame the counting task as an image-sentence alignment problem. We use the pretrained V&L models’ image-sentence alignment head either to predict whether the sentence (in Q+A or declarative format) matches the image or not (i.e., in a per-example comparison evaluated with \( acc, p_c, p_f \)), or to score correct and foiled pairs (i.e., in a pairwise comparison evaluated with \( acc_r \)). See Section 2.1 for details on how we compute these metrics. In Tables 2 and 3, we report our main results without and with additional finetuning on the counting training data, respectively.

**No Finetuning** As Table 2 shows, accuracy for both ViLBERT and LXMERT is below or close to the random baseline, improving slightly on the baseline on pairwise accuracy. We note that ViLBERT identifies correct image-sentence pairs relatively well (73–79 \( p_c \)), while failing on foils (24–27 \( p_f \)). This trend is also visible in ViLBERT 12-in-1; however all scores tend to improve when compared to ViLBERT (especially precision on foiled examples). Roughly, we can rank models according to their performance from worse to best: ViLBERT, LXMERT, ViLBERT 12-in-1. ViLBERT 12-in-1 is the only model that performs considerably above chance level according to standard accuracy when applied without direct finetuning (declarative format, standard and hard splits). From these initial results, it seems that whereas ViLBERT 12-in-1 is able to identify correct image-sentence pairs well (i.e., up to \( p_c \)), its most important gains come from improved precision on foiled examples (up to \( p_f \)).

Overall, results when performing pairwise scoring (\( acc_r \)) agree with the general trends in per-example results. We can clearly observe that ViLBERT performs close to chance, LXMERT is somewhat better (57–73), and ViLBERT 12-in-1 performs best (70–77). All models perform better when evaluated using pairwise accuracy compared to per-example accuracy. For example, ViLBERT 12-in-1 performs below chance level in per-example metrics (43 \( acc \)) but has good pairwise accuracy (77 \( acc_r \)). Thus, \( acc_r \) is a less strict metric than standard accuracy \( acc \).

**With Finetuning** In Table 3 we note that when models are directly finetuned on counting training data, results tend to improve (except for the interpolated data split, which we discuss separately further below). ViLBERT improves overall accuracy (\( acc \)) on the standard split considerably to about 71–74, but it still struggles on the hard and interpolated splits. Results on the hard split are not good and have very high variance, which could be due to the model overfitting on the small amount of counting training data. When finetuning ViLBERT 12-in-1
Further on counting data, results also improve compared to the ‘no finetuning’ setting. As expected, ViLBERT 12-in-1 clearly outperforms ViLBERT on both standard and hard splits according to all metrics evaluated: on per-example metrics (\(\text{acc}, \text{prec}, \text{prf}\)) and also according to a pairwise ranking comparison (\(\text{acc}_r\)).

Pairwise results exhibit more consistent differences between splits, i.e., interpolated \(\leq\) hard \(\leq\) standard. Both ViLBERT and ViLBERT 12-in-1 yield satisfying results of 76–83 \(\text{acc}_r\) in the standard and 56–77 in the hard split.

| Split | Format | \(\text{acc}\) & \(\text{prec}\) & \(\text{prf}\) & \(\text{acc}_r\) & \(\text{acc}\) & \(\text{prec}\) & \(\text{prf}\) & \(\text{acc}_r\) |
|-------|--------|-------|-------|-------|-------|-------|-------|-------|
| Random baseline | 50.0 & 50.0 & 50.0 & 50.0 & 50.0 & 50.0 & 50.0 & 50.0 |
| **VILOBERT** | | | | | | | | |
| std. | Q+A | 74.4 (0.2) | 49.3 (0.3) | 88.9 (0.3) | 78.2 (0.6) | 81.1 (0.3) | 60.3 (0.7) | 90.2 (0.2) | 83.5 (0.1) |
| | decl. | 71.7 (3.6) | 63.4 (1.1) | 88.6 (1.1) | 76.7 (2.7) | 81.1 (0.1) | 60.6 (0.1) | 89.7 (0.1) | 83.3 (0.1) |
| hard | Q+A | 56.7 (22.2) | 16.9 (11.9) | 75.4 (0.6) | 56.2 (0.7) | 64.3 (4.2) | 38.7 (3.3) | 86.0 (2.7) | 69.8 (2.6) |
| | decl. | 54.0 (21.4) | 38.6 (14.4) | 52.3 (37.0) | 57.5 (0.9) | 71.9 (1.9) | 46.4 (1.9) | 89.4 (0.8) | 77.5 (0.6) |
| interp. | Q+A | 48.0 (0.2) | 0.2 (0.1) | 65.6 (0.1) | 12.8 (0.3) | 52.5 (0.5) | 0.1 (0.1) | 67.6 (0.2) | 11.4 (1.4) |
| | decl. | 49.1 (0.6) | 0.3 (0.2) | 66.2 (0.3) | 17.9 (0.8) | 52.7 (0.3) | 0.0 (0.0) | 67.7 (0.1) | 13.5 (2.8) |

Table 3: Counting: test results for models fine-tuned on our counting training data. We report mean (std) over three runs: overall accuracy \(\text{acc}\), precision on correct examples \(\text{prec}\), precision on foiled examples \(\text{prf}\), and pairwise accuracy \(\text{acc}_r\). Splits: standard (std.), hard, and interpolated (interp.). Formats: Q+A and declarative (decl).

4.2.2 Counting as Masked Language Modelling

In this experiment, we set the image–sentence alignment head aside and employ the MLM capacity of LXMERT to further test its pretrained visual-linguistic representations on counting.

We mask the numeral in the declarative statements of our counting dataset and use LXMERT to predict the \([\text{MASK}]\) token (see Figure 3). The model assigns probabilities to all words in its English vocabulary, comprising more than 30k words. We remove vocabulary items that are neither numerals nor denote numerical quantities,\(^6\) sort the remaining items in order of descending probability, and obtain a list of all numerical quantities LXMERT predicts for the masked token, ordered by likelihood. In that list, we count the rank of the correct numeral in any formulation (e.g. “1”, “one”, “a”) and compute Recall@\(k\) and mean rank (MR). We report the results in Table 4, where we also show aggregate results per answer.

The results of the overall Recall@\(k\) and MR show clear differences between the standard and hard splits, whereas Recall@\(k\) and MR per numeral exhibit very consistent results for the same splits. Recall that in the hard split numerals are more evenly distributed, whereas in the standard split the frequencies of different answers follow a more Zipfian distribution (Figure 2). This shows that the model has a strong preference for the numerals “2”, followed by “1” and “0”, suggesting that performance is largely determined by the statistical bias in the training data, rather than the specifics of the visual input in relation to the text.

\(^6\)We include the indefinite article ‘a’ and the negation ‘no’ in our definition of a vocabulary item that denotes a numerical quantity, since they are interpretable as indicating ‘one’ and ‘zero’ respectively.
There are [MASK] zebras visible.
A: 2
LXMERT: two

There are [MASK] sides of the octagon shown.
A: 7
LXMERT: two

Table 4: Masked language modelling with LXMERT. We report Recall@k and the mean rank (MR) of the predicted numeral on our counting dataset’s test split.

| Num. | Recall@1 | | Recall@2 | | | | MR | | |
|------|----------|---|----------|---|---|---|---|---|
|      | std.    | hard | std.    | hard | std. | hard | overall | std. | hard |
|      | 55.0    | 31.5 | 71.1    | 45.0  | 6.6  | 13.5 |         |      |      |
| zero | 59.8    | 63.7 | 65.2    | 69.1  | 2.89 | 2.5  |         |      |      |
| one  | 86.2    | 86.6 | 89.7    | 89.0  | 1.6  | 1.7  |         |      |      |
| two  | 81.9    | 80.1 | 92.4    | 90.9  | 1.3  | 1.3  |         |      |      |
| three| 15.6    | 12.9 | 85.2    | 87.1  | 2.0  | 2.0  |         |      |      |
| four | 6.5     | 6.2  | 20.9    | 19.9  | 3.0  | 3.0  |         |      |      |
| five | 0.4     | 0.0  | 2.5     | 2.9   | 5.1  | 5.1  |         |      |      |
| six  | 0.5     | 0.0  | 0.5     | 0.0   | 6.7  | 6.7  |         |      |      |
| 7    | 0.0     | 0.0  | 0.0     | 0.0   | 11.0 | 10.7 |         |      |      |
| 8    | 0.0     | 0.0  | 0.0     | 0.0   | 9.7  | 9.3  |         |      |      |
| 9    | 0.0     | 0.0  | 0.0     | 0.0   | 12.3 | 12.3 |         |      |      |
| 10   | 0.0     | 0.0  | 0.0     | 0.0   | 17.6 | 18.2 |         |      |      |
| 11   | 0.0     | 0.0  | 0.0     | 0.0   | 20.4 | 20.4 |         |      |      |
| 12   | 0.0     | 0.0  | 0.0     | 0.0   | 15.1 | 15.2 |         |      |      |
| 13   | 0.0     | 0.0  | 0.0     | 0.0   | 25.4 | 25.5 |         |      |      |

Bias can also be observed when the MLM predictions are wrong. While Table 4 reports metrics depicting how often the model is correct, we further analyse the cases where model predictions are wrong. Among image-sentence pairs in the hard split where the model prediction is wrong, the model predicts: “two” 51% of the time, “no” 12% of the time (which we count as “0”), followed by “three” (12%), “four” (5%) and “five” (1%).

Good performance on low quantities reflects their frequent occurrence in V&L datasets (Goyal et al., 2017). The very poor performance on under-represented quantities suggests both a lack of generalisation of the V&L model, as well as limitations arising from models’ Faster R-CNN (Ren et al., 2015) visual backbone.

Table 5: Overall (in bold) and per-answer accuracy of LXMERT further fine-tuned on the VQA task (Antol et al., 2015) on the standard and hard counting splits.

<table>
<thead>
<tr>
<th>Answer</th>
<th>VQA accuracy</th>
<th></th>
<th>Answer</th>
<th>VQA accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>std.</td>
<td>hard</td>
<td></td>
<td>std.</td>
</tr>
<tr>
<td>overall</td>
<td>53.9</td>
<td>41.8</td>
<td>five</td>
<td>0.0</td>
</tr>
<tr>
<td>zero</td>
<td>94.4</td>
<td>93.7</td>
<td>ten</td>
<td>12.0</td>
</tr>
<tr>
<td>one</td>
<td>75.0</td>
<td>69.7</td>
<td>eleven</td>
<td>10.0</td>
</tr>
<tr>
<td>two</td>
<td>62.0</td>
<td>62.4</td>
<td>twelve</td>
<td>28.6</td>
</tr>
<tr>
<td>three</td>
<td>32.8</td>
<td>31.4</td>
<td>thirteen</td>
<td>12.5</td>
</tr>
<tr>
<td>four</td>
<td>25.0</td>
<td>21.3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>five</td>
<td>15.6</td>
<td>17.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>six</td>
<td>17.4</td>
<td>19.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>seven</td>
<td>4.0</td>
<td>3.8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>eight</td>
<td>18.0</td>
<td>14.5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>nine</td>
<td>3.6</td>
<td>2.5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ten</td>
<td>12.0</td>
<td>11.6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>eleven</td>
<td>10.0</td>
<td>8.7</td>
<td></td>
<td></td>
</tr>
<tr>
<td>twelve</td>
<td>28.6</td>
<td>32.3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>thirteen</td>
<td>12.5</td>
<td>7.1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

4.2.3 Counting as VQA

Finally, we frame the counting probe in its standard setting as a VQA problem, without foiling. We use the publicly available LXMERT model further finetuned on the VQA v2.0 dataset (Goyal et al., 2017). The test setting is the same as for the original VQA task: the model receives questions ("How many ...?") from our counting dataset as input and has to predict the most likely answer from a list of 3,129 possible answers. All answers in our dataset are contained in the model’s answer list. We report detailed results in Table 5.

The model achieves an overall 53.9 accuracy on the standard split and 41.8 on the hard split. The detailed accuracies per numeral show the same trend as Table 4. Differences in performance between the standard (Zipfian) and the hard (more balanced) splits are predominantly due to the different proportion of quantities in the splits. Once again, this reveals a lack of generalisation coupled with a surplus of bias exploitation potential: the model relies on highly frequent quantities like “one” or “two” as a “safe bet” when predicting under uncertainty.

A notable difference between using LXMERT’s MLM head without direct finetuning (as in Table 4), and using LXMERT’s VQA head further finetuned on VQA v2.0 (Table 5) is seen for the numeral “zero”: the model’s capacity to predict “zero” is enhanced by the finetuning process, to the detriment of the frequent quantities “one” and “two”. Fine-tuning also seems to improve prediction for numerals “four” to “six”, and also for “12”. Counting further than that is a challenge for LXMERT.

5 Related Work

Originally proposed for text-only models (Devlin et al., 2019; Wang et al., 2019; Lewis et al., 2020), “12” or “a dozen” is a frequent answer in VQA v2.0.
the pretrain-and-finetune paradigm has become the de facto standard for vision and language tasks. The core idea is that pretraining on large and diverse datasets should lead to robust multimodal representations, so that models can be easily fine-tuned for different tasks.

**Pretrained Vision & Language Models** Based on the pretrain-and-finetune paradigm, many pretrained V&L models have recently been proposed which combine images and text using BERT-like architectures. They include ViLBERT (Lu et al., 2019, 2020), LXMERT (Tan and Bansal, 2019), VisualBERT (Li et al., 2019), UNITER (Chen et al., 2020), Unicoder-VL (Li et al., 2020), VL-BERT (Su et al., 2020), among others. They can be classified into single- or dual-stream architectures: single-stream models concatenate words to object bounding box features and encode this sequence using a single transformer stack; dual-stream models have separate transformer stacks for visual and textual inputs, with layers to fuse these into multimodal features (i.e., co-attention layers). ViLBERT, ViLBERT 12-in-1, and LXMERT, i.e. the models we use with in this work, are all dual-stream.

Bugliarello et al. (2020) find that single- and dual-stream models perform comparably under similar conditions. Ilharco et al. (2020) show that contextual text-only language models such as BERT encode visual representations reasonably well, though they fall short of human performance.

In the context of the VALUE benchmark, Cao et al. (2020) report results on multiple V&L tasks, some of which we also corroborate, notably, the dominance of textual features compared to visual features in the model’s predictions.

**Vision & Language Models for Counting**

Counting is known to be hard for V&L models, and has been studied extensively (Seguí et al., 2015; Chattopadhyay et al., 2017; Trott et al., 2018; Zhang et al., 2018; Acharya et al., 2019).

Chattopadhyay et al. (2017) investigate strategies based on object detection, regression, subtitising and averaging over the results returned by a model ensemble. Trott et al. (2018) create the HowMany-QA counting dataset. Their Interpretable RL Counter (IRLC) model solves counting by iteratively including objects in a pool, whose size is then reported. Acharya et al. (2019) propose TallyQA, a large counting dataset which includes both simple questions (e.g. *How many giraffes?*) and harder cases involving additional properties (e.g. *How many giraffes are sitting down?*). Finally, Zhang et al. (2018) argue that attention bottlenecks compromise counting capabilities (see Zhang et al., 2018, Section 3), showing that an alternative architecture which includes a branch specifically designed to overcome the bottleneck for counting leads to considerable improvements.

**Counting and the attention bottleneck** The ‘attention bottleneck’ noted by Zhang et al. (2018) and further discussed by Acharya et al. (2019) generally afflicts architectures where the image pipeline has the general form “image → CNN → convolutional feature maps → attention bottleneck → prediction”. The ‘bottleneck’ is created by the attention mechanism between input and prediction layer. For details and examples, see Appendix A.4.

This issue does not apply to the pretrained V&L models reviewed above, or the models we experiment with. ViLBERT, ViLBERT 12-in-1, and LXMERT have two multi-layer transformer stacks to encode image and text, respectively. None of these models have an attention bottleneck; rather, the outputs of modality-specific encoders are integrated via multiple co-attention layers. When finetuning the model on a target task, a prediction head is commonly trained from scratch and uses the output of the last co-attention layer as input.

**6 Conclusions and Future Work**

We probed three pretrained V&L models on image-sentence alignment and counting: two tasks that require joint understanding of image, text and their correspondence. Our results show image-text alignment capabilities which range from good (for ViLBERT and ViLBERT 12-in-1) to satisfactory (for LXMERT). Our results highlight that LXMERT (and to a lesser extent, ViLBERT 12-in-1) may be suffering from catastrophic forgetting. As for counting, we observe sub-optimal performance in all models investigated, even after finetuning on counting data. In these models, there is limited evidence of grounding of symbols in visual data after pretraining; all models exploit biases in the data and seem to lack the capability to individuate entities in the visual input, a prerequisite for counting. Our results raise concerns about heavyweight V&L models, whose main selling point is their ability to solve complex tasks. Our findings suggest that understanding their capabilities requires more targeted investigations on specific phenomena. In
line with this reasoning, our ongoing work is aiming towards a benchmark that will address several linguistic phenomena in addition to counting. We hope such a benchmark will serve the community to probe the grounding capabilities of vision and language models on a broad range of linguistic phenomena.

More generally, we encourage researchers i) to report the performance on pretraining tasks, ii) to work towards effective pretraining, and iii) to test for catastrophic forgetting during fine-tuning. The high computational and environmental cost of current pretraining practices may outweigh the benefits of reusing such models, leaving the prospect of lightweight and green AI as a distant goal.

References


References


A Appendix

A.1 Training and evaluation setup

In our experiments with ViLBERT and ViLBERT 12-in-1, we use the https://github.com/facebookresearch/vilbert-multi-task codebase. In our “zero-shot” experiments, we use ViLBERT pretrained on image-sentence ranking and ViLBERT 12-in-1 pretrained on twelve tasks. In our experiments with LXMERT, we use the https://github.com/huggingface/transformers codebase. For evaluating LXMERT on image-sentence alignment or counting as MLM, we use the publicly available pretrained model.

When evaluating LXMERT on counting as VQA, we use the publicly available model additionally fine-tuned on the VQA 2.0 dataset (Goyal et al., 2017).

When finetuning ViLBERT and ViLBERT 12-in-1 on our counting dataset, we train models on the training split and evaluate on the concatenation of the validation and test splits (see Table 6 for details on the splits). We train all models for 20 epochs and evaluate always at the end of each epoch, therefore 20 times. For each model, we report the best scores obtained across all 20 evaluations. ViLBERT and ViLBERT 12-in-1 are finetuned on our counting data following the standard finetuning procedure of ViLBERT 12-in-1: AdamW optimiser (Loshchilov and Hutter, 2019) with a learning rate 4e-5 and a linear warm-up scheduler, batch size 16, and a maximum of 100 detected objects per image, a text backbone bert-base-uncased and configuration file bert_base_6layer_6conect.json. We finetune models using the binary cross-entropy loss where the task is to decide if an image-sentence pair is correct or a foil, and each instance consists of a question (or statement, see Section A.2 below) about the number of objects in the image and an answer (that might be correct or foiled).

A.2 Question-to-statement template

We create a few simple templates to convert <question, answer> pairs into a declarative state-ment. We denote the answer as A, and by definition it is always a number. Other capitalised letters (e.g., B, C, etc.) denote entire sets of words that are either copied over to the declarative sentence or removed according to the template. If a set of words is optional in the template, it is enclosed in brackets, e.g. [D]. A template is selected if there is substring match between the template’s key and the question. We denote negation by ~. We process templates in order so that if a template matches, it “consumes” the QA pair and produces a declarative sentence. If no template matches, the QA pair is ignored and not added to our counting dataset.

“are” How many B are there [C]? → There are A B [C]. E.g.: “How many black cats are there in the picture?” → “There are A black cats in the picture.”

“can you see” How many B can you see [C]? → You see A B [C]. E.g.: “How many elephants can you see?” → “You see A elephants.”

“do you see” How many B do you see [C]? → There are A B [C]. E.g.: “How many people do you see by the tree?” → “There are A people by the tree.”

“are” How many B are C? → There are A B C. E.g.: “How many glasses are on the table?” → “There are A glasses on the table.”

“can” How many B can C? → A B can C. E.g.: “How many surcoats can be found in the storage?” → “A surcoats can be found in the storage.”

“do” and “have” How many B do C have [D]? → C have A B [D]. E.g.: “How many headphones do the people have?” → “The people have A headphones.”

“does” and “have” How many B does C have D? → C has A B C. E.g.: “How many holes does he have in his pants?” → “He has A holes in his pants.”

“have” How many B have C? → A B have C. E.g.: “How many bottles have blue caps?” → “A bottles have blue caps.”

~ “is” and ~ “will” and ~ “does” and ~ “has” How many B? → There are A B. E.g.: “How many cars in the picture?” → “There are A cars in the picture.”
### Table 6: Counting data statistics.

<table>
<thead>
<tr>
<th>Split</th>
<th>#Train</th>
<th>#Valid</th>
<th>#Test</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Standard</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Correct</td>
<td>6,001</td>
<td>2,439</td>
<td>3,622</td>
</tr>
<tr>
<td>Foiled</td>
<td>17,896</td>
<td>7,283</td>
<td>10,800</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td>23,897</td>
<td>9,722</td>
<td>14,422</td>
</tr>
<tr>
<td><strong>Hard</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Correct</td>
<td>1,567</td>
<td>1,130</td>
<td>1,352</td>
</tr>
<tr>
<td>Foiled</td>
<td>4,672</td>
<td>3,378</td>
<td>4,040</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td>6,239</td>
<td>4,508</td>
<td>5,392</td>
</tr>
<tr>
<td><strong>Interpolated</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Correct</td>
<td>3,303</td>
<td>1,331</td>
<td>2,013</td>
</tr>
<tr>
<td>Foiled</td>
<td>9,840</td>
<td>3,969</td>
<td>5,998</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td>13,143</td>
<td>5,300</td>
<td>8,011</td>
</tr>
</tbody>
</table>

Table 7: Percentage of numerals in the counting data.

<table>
<thead>
<tr>
<th>Numeral</th>
<th>Train</th>
<th>Percentage (%)</th>
<th>Valid</th>
<th>Test</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>std.</td>
<td>hard</td>
<td>std.</td>
<td>hard</td>
</tr>
<tr>
<td>zero</td>
<td>9</td>
<td>11</td>
<td>7</td>
<td>15</td>
</tr>
<tr>
<td>one</td>
<td>30</td>
<td>11</td>
<td>26</td>
<td>14</td>
</tr>
<tr>
<td>two</td>
<td>25</td>
<td>9</td>
<td>32</td>
<td>15</td>
</tr>
<tr>
<td>three</td>
<td>14</td>
<td>11</td>
<td>13</td>
<td>13</td>
</tr>
<tr>
<td>four</td>
<td>8</td>
<td>9</td>
<td>9</td>
<td>18</td>
</tr>
<tr>
<td>five</td>
<td>5</td>
<td>12</td>
<td>4</td>
<td>10</td>
</tr>
<tr>
<td>six</td>
<td>3</td>
<td>11</td>
<td>3</td>
<td>6</td>
</tr>
<tr>
<td>7</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>8-10</td>
<td>3</td>
<td>11</td>
<td>2</td>
<td>5</td>
</tr>
<tr>
<td>10-20</td>
<td>2</td>
<td>7</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>21+</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

### A.4 Counting and the attention bottleneck

The attention bottleneck takes place when there is an image encoder model and there is a bottleneck between the model input and the layer that makes the predictions of interest. This situation can be exemplified where the image pipeline has the general form “image $\rightarrow$ CNN $\rightarrow$ convolutional feature maps $\rightarrow$ attention bottleneck $\rightarrow$ prediction”. We now use an idealised example meant to illustrate the attention bottleneck issue, similar to the one used in Zhang et al. (2018). The goal is to clarify when the issue should arise and in what conditions.

Imagine there is a *cat prediction* model, and we present it with an image with a single cat. After a number of CNN layers, the model computes a convolutional feature map $c_{i,j}$. In the attention bottleneck, a perfectly trained model will assign probability close to 1 to the “cat” feature vector, e.g., say feature map $c_{4,7}$, and 0 elsewhere, and the attention output will roughly be $1 \cdot c_{4,7} + 0 \cdot \sum_{i\neq4,j\neq7} c_{i,j}$.

We can now think of an idealised scenario where we create an identical copy of the cat image and paste it side-by-side with the original image (so that there are two cats), or we can think of another image which depicts two identically looking cats. When encoding any such image, each “cat” feature vector should get $\sim 0.5$ probability in the attention layer, again assuming an idealised and perfectly trained model. By design of the attention mechanism, the attention output will consist of the two sets of “cat” features multiplied by $\sim 0.5$ each and summed together. Therefore the attention output for the image with two cats would virtually be indistinguishable from the output for the image with a single cat. If the model only has access to these features, i.e., the attention mechanism is a bottleneck, it becomes very hard for the model to count, which by definition would require being able to differentiate the number of cats in the input image.

---

... standard split follow a more natural distribution, where numerals like “one”, “two” or “three” are more common than large quantities or mentions of empty sets (see Figure 2). This more skewed distribution is made even more evident in Table 7, which shows the percentage of occurrence of numerals in the standard split. The less skewed distribution in the hard split would be expected to be harder, since we artificially lower the relative frequency of frequent answers (compare the inner to the outer circles in Figure 2).
How Vision Affects Language: Comparing Masked Self-Attention in Uni-Modal and Multi-Modal Transformer

Nikolai Ilinykh  Simon Dobnik
Centre for Linguistic Theory and Studies in Probability (CLASP)
Department of Philosophy, Linguistics and Theory of Science (FLoV)
University of Gothenburg, Sweden
{nikolai.ilinykh,simon.dobnik}@gu.se

Abstract

The problem of interpretation of knowledge learned by multi-head self-attention in transformers has been one of the central questions in NLP. However, a lot of work mainly focused on models trained for uni-modal tasks, e.g. machine translation. In this paper, we examine masked self-attention in a multi-modal transformer trained for the task of image captioning. In particular, we test whether the multi-modality of the task objective affects the learned attention patterns. Our visualisations of masked self-attention demonstrate that (i) it can learn general linguistic knowledge of the textual input, and (ii) its attention patterns incorporate artefacts from visual modality even though it has never accessed it directly. We compare our transformer’s attention patterns with masked attention in distilgpt-2 tested for uni-modal text generation of image captions. Based on the maps of extracted attention weights, we argue that masked self-attention in image captioning transformer seems to be enhanced with semantic knowledge from images, exemplifying joint language-and-vision information in its attention patterns.

1 Introduction

Recently, we have seen a surge of interest in explainability research for large-scale neural networks, e.g. transformers (Vaswani et al., 2017). A lot of the existing literature focuses on the analysis of attention (Bahdanau et al., 2015) in terms of linguistic knowledge it encodes (Belinkov and Glass, 2019). Clark et al. (2019) show that attention heads’ patterns in BERT (Devlin et al., 2019) resemble syntactic dependencies present in the text. They also use a probing classifier to identify how knowledge of syntax is distributed between attention heads. Vig and Belinkov (2019); Hoover et al. (2020) have shown that visualising the structure of attention in transformer models can help us see which parts of the model capture specific syntactic knowledge. Voita et al. (2019) demonstrate that not all attention heads are equally suitable for learning syntactic information. Thus, pruning such heads can be an option to reduce the model’s complexity. While attention is not always an explanation (Jain and Wallace, 2019), some work (Ravishankar et al., 2021) has shown that extra fine-tuning on a syntax-related task can guide the model’s attention to truly resemble syntactic information about the text. Other approaches to the model’s interpretability include, for example, a work by Rethmeier et al. (2020), which inspects how knowledge is transferred on the neuron level rather than attention level.

While most of the existing research has placed the problem of model’s explainability in the context of uni-modal text-based tasks, e.g. machine translation, the field of language-and-vision is somewhat lacking similar analysis for models trained to solve multi-modal tasks. This becomes especially important with the increasing interest in adopting transformers for learning better cross-modal representations (Tan and Bansal, 2019). In addition, using large-scale models to improve grounding between language and vision representations (Lu et al., 2019) requires vigilance regarding how information is learned in different parts of such densely structured models. Multi-modal transformers are required to not only learn to perform symbol grounding, e.g. mapping natural language symbols into visual representations as defined by Harnad (1990) and a language model, but also learn to fuse information from two modalities, the nature of which has been an open question in the field (Lu et al., 2017; Caglayan et al., 2019; Ilinykh and Dobnik, 2020). The effect that such multi-modal representations have on the attention in large-scale models has not been addressed a lot in the language-and-vision literature. More specifically, we need a
better understanding of how self-attention in transformer processes the multi-modal information.

In this paper, we analyse the masked self-attention part of the image captioning transformer, which performs a standard language masking task based on the textual input, and compare its attention patterns with masked attention in distilgpt-2, a text-only transformer. Our goal is to identify what kind of knowledge is captured in representations learned by this part of the model and whether it is affected in any way by the visual modality, which is not directly accessible for this particular self-attention. We aim to answer the following questions:

- Does masked self-attention show patterns which resemble any syntactic knowledge of the input text?
- What are the differences in attention on previous words when generating the next word in either the uni-modal or multi-modal task set-up?
- What is the task’s effect (uni-modal vs. multi-modal) on the semantics of words captured by masked-self attention in image captioning transformer?

In addressing these questions, we believe that we show novel insights into how the information is transferred between inner self-attentions of complex architectures such as a transformer and how representations from specific components of such models are affected by the training objective and multi-modality.

2 Model

Fig. 1 shows the architecture of the image captioning transformer that we use for our experiments, first introduced by Herdade et al. (2019) and built on top of the basic image captioning transformer (Luo et al., 2018). This architecture resembles many parts of the classic transformer (Vaswani et al., 2017), which was initially introduced for machine translation, consisting of three multi-head self-attention mechanisms. The standard transformer’s encoder learns representations of the input text by passing it through two sub-layers: multi-head self-attention and feed-forward network. Each sub-layer has a residual connection around itself, followed by layer-normalisation operation. The decoder contains masked self-attention, which is used to learn linguistic knowledge of the ground-truth target translation. In a uni-directional task, it masks the words in the future so that the model learns to attend to the previously generated words only. The third self-attention is performing a cross-modelling task, using information from both encoder and decoder. This cross self-attention identifies correlations between the source text and currently generated target text in a machine translation context.

Once we reformulate the model’s task from machine translation to image captioning (Fig. 1), we naturally change the encoder’s inputs. Instead of
the source sentence, the encoder uses representations of objects from the image as its input. On the decoder’s side, the ground-truth captions that the model learns to generate are used as inputs during training. To prepare inputs to our encoder, we first extract visual features of the detected objects \( X = \{x_1, ..., x_N\} \), where \( x_n \in \mathbb{R}^{1 \times D} \) with \( N = 36 \) and \( D = 2048 \). We use a bottom-up feature extractor (Anderson et al., 2018), which is based on Faster-RCNN (Ren et al., 2015) and pre-trained on Visual Genome (Krishna et al., 2016) with the ResNet-101 as its backbone (He et al., 2016). For each detected object we also extract geometry features \( G = (x, y, w, h) \) (centre coordinates, width, height). In the next step, queries \( Q = W^Q X \) and keys \( K = W^K X \) are used to get scaled dot product \( \Omega^V \):

\[
\Omega^V = \frac{QK^T}{\sqrt{d_k}}
\]

Then, \( \Omega^V \) and geometric features \( G \) are combined, taking into account the displacement between the objects and producing a fused representation \( \Omega \).

Finally, each attention head \( h \) from each encoder layer \( l \) outputs a combination of values \( V \) and geometry-aware visual features \( \Omega \):

\[
\text{head}_{l,h} = \text{self-attention}(Q, K, V) = \Omega^V
\]

**Masked self-attention in the decoder** The idea of self-attention is that each token from the input text learns to attend to the other tokens from the same sequence. However, this is not feasible for the caption generation task since attending to the future tokens is unfair and it cannot be used when generating text. Therefore, the self-attention in the decoder is using masking of future tokens to keep the auto-regressive nature of the model. In particular, the token \( w_l \) and the future tokens \( w_{l+1}, ..., w_W \) are replaced with \([MASK]\). Then, \( w_l \) is predicted using the previous context in the standard left-to-right fashion: \( \hat{W}_{i:l} := (w_1, ..., w_{l-1}) \).

We have specifically focused on the analysis of the attention weights in the decoder’s masked self-attention of the image captioning transformer. We extract the attention weights for each head \( h \) in each layer \( l \) of this self-attention and use them for our visualisations and analysis. These weights are calculated similarly to the attended visual features (Eq. 1). Our masked self-attention has six layers, consisting of eight heads in each of them.

For the model checkpoint, we use the best model released by the authors of the architecture\(^2\). This checkpoint has been chosen on the basis of automatic evaluation scores: the model uses bottom-up representation of images, geometry features and self-critical training (Rennie et al., 2017). The captions are generated using beam search with beam width \( bw = 5 \) in the standard auto-regressive manner.

### 3 Learning syntactic knowledge

In our first experiment we investigate whether the attention weights of the masked self-attention are able to capture any general syntactic knowledge about the input text. It has been shown that the multi-head attention patterns in the transformer trained for the task of machine translation resembles syntactic properties of language at the level of part-of-speech tags and syntactic dependencies (Mareček and Rosa, 2019; Ravishankar et al., 2021). Since the self-attention that we are focused on is trained in a very similar task (masked language modelling), we first explore if particular layers and heads attend to specific part-of-speech tags the most. Then, we continue with the analysis of how information about syntactic dependencies is reflected in the learned attention patterns.

**Attention on Part-of-Speech** We follow Vig and Belinkov (2019) who compute the proportion of attention from each head that this head pays to tokens of a particular part-of-speech tag and accumulate the results over our test set:

\[
P(\alpha|tag) = \frac{\sum_{s \in S} \sum_{i=1}^{\|s\|} \sum_{j=1}^{\|s\|} \alpha(s_i, s_j, \text{pos}(s_j) = \text{tag})}{\sum_{s \in S} \sum_{i=1}^{\|s\|} \sum_{j=1}^{\|s\|} \alpha(s_i, s_j)}
\]

where \( S \) is the corpus of generated captions, \( \text{tag} \) is the part-of-speech tag of the attended word, and \( \alpha(s_i, s_j) \) is the attention from \( i^{th} \) word to \( j^{th} \) word for the given head. We use Spacy (Honnibal et al., 2020) to get part-of-speech tags of words and syntactic dependencies between them for all our experiments. We also perform normalisation (linear scaling) on the values of the calculated attention proportion to place all values in a single scale from 0 to 1. The masked self-attention is always given

\(^{1}\)For more details on how geometric information is combined with visual features in this model, we refer the reader to Herdade et al. (2019).

\(^{2}\)Available at: https://github.com/yahoo/object_relation_transformer
the START token at the start of the generation. We consider attention on this token non-informative (as it is over-attended) and ignore the corresponding attention weights for better visualisations. The heads pay only $\sim 26\%$ of their attention to the START token on average per caption. We use BertViz tool (Vig, 2019) to produce our visualisations.

The results for the five most frequently occurring part-of-speech tags (more than 1000 individual instances) are shown in Fig. 2. Words of such part-of-speech tags, which can be grounded in visual signals (nouns for objects, adjectives for attributes), receive attention from a large number of attention heads. On the other hand, only specific heads focus on words describing relations (verbs, adpositions). Specifically, seventeen heads (out of forty-eight) put more than 40% of their attention to the nouns, while only three heads give more than 30% of their attention to the verbs.

We also find supporting evidence for the previous studies (Belinkov, 2018; Vig and Belinkov, 2019), showing that deeper layers focus on more complex properties, e.g. relational part-of-speech tags (verbs), which require knowledge of objects learned from earlier layers (nouns). For example, the top 3 attention heads that attend to basic parts-of-speech such as determiners are all located in the model’s first three layers. For adjectives, the top 3 heads are similarly located in the first three layers of the model, with the maximum value of the attention head being 0.25. However, attention on adjectives is more spread across many heads in different layers, with the attention value being 0.14 for more than half of the heads, which is also a mean value for attention on this part-of-speech tag. A less clustered pattern is observed for nouns: its top 3 heads are located in layers 1, 3, and 4, with thirty-three heads paying more than 30% of their attention to nouns. We argue that the reason why the attention on nouns is scattered over many heads, with most of them paying nearly one-third of their attention to the nouns, is because nouns are continuously required for caption generation: the model needs to take them into account when generating either a relation or an attribute.

Somewhat differently, verbs are attended mostly in the model’s deeper layers: the top 3 most attentive heads are located in layers 3, 4, and 5 with values higher than 0.3. The vast majority of the heads (forty-three) have smaller attention values (less than 0.2), indicating that the model needs verbs only for specific situations, for example when a relationship needs to be generated. Overall, our visualisations demonstrate that masked self-attention weights resemble task-specific syntactic information about part-of-speech tags. For example, nouns are similarly attended across all heads since they are required for the captioning task the most (to describe, refer to, use in phrases, etc.). In contrast, more function-dependent parts of speech (verbs, adpositions) are attended to by fewer heads in the deeper layers of the model.

### Attention on Syntactic Dependencies

Fig. 3 shows the proportion of attention from the heads in masked self-attention for the most frequently occurring syntactic dependency relations. The proportions are calculated similarly to Eq. 3. In particular, we used the attention weights from root to the non-root part of the dependency phrase or vice versa, extracting dependencies in advance. This choice was affected by the auto-regressive nature of the generation task: for each word, we could only inspect attention focus on previous words. The attention on different dependencies seems to be distributed similarly to the attention on part-of-speech tags. More specifically, attention heads from the surface layers (1:5 and 2:1) seem to be focused on the determiner in the det relation. Comparing heatmaps of attention distribution on part-of-speech and syntactic dependency may give us intuition.

---

We use layer: head notation.
Figure 3: Attention distribution on different constituents of the specific syntactic dependencies. For det, compound, nummod we visualise which heads look the most on the non-root element of the dependency (e.g. “man” → “a” in “a man”). For pobj and prep we show attention in a different direction (e.g. “table” → “on” in “on table”, “with” → “bathroom” in “bathroom with”).

Figure 4: Attention distribution for the prep syntactic dependency. The left-side heat-map is computed for phrases where noun is a head in the phrase (“kitchen with”, while for the right-side heat-map it is the verb (“sitting at”).

about the specific heads’ role. For example, the heads 3:4 or 3:5 are not intensely active for the det relation, although they are among the most active heads when attending to the determiners. This indicates that these heads 1:5 and 2:1 may be more responsible for focusing on determiners when the phrase in the det relation is generated. Interestingly, many heads strongly attend to the numeral in the nummod dependency compared to all other relations. This could be related to the importance of learning about the number of objects in the scene, while other, simpler noun-based dependencies (det, compound) do not have to be attended so strongly.

Only a few heads specialise in dependencies that capture more complex properties (e.g. relations between different objects), with heads 3:3 and 3:6 being the most attending heads for pobj. The root of the prep phrase is often attended in the first layer, with only a few more heads in the later layers being activated. Could this pattern be mapped with the fact that roots in these phrases are often nouns and verbs? Fig. 4 shows that heads 1:3 and 1:7 are the most active heads when a noun is a root in the phrase of prep dependency. Same heads in the first layer are also active the most when looking at the nouns, according to Fig. 2b. This indicates that the model acquires basic knowledge of language syntax (dependencies, part-of-speech information) in its first layers. Similarly, as Fig. 4b demonstrates, the head 3:6 is the single most active head for the prep dependency. At the same time, according to Fig. 2d, this particular head is one of the few most active heads when the attention focus is on verbs. This might be interpreted as if this head is better at learning information about syntactic dependencies than other activated heads. We argue that it is helpful to look at the correspondence between attention on parts-of-speech and syntactic dependency since it is informative when determining specific heads’ roles and how important they are for different language tasks, e.g., part-of-speech tagging and syntactic dependency identification.

4 Multi-modality and masked self-attention

In this section, we look at how a multi-modal task of image captioning affects attention on the previous words when a masked self-attention model predicts the next word. We also compare our model’s attention patterns with patterns from an auto-regressive model, distilgpt-2 (Radford et al., 2019), which has been pre-trained on OpenWebTextCorpus. This model has 6 layers with 12 heads in each layer, which makes it more comparable to our captioning transformer than the standard GPT-2 model with 12 heads in each of the 12 layers.

Semantics of Attention Patterns Here, we compare the text-only uni-modal language model and its attention patterns with our multi-modal transformer’s masked self-attention. We do this because we want to investigate to what extent the attention patterns produced by the language model in the
Figure 5: Here are several examples of different attention visualisations for masked-self attention (MSA) from our image captioning transformer (TRSF) and distilgpt-2 (GPT2). The top row shows example images for which we generate a caption. The second and third rows show attention on the available context (indicated by the Source axis) when generating the next word (the Target axis). Word of the generated caption are displayed on the Source axis. To get more fine-grained visualisations in the third row, we exclude attention on the first token of each sentence for distilgpt-2 attention patterns since, based on our experiments and literature (Vig and Belinkov, 2019), attention on the first token is always very strong and not relevant. The fourth and the fifth rows show attention dispersion (AD) for each head in each layer. The colour bar in the second row indicates the range of values in all visualisations in this figure.

multi-modal setting differ from patterns where the task is uni-modal. For this, we run distilgpt-2 (Radford et al., 2019) on the captions generated by our image captioning transformer, where both
Figure 6: Visualisation of attention for example attention heads. The first row shows heads from the masked self-attention in our transformer; the second row depicts the head’s attention from distilgpt-2. The side to the left of the vertical line in the middle includes heads with high entropy in either of the models, while the right side contains heads with low entropy. The heads are denoted by a layer:head notation; they can be traced back to the more general attention concentration in Fig. 5m and Fig. 5q. Each figure displays attention from target (left) to source (right).

Figure 7: Mean normalised entropy of attention per head / layer calculated for the set of generated captions.

As demonstrated by Fig. 5, attention can constantly focus on particular words (e.g., nouns) while the caption is generated. We seek to identify which attention heads are responsible for focusing on its surroundings the most, and the attention does not generally look at a single word for too long.

We believe that this is an artefact of the training for image captioning task: our masked self-attention learns to focus on nouns because they ground objects, and most of the time, the following words form a single phrase referring to these objects. For example, attention on “lamp” for the third image is very strong throughout the generation of the whole phrase “lamp and a bunch of”. Once a new object is introduced (“oranges”), the attention shifts to this object for a different phrase (“oranges on a table”). The visualisations show that captioning transformer’s masked self-attention learns global, phrase-based semantic features of sentences. In contrast, in the text-only setting, the model learns about local relations between words in a sentence. For example, distilgpt-2 continuously shifts its maximum attention after every 2-3 words are generated, indicating that it learns to capture local relations between words (“bunch of”, “oranges on”).

Attention Focus As demonstrated by Fig. 5, attention can constantly focus on particular words (e.g., nouns) while the caption is generated. We seek to identify which attention heads are responsi-
ble for such observed patterns in the masked self-attention of the image captioning transformer. This is potentially important for reducing the model’s complexity by pruning non-important heads, which do not have an interpretable role defined by the measure of choice. Therefore, we calculate the entropy of attention distribution (Ghader and Monz, 2017) and use it as the measure of dispersion between attention weights:

\[ \text{Ent}_\alpha(s_j) = - \sum_{i=1}^{[s]} \alpha(s_i, s_j) \log(\alpha(s_i, s_j)) \]  

As Fig. 7a demonstrates, many heads in distilgpt-2 have high entropy scores which means that attention here is highly dispersed. The entropy increases in the deeper layers of the model. This correlates with the fact that deeper layers capture more distant syntactic relations and, therefore, lead to higher entropy scores (Vig and Belinkov, 2019). Fig. 7b shows the entropy scores for attention heads in captioning transformer’s masked self-attention. Here, most heads have a relatively low entropy, with only some of them with higher entropy in the model’s first layers.

Do heads have high/low entropy? Based on the examples of attention heads from Fig. 6, we can conclude that high entropy reflects a stronger concentration of attention from target words on particular source words to learn specific information. Such pattern can be observed, for example for captioning transformer’s masked self-attention heads in Figures 6a and 6b. Note that these heads heavily link several words with nouns (e.g. “plate”), which increases the head’s entropy - many words in the target sentence attend to a single word from the context. Another important observation is that the attention distribution from target to source is not always strong: not every word on the left side has a connecting line to the right side, indicating that attention is used to learn only specific properties. For example, as Fig. 6b demonstrates, focusing on “plate” when other objects (“sandwich”, “salad”) are mentioned may indicate that the model learns the notion of scene structure reflected in the text. At the same time, Fig. 6a shows that focusing on “plate” can be required when generating relations between objects, e.g. “plate with a sandwich and a salad”. However, as figures 6e and 6f demonstrate, distilgpt2 learns somewhat different attention between the source and the target words. While these patterns demonstrate that many words in the target sequence tend to focus on the specific words from context, each attention connection is not as strong as for the heads of the captioning transformer’s masked self-attention. The distilgpt-2 model does not focus on the caption’s specific relations or properties. Instead, it learns weak attention between all words. The heads’ entropy is high as the attention is dispersed, but each attention connection’s is also not as strong as it is in the captioning transformer’s masked self-attention. The examples of heads with low entropy (the right side of the Fig. 6) indicate that there is a word in the context that will be attended for each generated word.

5 Attention Alignment

It may be the case that the observed differences in attention patterns discussed in the previous section are simply due to different frequencies of words (in particular nouns) in the dataset on which the models are trained. For example, the multi-modal decoder also attends on the closest syntactic relations in the same way as a uni-modal decoder, but these happen to be nouns simply because there are more nouns in image captions. To test this hypothesis we calculated the Pearson correlation coefficient between the frequency of the nouns in our captions and attention distribution on the context words attended by heads when the next word is produced. The test has not shown a statistically significant correlation between the frequency of the nouns versus attention distribution on the context words in multi-modal decoder’s self-attention \((r = 0.49, p = 0.056)\). However, we observed a moderate positive correlation between the frequency of the nouns versus attention distribution in the uni-modal decoder’s attention \((r = 0.60, p = 0.014)\). These differences in correlations show that the uni-modal architecture is more biased to frequencies, whereas in a multi-modal setting, the effect of noun frequency is diminished. This provides support to our hypothesis, namely that this bias towards nouns is coming from somewhere else, e.g. the multi-modal representations that the language model is grounded in.

Since the model’s parameters are jointly updated with an end-to-end training through back-propagation, representations learned by different self-attention mechanisms are expected to be aligned with each other. We present a small preliminary analysis of whether the attention weights in
Figure 8: Attention shifts in cross-modal attention. The left-side column of each sub-figure shows the generated caption one word at a time. The right-side column depicts the labels of the 5 most attend objects in images when generating each word.

The cross-modal self-attention (cross-self-attention from Fig. 1) are responsible for information fusion between image encoder and text decoder. Our hypothesis is as follows: if cross-modal self-attention pays a significant portion of attention to the objects, which are generated as nouns in the caption as content words, we can conclude that due to the learning objective and nature of the information flow within the model’s components, decoder’s self-attention aligns with a higher-level cross-modal self-attention. In this case, we also expect that for every non-content word (e.g., determiner, preposition), the cross-attention keeps its attention on the most recent content word similar to what we observe for decoder’s self-attention in Figs. 5e–5h. We use two example images and examine the differences among the top 5 most-attended objects for every word generated in image captions. We use the predicted labels from the feature extractor (Anderson et al., 2018) to refer to the detected objects. Fig. 8 shows changes in cross-modal attention on objects during generation of descriptions. From Fig. 8a we can see that every time a new content word is generated ("plate", "sandwich", "salad"), the cross-modal attention tends to focus on objects with labels that are similar to the generated content words. For example, “lettuce” and “salad” are among the most attended objects when the transformer is preparing to generate the content word “salad”. Also, the same objects are continued to be attended when other non-content words are generated. This example provides initial evidence how text generation of nouns as exemplified by the decoder’s attention is linked to multi-modal representations as exemplified by cross-modal attention on objects. The results suggest that in multi-modal settings models learn representations that are fused and aligned with each other. Since the self-attention in the uni-modal architecture only needs to generate the text one word at a time by taking into account only previously generated words, it learns a pattern over local syntactic dependencies. In our future work, we would like to provide a more detailed analysis of the cross-modal attention and the uni-modal visual attention and therefore further strengthen the arguments how multi-modality affects knowledge that different parts in the large scale transformer models learn.

6 Conclusion

We have shown that attention patterns learned by a sentence decoder module of a multi-modal transformer are highly affected by the task that the model is optimised for. We focused on the masked self-attention in a sentence decoder in an image captioning transformer, demonstrating that its attention weights resemble linguistic knowledge, which is affected by the task of image captioning. This indicates that such language model acquired important aspects of grounded semantics. Simultaneously, we show that it is important to be cautious when applying large-scale pre-trained models on specific tasks to different semantic tasks as the original task does have an impact on the semantic representations learned. Our future work will focus on further examination of self-attention in the other two components of the multi-modal models which will give us an even clearer picture on what representations are learned by them.

Acknowledgements

The research reported in this paper was supported by a grant from the Swedish Research Council.
(VR project 2014-39) for the establishment of the Centre for Linguistic Theory and Studies in Probability (CLASP) at the University of Gothenburg. We thank anonymous reviewers whose comments helped improve this manuscript.

References


Matthew Honnibal, Ines Montani, Sofie Van Lengdeghem, and Adrian Boyd. 2020. spaCy: Industrial-strength Natural Language Processing in Python.


EMISSOR: A platform for capturing multimodal interactions as Episodic Memories and Interpretations with Situated Scenario-based Ontological References

Selene Báez Santamaría, Thomas Baier, Taewoon Kim, Lea Krause, Jaap Kruijt, Piek Vossen
Computational Linguistics and Text Mining Lab (CLTL)
Vrije Universiteit Amsterdam, Netherlands
s.baezsantamaria,t.baier,t.kim,l.krause,j.m.kruijt, piek.vossen@vu.nl

Abstract

We present EMISSOR: a platform to capture multimodal interactions as recordings of episodic experiences with explicit referential interpretations that yield an episodic Knowledge Graph (eKG). The platform stores streams of multiple modalities as parallel signals. Each signal is segmented and annotated independently with interpretation. Annotations are eventually mapped to explicit identities and relations in the eKG. As we ground signal segments from different modalities to the same instance representations, we also ground different modalities across each other. Unique to our eKG is that it accepts different interpretations across modalities, sources and experiences and supports reasoning over conflicting information and uncertainties that may result from multimodal experiences. EMISSOR can record and annotate experiments in virtual and real-world, combine data, evaluate system behavior and their performance for preset goals but also model the accumulation of knowledge and interpretations in the Knowledge Graph as a result of these episodic experiences.

1 Introduction

Multimodal interaction in real-world settings using sensors between humans and agents is a complex process. Furthermore, it typically evolves over time and within a shared (physical) space, being bound yet remaining continuously dynamic. The fact that certain contextual factors are not physically present, such as past episodic encounters, background knowledge and intentions, adds to this complexity. Agents designed to behave intelligently need to handle this complexity and form teams with people to collaborate and achieve shared goals.

Within the Hybrid Intelligence framework,\(^1\) we are specifically interested in such collaborative settings and focus on analysing what causes such systems to succeed or fail. Collaboration requires shared grounding and partially shared understanding of situations, communications, and references across modalities. As humans and agents may have different beliefs and perceptions of these situations, we argued in previous work (Vossen et al., 2018, 2019a) that agents need a theory-of-mind (ToM) (Premack and Woodruff, 1978; Leslie, 1987) to handle conflicts, miscommunication and errors in referential grounding and interpretations.\(^2\)

Although there are many initiatives for representing multimodal interactions, referential grounding is hardly handled in its full complexity. Most approaches to multimodal interaction data either label media such as video, images, or audio with annotations or simply present situated agent interactions in the form of dialogues or actions without labeling. In these approaches, annotations may be seen as interpretations of direct “behavioral” responses (utterances or actions) to the preceding signals. However, they lack a formalization of these interpretations into an explicit model that supports transparent reasoning. Such multimodal data sets can be seen as episodic experiences but not yet as knowledge-aware episodic memories that reflect the cumulative result. The latter requires interpretations of different multimodal signals to be combined in an explicit knowledge structure according to an ontological model that reflects our conceptualization of the world. In addition, this model needs to handle alternative interpretations, uncertainties and conflicts as the interpretations are not always correct or consistent.

We propose a generic model that can capture multimodal interactions as recordings of episodic experiences with explicit referential interpretations that also yield an episodic Knowl-

\(^1\)www.hybrid-intelligence-centre.nl

\(^2\)makerobotstalk.nl
edge Graph (eKG) as a ToM. **EMISSOR** stands for Episodic Memories and Interpretations with Situated Scenario-based Ontological References. The platform stores multiple streams of modalities as parallel signals. Each signal can be segmented and annotated independently with interpretation, providing robustness and simplicity. Signals can thus represent natural conversations in situated contexts in which (visual) actions and (verbal) utterances can be responses to each other but can also happen independently. EMISSOR can represent any (multimodal) interaction that takes place either in virtual or real-world settings, involving any virtual or real-world agent.

Annotated signals do not necessarily stand on their own, but can be mapped to explicit identities, relations, and properties in an eKG for capturing time-series of instances of situations. These mappings ground signal segments to formal instance representations and ground different modalities across each other. Time-bound experiences are thus captured as episodic experiences, i.e. as an explicit cumulative interpretation of streams of signals. The eKG models knowledge and interpretation shifts over time and supports reasoning over the interpretation. By keeping track of the provenance of signals and interpretations, our model reflects alternative ToM interpretations from different sources, modalities and experiences.

In the current paper, we set out the basic design and structure of our representation and our motivation. We first discuss in Section 2 representations of multimodal interaction proposed in various paradigms such as virtual games, agent interactions, multimodal dialogue systems. In Section 3, we describe the desiderata for our proposal for representing interactions, which combines aspects from the different approaches in the related work but adds a KG as the host of such transparent episodic memories for situated experiences. We elaborate on the different data layers and relations in our proposal. We discuss how different types of data sets can be converted, aligned and annotated such that segments get grounded to identities in a Knowledge Graph using an annotation tool. Future research and conclusions are presented in the final Section 4.

### 2 Related work

Interaction data takes many forms. Not only can it come in different modalities (visual, audio, text, action), but we can also have different types of interactions, e.g., search, question-answering, command-action sequences, (task-based) dialogues, navigation tasks, games, graphical interfaces, plain video, and audio recordings. It is impossible to provide a comprehensive overview of representations in each separate modality and interaction type in this paper.

For our research on social communicating robots, we are interested in representations of multimodal interactions with referential grounding across modalities and the representations of these modalities as such. Therefore, we discuss mainly works on modalities aligned in time series representing interactions. This excludes data in single modalities, such as plain text corpora with dialogues (spoken or text) and image or video data without dialogues. It also excludes static data that does not represent temporal sequences of data. For example, visual data labeled with textual descriptions and textual data augmented with visual scenes do not necessarily represent interactions. In interactions, modalities partially complement each other, such as speech responding to speech or to scenes and actions following speech. Such sequences often react to and complement each other and reflect some degree of causality and coherence, but not entirely. Augmented modalities, on the other hand, mainly represent paired data where one modality describes or illustrates the other. Collections of augmented data, e.g., labeled Flickr images, do not exhibit coherence across data points and do not reflect causal interaction. Nonetheless, single modality data and non-interactive data can still aid in processing multimodal interaction. Models and classifiers trained on such data can automatically annotate scenes in a time series of multimodal interaction data. An interesting research question would be whether static or single modality annotations also model collaborative interactions without considering temporal, causal, and coherence relations across data.

A recent survey of interactive dialogue datasets is given in (Serban et al., 2018). Serban et al. (2018) differentiate dialogue systems by the types of interaction (goal(s), non-goal chit-chat, topical); their modalities (written, spoken, video); the participants (Human, Agent); being constrained, spontaneous, scripted, fictional; being goal-oriented, domain-specific or open. Most of the data described is, however, non-situated or not situation-grounded. There is hardly any reference to and interaction with phys-
ical contexts. Many of these datasets and tasks have been developed and presented in SigDial\(^3\), the ACL Special Interest Group, for research on dialogues structures and models. A more applied perspective is taken by the Dialogue System Technology Challenge (DSTC\(^4\)). DSTC provides a platform for researchers and industry to develop and evaluate agent-interaction systems. Older datasets mostly contain conversational data for chatbots. A more recent challenge, Audio Visual Scene-Aware Dialog (AVSD, (Alamri et al., 2019)), contains short video clips with audio, descriptive captions, and a dialogue history. Participating systems need to answer a follow-up question or ground an answer to an image from the video or an audio fragment. This challenge represents situated-references and grounding across modalities, but the conversation is very descriptive: the conversations describe situations rather than being embedded in them. Below, we briefly describe some well-known datasets and challenges that represent various types of interaction data.

ParlAI\(^5\) released more than a hundred conversational datasets covering a wide range of topics, but most are single modality chat (Miller et al., 2017). SIMMC\(^6\) is Facebook’s sequel to ParlAI with Situated and Interactive Multi-Modal Conversation (Crook et al., 2019). It consists of task-oriented dialogues in multimodal contexts represented by collections of images. The data contains referential relations between the dialogues and situations, but the current challenge is restricted to the e-commerce contexts of buying furniture and fashion items. There is no grounding to complex situations but intentions and goals are explicitly represented. Facebook-research also launched various other related tasks, among which RECCON\(^7\): Recognizing Emotion Cause in Conversations (Poria et al., 2020), and MINIRTS: Hierarchical Decision Making by Generating and Following Natural Language Instructions in Real-time strategy game environments\(^8\) (Hu et al., 2019). The former grounds dialogues to emotions and their causes but not to visual or audio data. The latter grounds language to a closed virtual world by references to objects, agents, and actions. The dialogues are limited to commands and instructions to operate the game.

Google developed Schema-Guided-Dialogue (SGD\(^9\)) for task-oriented conversational agents (Rastogi et al., 2020). In addition to e-commerce services, the tasks involve intent prediction, slot filling, dialogue state tracking, policy imitation learning, language generation, and user simulation learning. The goals are defined, but there is no multimodal situational grounding.

An older comprehensive robot platform is provided by openEASE\(^10\), which is a web-based knowledge service providing robot and human activity data constituting episodic memories (Beetz et al., 2015). It produces semantically annotated data of manipulation actions, including the agent’s environment, the objects it manipulates, the task it performs, and the behavior it generates. EASE uses so-called NEEMS (Narrative Enabled Episodic Memories) as episodic memories. NEEMS consist of a video recording by the agent of the ongoing activity. These videos are enriched with a story about the actions, motion, their purposes, effects, and the agent’s sensor information during the activity. EASE is not data-centric but a service platform that uses a knowledge database as a back-end. The database can be explored through prolog queries. The focus of openEASE is on physical interactions and not on conversations with complex referential relations between expressions and situations.

Microsoft created a Platform for Situated Intelligence, PSI\(^11\) (Bohus et al., 2017). PSI offers multimodal data visualization and annotation tools, as well as processing components for various sensors, processing technologies, and platforms for multimodal interaction. PSI models multimodal situations and interactions within and comes close to a comprehensive solution. However, PSI is a software integration platform through which developers can share modules using a streaming architecture for signal annotation. Interactions are not stored in a shared representation and the platform cannot be used for sharing experimental data independently.

Action Learning From Realistic Environments and Directives (ALFRED\(^12\)) is a recent benchmark for mapping natural language instructions and ego-
centric vision to sequences of actions for household
tasks (Shridhar et al., 2020). ALFRED releases
challenges and leader boards for trajectory tasks in
which a human instructs an agent through natural
language to carry out specific household tasks in
a virtual world. The task is grounded in situations
and combines video, audio, and text for clear goals.
There is no natural dialogue that is independent of
the task.

The previous datasets all involve agents. Video
recordings of interacting people can be seen as
another source of data. MELD\(^\text{13}\) and COSMIC\(^\text{14}\)
represent the Friends sitcom through videos, time-
stamped dialogues and emotion annotations (Por-
ria et al., 2019; Ghosal et al., 2020). Two sea-
sons from the Friends dialogues were also anno-
notated with person references and identities by (Choi
and Chen, 2018) for the SemEval2018-task4 on
and for Q&A on open dialogues (Yang and Choi,
2019). IEMOCAP\(^\text{15}\) created detailed multimodal
recordings of scripted human-human conversations
with annotations of participants, gaze, gestures,
etc. (Busso et al., 2008). Both datasets do not
provide any further situated-references, and there is
no specific goal set for the conversation. Other similar smaller datasets with audio-visual emotion
expression are RAVDESS (Livingstone and Russo,
2018), TESS (Dupuis and Pichora-Fuller, 2010)
and SAVEE (Haq and Jackson, 2010).

2.1 Annotation schemes

Interaction data come in different formats and fol-
dow different schemes. DiaML (Bunt et al., 2012)
is a modeling language for the annotation of dia-
logues as a discourse. However, it does not tackle
the grounding problem. It does not make use of
identifiers that represent identities of entities, con-
texts, and situations independently of their men-
tions as it targets single modality data.

VOXML (Pustejovsky and Krishnaswamy,
2016) is a formal modeling language for captur-
ing spatial semantics of object entities in 3D sim-
ulations. VOXML tackles grounding but does not
model dialogue interaction nor the sequential align-
ment of cross-modality segments. It is a model
for defining the semantics of linguistic expressions
through physical world simulations, and it does not
model this world per se independently of these
expressions. Furthermore, it is a formal symbolic
representation that relies on a fully descriptive rel-
lation between language expressions and situation
modeling.

The Simple Event Model or SEM is a Resource
Description Framework (RDF) model for captur-
ing situations following semantic web principles
(Van Hage et al., 2011). SEM represents situations
as event instances through URIs, with ac-
tors, places, and temporal relations to OWL-Time
objects\(^\text{16}\) either defined as time points or as pe-
riods. Situations can be related as sequences in
time series through OWL-Time grounding, as well
as through explicit temporal and causal relations
between events. SEM can be used to construct
event-centric KGs rather than entity-centric KGs.
Event-centric KGs are well-suited for represent-
ing temporal properties of situations and entities
within these. Furthermore, they are not limited to
the predefined properties of entity-centric graphs
but exploit abstract event-participant relations that
can be further modeled in additional ontologies
(Segers et al., 2018).

The Grounded Representation and Source Per-
spective (GRaSP) model (Fokkens et al., 2017),
augments SEM with grasp:denotes relations be-
tween linguistic expressions (so-called mentions)
and their referential identities. Through GRaSP,
any segment in a signal (verbal, audio, or video)
can be mapped to an instance in a SEM model, as
such providing a flexible framework for referential
grounding. Although SEM and GRaSP can be used
for any modality, they have mostly been used for
representing events in text (Vossen et al., 2016). In
(Vossen et al., 2018, 2019a), we have shown that
GRaSP can also be used for modeling multimodal
situations with unaligned signals to model a theory-
of-mind or ToM in which different modalities and
different sources can generate alternative interpre-
tations that can co-exist in the robot’s eKG. Similar
to (Kondratyuk and Kennington, 2017), our robot
eKG reflects the episodic accumulation of knowl-
edge through interaction over time. Our model
differs from theirs in that our model allows for
alternative facts and properties.

DIAML, VOXML and GRaSP are complex
XML representations. Most of the DSTC datasets,
however, follow a more basic schema in JSON
format. Data elements represent sequences, possi-
bly including time stamps and pointers to sepa-
rate media files, possibly including bounding box

\(^{13}\)affective-meld.github.io/
\(^{14}\)github.com/declare-lab/conv-emotion
\(^{15}\)sail.usc.edu/iemocap/
\(^{16}\)www.w3.org/TR/owl-time/
coordinates, scene interpretations, participants, utterances, and goals to achieve. Google’s Schema-Guided-Discourse, Amazon’s Alexa Topical-Chat and Facebook’s SIMMC provide comparable JSON formats for capturing simple situations, goals, participants and dialogues with communication.

2.2 Our contribution

Although we can use many aspects of the previously discussed data and representation models, none of these completely provide what we need for modeling streaming data in different modalities representing parallel sequences of signals within physical or virtual world contexts while allowing for alternative interpretations. Most of the described data and models do not entirely represent the contextual situation in which conversations are embedded. Furthermore, they lack the means to represent the cumulative result of interpreting streams of multimodal signals over time.

In our representation, we combine the best of two worlds. On the one hand, we use light-weight JSON-LD\(^{17}\) for data in different modalities, their segmentation, alignment, and annotations, which provides us with the flexibility to easily represent any data streams; on the other hand, we use RDF to represent interpretations of such multimodal situations and the referential relations to explicit identities following GRaSP, which allows us to reason over the data in a robust way. Likewise, our framework can model the interpretation from raw signals to interpreted segments up to the situation-centric aggregation of triples over time as an episodic memory.

Our approach to connecting multimodal situated interactions to an explicit Knowledge Graph comes close to openEASE (Beetz et al., 2015), except that we focus on complex reference relations between conversations and situations rather than on robot actions only. Furthermore, we follow a data-centric approach in which an open representation of the interaction forms the basis for sharing data, tools, and solutions, whereas interactions in openEASE can only be accessed through queries. In contrast to the episodic triples from conversation generated by (Kondratyuk and Kennington, 2017) and the episodic NEEMS of openEASE, our eKG incorporates a ToM model based on SEM and GRaSP which supports reasoning over conflicting information, knowledge gaps and uncertainties across modalities and sources.

Nevertheless, EMISSOR is not restricted to a specific annotation scheme nor a specific formal model of situations. The use of JSON-LD allows seamless integration with any eKG, i.e. from raw signal, to annotation, to explicit symbolic representation. Our EMISSOR platform supports converting different datasets to a shared model that imposes alignments of spatial and temporal signals within a referential grounding framework and an episodic memory with understanding. The framework combines modalities, generates segmentation for each, and creates referential grounding and corresponding triple representations that capture identities and relations. We furthermore provide an annotation tool to create gold annotations for referential grounding, both from dynamic streams and manually designed scenarios using controlled static data.

3 EMISSOR: design and specification

In this section, we describe the different data layers of our model and their interrelations. We summarise the design desiderata for our model as follows:

1. Support parallel unaligned streams of multimodal signals
2. Detect sequences of segments within signals grounded in time and space
3. Allow segment alignment, overlap, and disjointedness across modalities
4. Model situated-references in segments to unique identities in a Knowledge Graph
5. Model causal coherent relations between subsets of segments across modalities
6. Model physical and virtual real-world contexts
7. Model streams of signals as a transparent cumulative symbolic interpretation of experiences
8. Provide an episodic memory of situated references and interpretations in a knowledge graph that supports reasoning

To meet these requirements, we use GRaSP as a referential framework to connect segments in signals to identities in an eKG. These identities are individual people, objects, and places, or relations and properties of these. In the former case, identities are represented as instances through their URIs, possibly with names as labels and instances of a

\(^{17}\)json-ld.org/
particular type. In the latter case, constellations of instances are interpreted as relations or properties represented by RDF triples. In addition, media streams are stored as separate data files for each modality. Separate JSON-LD files are provided for each modality that 1) divide this modality signal into segments grounded in time and space and 2) annotate these segments as representing identities and relations registered in the eKG. A stream of media interpretations emitting triples over time then results in the cumulative growth of an eKG representing an episodic memory. With the use of JSON-LD elements, signal metadata can be mapped to our referential framework and included in the eKG.

Motivating example Let us consider a simple example (1). A face detector module detects a human face in a video frame at time $t1$ which results in a box segment. Face recognition cannot recognize this person, so it is identified as a new instance of the type PERSON. Next, at $t2$, another face is detected as a box segment. This face is identified as a known person in the eKG: a URI with the name ‘Carl’. At $t3$ a speaker is detected whose voice is mapped to the same identity of ‘Carl’. The speaker says “Do you know my daughter Carla?” A text understanding module processes the text representation of the audio at $t3$. Words such as “I” and “me” are linked to the same URI as the speaker, while “Carla” will be mapped to another identity (not yet visually grounded) and related to “Carl” as his daughter. Through contextual reasoning, the model may ground the first unidentified face perception to this newly created identity of “Carla” in hindsight. Let us consider an alternative variation on this scenario in which the detected speaker at $t3$ is mapped to “Alice” rather than “Carl”. Alice says, “That is Carl and his daughter Carla”. The deictic references to “Carl” and “Carla” and the pronominal reference “his” can only be resolved by combining the earlier perceptions with the semantics of this utterance. On the other hand, in both scenarios, “being a daughter” is knowledge that cannot be concluded from visual and audio signals and is solely conveyed by interpreting the semantics of the utterance. This demonstrates that neither audio-visual nor textual segments contain all the information needed to come to the correct interpretation: language utterances due to their referential nature rather than being descriptive, audio-visual segments due to their limitation to signal social, conceptual, and cultural framing.

Example 1: Carl - Carla scenario

3.1 Model description

Figure 1 shows an overview of our representation’s different data layers in terms of an entity-relationship model. For grounding data, we define different layers for segments, rulers, and containers. Segments can have complex causal coherence relations across modalities. Since the segmentation of these modalities is done separately, we need temporal and spatial containers to order and connect segments across modalities, following the principles of TimeML (Pustejovsky and Stubbs, 2011) and VOXML. The temporal and spatial containers form the basis for constituting potential causal relations (forward and backward and across distances). Therefore, each container will consist of a ruler that defines the granularity of segments across modalities (e.g., a sequence or region). The ruler positions the segments relative to each other and makes them conditional for defining relations and for predictive models.

A scenario (bottom left) is an instance of a context in a specific time and space and acts as a container for parallel streams of multimodal signals, which are divided into segments. Scenarios can have specific attributes to qualify them, including names, overall scenario type and location, the purpose or intention of (specific) participants. When segments get annotated, mentions (linguistic) or perceptions (video or audio) are created, pointing to one or more segments and an annotation value.Annotations can be added freely and there are no restrictions on the values for annotations in the JSON structure. Figure 1, shows a few examples of typical annotation values such as Face for boxes in images, Tokens in texts or NamedEntity for Named Entity expressions. However, EMISSOR additionally uses JSON-LD to also support the direct linking of interpretations to the eKG in which people, objects and situations are modeled through explicit URIs. We therefore allow explicit URIs as annotation values to ground the segments to these identities. In that case, a box segment, a
name or pronoun in the text is annotated with the unique URI of a person rather than a conceptual label as a value.

In Figure 1 due to limits of space, we only show a place holder for the eKG as a formal model of situations at the right side. This placeholder stands for any ontological model and its population with instances. In practice, we use a wide range of ontologies to model situations as populations of instances, as described in (Vossen et al., 2019b). Note that EMISSOR allows for both types of annotations next to each other, e.g. a segment can be annotated as a human face without a specific identity but the same segment can have an additional annotation with the URI from the eKG.

**Scenario structure** We consider an interaction as a scenario. Scenarios are organized in folders on disk. Within a scenario folder, we store the source data as separate files in a modality subfolder, e.g. text, video, image, audio. Furthermore, one JSON-LD file per modality defines the metadata and the segments present for each signal in the modality. In addition to these segments, there may be lists of mentions or perceptions as JSON-LD elements. Each mention or perception specifies a range of segments (at least one) and the interpretations as annotations representing the instances and concepts in Figure 1. Next to the modality JSON-LD file, a specific folder contains the RDF triples extracted from the annotated signals. For example, an utterance in a conversation may mention somebody’s age, which yields an RDF triple with the person’s URI as the subject, the has-age property, and the actual age as a value.

Finally, there is a separate JSON file with metadata on the complete scenario. This scenario JSON defines the temporal and spatial ruler within which the scenario is located (date, begin and end time, geo-location, place-name), the interacting participants (e.g., the agent and the human speaker(s)), and any other people and objects that participate in the scene. The specification of participants and props can be based on the instances from the eKG. This scenario JSON file has the same name as the folder name of the scenario.

**Modalities** The different modalities are represented in parallel streams of signals that are aligned by temporal and spatial rulers in the containers (Figure 2). We currently support text, audio, and visual modalities enriched by the knowledge layer as extracted by annotations.

Within each modality, a signal is broken down into *segments* positioned relative to the temporal and spatial ruler through begin and end points or box coordinates respectively. The granularity can vary but depends on the minimal unit of the rulers. Figure 2 shows an example of a scenario with layers for these four modalities with a temporal ruler on the horizontal axis. In this scenario, a person, “Carl”, tells a robot, “Leolani”, that he cannot find a pillbox. The robot spots the box under the table and communicates this to Carl, who confirms finding it. At every turn in the conversation, we see interaction data as segments (bars) aligned through the temporal ruler and its corresponding subgraphs generated from the interpretations as added to the eKG. The triples in the subgraph not only contain the representations for the participants and the pillbox.
Annotations and mentions Any segment can be annotated, which results in mentions or perceptions added to the JSON-LD file for the specific media. Mentions define a relation between text segments and interpretations, whereas perceptions relate audiovisual segments to interpretations. Each annotation has the following attributes: 1) type: kind of annotation; 2) value: the actual interpretation (e.g., label, reference); 3) source: software or person that created the annotation; 4) timestamp: when the annotation was created. We can have any number of segments with any number of annotations defined per mention/perception. Furthermore, annotations can be added on top of other annotations, following the Layered Annotation Framework (Ide and Romary, 2007). Finally, annotations of segments in different modalities with the same identifier will automatically create cross-modality co-reference; for example in the CarLani scenario, perception of the pillbox and its mention in the utterances are mapped to the same instance URI in the eKG.

Identities, properties, and relations The core idea is to create a mapping between a segment of a signal (e.g., a bounding box in an image or an offset position and length in a text) and the signal’s interpretation (e.g., a person’s face or a pronoun). Through referential grounding, we generate triples expressing properties and relations across different modalities. Shared identifiers (URIs) aggregate these properties and relations, resulting in a world model over time. In Figure 2, this is demonstrated by the sequence of subgraphs at the top showing different states of interpretation going from lack of knowledge about the location of the pillbox (negative polarity) to the perception and having it in possession (positive polarity). The triples stored in an eKG likewise reflect this accumulation over time, while each triple is still grounded to a segment in a modality.

As explained in previous work (Vossen et al., 2018, 2019a), our framework focuses on storing information related to episodic experiences and their interpretations as perspectives that agents have. By nature, our framework is flexible in dealing with incomplete or contradicting information and can reason over knowledge with uncertainty while considering the sources’ trustworthiness. For the scenario in Figure 2, the model represents two realities at the same point: “Carl” not knowing the location and ”Leolani” knowing the location. Querying the model for the location of the pillbox at that time generates an answer according to ”Leolani”. Reasoning is thus not only used to derive knowledge or answer factual questions, but also to evaluate the quality of the knowledge itself. Agents can use such qualitative evaluations to formulate strategies and actions to improve knowledge states.

Following the principles of Linked Data, our framework reuses existing ontologies for provenance (PROV-O), text processing (NAF), event (SEM) and perspective modeling (GRaSP, GAF). The usage of RDF allows us to integrate information from other existing open Knowledge Graphs, e.g., WikiData or DBpedia, to include prior knowledge. Using JSON-LD elements in our representation enables us to directly attach the referential grounding to the eKG by mapping elements of our JSON structure described in Figure 1 to elements of the underlying ontologies of the eKG without losing the lightweight representation of plain JSON.

4 Conclusions In this paper, we described eight desiderata for the representation of multimodal interactions in collaborative contexts. We argued that existing
representations do not satisfy all these desiderata and therefore presented EMISSOR for referential interpretations of multimodal interactions to yield a Knowledge Graph as an explicit episodic memory of the experiences (eKG). EMISSOR combines light-weight JSON-LD representations for sequential media with semantic web-based RDF models of interpreted worlds. Through this we model cumulative growth of knowledge and information in the eKG as a result of processing multimedia streams over time. EMISSOR is designed to address all eight desiderata. It enables to create and compare recordings, annotations and interpretations of interactions in real-world contexts. This allows researchers to more easily share experiments and compare the interactions across different experiments, regardless of the specifics of agent systems or humans that participate in the experiment. Our model, software and converted data sets are available\textsuperscript{18} according to the Apache open source license. Our release includes an annotation tool to create scenarios manually or inspect and annotate any recording of data.

As next steps, we develop more tooling and automatic linking of multimodal segments to identities. We will also provide more data either by converting existing public data to our framework or by rendering data through our robot platform. So far we focused on grounding segments in temporal containers but not yet in spatial containers. Our interactions do not include motion and navigation. In future work, we hope to include spatial grounding and reasoning. Finally, we will include an evaluation framework for analysing system performance in relation to 1) qualitative properties of the interaction, 2) goals and intentions following a reinforcement learning approach and 3) by evaluating the resulting eKG.

5 Acknowledgements

This research was funded by the Vrije Universiteit Amsterdam and the Netherlands Organisation for Scientific Research (NWO) via the Spinoza grant (SPI 63-260) awarded to Piek Vossen, the Hybrid Intelligence Centre via the Zwaartekracht grant (024.004.022) and the research project (9406.D1.19.005) Communicating with and Relating to Social Robots: Alice Meets Leolani.
References


Appendix A: Annotation tool and example data sets

A Example Data Sets

Any multimodal interaction data can be represented and annotated in the EMISSOR annotation format with minimal effort. We released EMISSOR representations and annotations of some popular public data sets (e.g. MELD and IEMOCAP), together with the scripts to convert them from their original formats. In the near future, we will add scripts for other popular data sets. In addition to the conversion scripts, we also created scripts for segmentation of modalities, such as bounding boxes for objects, faces and text tokenization with named entity detection. An additional baseline script resolves the identities of faces and entities against an eKG by selecting the first matching name. These scripts prepare any video recording or collection of multimedia data for annotation in the annotation tool described in Section B. In the future, we replace the baseline scripts with SOTA modules for resolving referential ambiguity.

We also created our own data set called CarLani directly rendered from interacting with our robot platform. Figure 2 shows an example dialogue of three utterances from this data set between the human (Carl) and the robot (Leolani), assuming the context of a care taking robot in an elderly home.

By running interactions through our robot platform with humans in a physically perceived world, the multimodal data is automatically grounded in the knowledge graph according to the EMISSOR framework. This will automatically generate rich referential relations between mentions and perceptions with identities, within a functional communicative contexts. These can be analysed, evaluated and adapted to gold annotations for training and testing.

B Annotation Tool

Along with the proposed data representation, we are developing a GUI tool capable of reading EMISSOR data representations (with or without annotations). The purpose of the tool is a first inspection of data sets by providing a comprehensible visualisation of the signals in different modalities, their grounding to the temporal (and spatial) containers, as well as their interpretations, including segment alignments, situated references and explicit semantic representations. Second, it allows modification of the aforementioned properties for a given data set, e.g. to add gold annotations, perform corrections or add additional interpretations. Third, gold scenarios for a given task or problem can be created manually from scratch without the need for an actual agent implementation.

Besides conversion issues to other data representations\(^\text{19}\), existing tools like e.g. Anvil\(^\text{20}\) (Kipp, 2001) or Elan\(^\text{21}\) (Brugman et al., 2004) only ground the conversation to speakers, audio, faces, gestures but do not ground referential expressions to the situation. Our tool focuses on segmentation and grounding to mediate between the media data and the identities in the Knowledge Graph.

The current version of the tool supports image, audio, and text as modalities in a scenario, allows to add and remove signals to them and to position (ground) the signals on (to) the timeline of the scenario. In any situation, it is possible to create segments and annotations automatically or manually. On image signals, rectangular segments (bounding boxes) can be defined manually and annotated. Alternatively, during data preparation, boxing scripts can be used to automatically generate bounding boxes beforehand. Text signals can be automatically tokenized using scripts beforehand as well. The tool then allows for token selections to be annotated. The tool also provide choices for reference linking to known (listed) entities as annotation values. These entities can be taken from the eKG or from any other registration. In addition, you can create new identities directly in the tool through annotations, as well create triples that express properties or relations between entities. The URIs and triples created during the annotation can then be added as gold knowledge to the eKG a posteriori.

Finally, the annotation tool can be used to create scenarios manually in a very controlled way. Researchers can store images and conversations in the corresponding media folders manually and next use

\(^{19}\) Anvil supports DIAML and stores conversational units in sequences with name identifiers and time stamps for an associated video file. ELAN stores conversations in EAF (Eudico Annotation Format), which is a propriety XML format.

\(^{20}\) www.anvil-software.org

\(^{21}\) www.mpi.nl/corpus/html/elan/
Figure 3: Tool for visualizing and annotating EMISSOR

the tool to place them in the proper order. In the near future, we will add a function to play such a scenario as well following the temporal specification. Currently, the user can play it by moving forward manually.

Appendix B: CarlLani Example Data from redacted source

In the following we include an excerpt of the CarlLani data set as the original source was redacted from the submission for anonymization purposes. For space and readability reasons text.json and image.json metadata files are shortened by removing part of the signals and/or mentions. Also the context referenced in the JSON-LD @context element is included.

Scenario structure

|  |- carl-robot/
|   |  |- audio/
|   |  |  |- carl-robot-000_frame0_0.wav
|   |  |  |- carl-robot-000_frame30_1000.wav
|   |  |  |- carl-robot-000_frame60_2000.wav
|   |  |  |- ....
|   |  |  |- image/
|   |  |  |  |- carl-robot-000_frame0_0.jpg
|   |  |  |  |- carl-robot-000_frame30_1000.jpg
|   |  |  |  |- carl-robot-000_frame60_2000.jpg
|   |  |  |  |- ....
|   |  |  |  |- rdf/
|   |  |  |  |  |- episodic_memory.trig
|   |  |  |  |  |- statement1.trig
|   |  |  |  |  |- objectdetection1.trig
|   |  |  |  |  |- statement2.trig
|   |  |  |  |  |- statement3.trig
|   |  |  |  |  |- text/
|   |  |  |  |  |  |- carl-robot.csv
|   |  |  |  |  |  |- video/
|   |  |  |  |  |  |  |- carl-robot-000_frame0_0.mp4
|   |  |  |  |  |  |  |- carl-robot-000_frame30_1000.mp4
|   |  |  |  |  |  |  |- carl-robot-000_frame60_2000.mp4
carl-robot.csv
speaker,utterance,time
Carl,"I need to take my pills, but I cannot find them.",0
Leolani,"I found them. They are under the table.",3933
Carl,"Oh! Got it. Thank you.",7133

carl-robot.json

```json
{
    "@context": "http://emissor.org/jsonldcontext.jsonld",
    "type": "Scenario",
    "id": "carl-robot",
    "context": {
        "agent": "robot_agent",
        "objects": [],
        "persons": [],
        "speaker": {
            "@context": "http://schema.org/docs/jsonldcontext.jsonld",
            "id": "bc913d64-a597-4876-a3fe-fe47472cd274",
            "type": "Person",
            "birthDate": "1995-04-09T20:00:00Z",
            "gender": "Male",
            "name": "Carl"
        }
    },
    "ruler": {
        "type": "TemporalRuler",
        "container_id": "carl-robot",
        "end": 11133,
        "start": 0
    },
    "signals": {
        "image": "/image.json",
        "text": "/text.json"
    }
}
```
"bounds": [2830, 241, 3034, 521],
}
]
}
}]

{
    "@context": "http://emissor.org/jsonldcontext.jsonld",
    "type": "ImageSignal",
    "id": "88a31791-4410-45f2-b611-f61cb4d321ff",
    "files": [
        "image/carl-robot-000_frame30_1000.jpg"
    ],
    "modality": "image",
    "time": {
        "type": "TimeSegment",
        "container_id": "carl-robot",
        "start": 1000,
        "end": 1033
    },
    "ruler": {
        "type": "MultiIndex",
        "container_id": "88a31791-4410-45f2-b611-f61cb4d321ff",
        "bounds": [0, 0, 3840, 1080]
    },
    "mentions": [
        {
            "type": "Mention",
            "id": "92af1ea9-41d4-421e-b3f4-7955e71a1a97",
            "annotations": [
                {
                    "type": "Annotation",
                    "source": "machine",
                    "timestamp": 1000,
                    "type": "person",
                    "value": {
                        "type": "Face",
                        "instance": {
                            "@context": "http://schema.org/docs/jsonldcontext.jsonld",
                            "@id": "bc913d64-a597-4876-a3fe-fe47472cd274",
                            "type": "Person",
                            "birthDate": "1995-04-09T20:00:00Z",
                            "gender": "Male",
                            "name": "Speaker"
                        },
                        "age": 21,
                    }
                }
            ]
        }
    ]
}
"gender": "male",
"faceprob": 1.0
}
]
"segment": [
{
"type": "BoundingBox",
"container_id": "88a31791-4410-45f2-b611-f61cb4d321ff",
"bounds": [2831, 235, 3036, 514]
}
]
}]}}, .....]
"@context": "http://emissor.org/jsonldcontext.jsonld",
"files": [
"text/carl-robot.csv#2"
],
"id": "2142b6d8-4cda-481b-a056-1b6d874da648",
"mentions": [

  {"type": "Mention",
   "id": "c851ca48-81b6-44fe-a772-9f62840ca2f6",
   "annotations": [
     {"source": "annotation_tool",
      "timestamp": 1616442473,
      "type": "token",
      "value": {
        "id": "d7770947-0be5-413f-9c1e-4e9d130e6a41",
        "container_id": "d7770947-0be5-413f-9c1e-4e9d130e6a41",
        "type": "AtomicRuler",
        "type": "Token",
        "value": "Oh"
      }
    },
    {"segment": [
      {"container_id": "2142b6d8-4cda-481b-a056-1b6d874da648",
       "start": 0,
       "stop": 2,
       "type": "Index"
     }
    ],
    {"source": "annotation_tool",
     "timestamp": 1616442473,
     "type": "token",
     "value": {
       "id": "f87af3f6-11c4-486c-bd60-aeedd4377bb7",
       "container_id": "f87af3f6-11c4-486c-bd60-aeedd4377bb7",
       "type": "AtomicRuler",
       "type": "Token",
       "value": "."
     }
    },
    {"segment": [
      {"container_id": "2142b6d8-4cda-481b-a056-1b6d874da648",
       "start": 21,
       "stop": 22,
       "type": "Index"
     }
    ],
    "modalitiy": "text",
    "ruler": {
      "container_id": "2142b6d8-4cda-481b-a056-1b6d874da648",
      "start": 0,
      "stop": 22,
      "type": "Index"
    }
  }
]
<table>
<thead>
<tr>
<th>Line</th>
<th>Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>`{</td>
</tr>
<tr>
<td>2</td>
<td>&quot;@context&quot;: {</td>
</tr>
<tr>
<td>3</td>
<td>&quot;@base&quot;: &quot;<a href="http://experiment.my/">http://experiment.my/</a>&quot;,</td>
</tr>
<tr>
<td>4</td>
<td>&quot;@vocab&quot;: &quot;<a href="https://emissor.org/emissor#">https://emissor.org/emissor#</a>&quot;,</td>
</tr>
<tr>
<td>5</td>
<td>&quot;type&quot;: &quot;@type&quot;,</td>
</tr>
<tr>
<td>6</td>
<td>&quot;id&quot;: &quot;@id&quot;,</td>
</tr>
<tr>
<td>7</td>
<td>&quot;emissor&quot;: &quot;<a href="http://emissor.org/emissor#">http://emissor.org/emissor#</a>&quot;,</td>
</tr>
<tr>
<td>8</td>
<td>&quot;grasp&quot;: &quot;<a href="http://groundedannotationframework.org/grasp#">http://groundedannotationframework.org/grasp#</a>&quot;,</td>
</tr>
<tr>
<td>9</td>
<td>&quot;container_id&quot;: {&quot;@type&quot;: &quot;@id&quot;},</td>
</tr>
<tr>
<td>10</td>
<td>&quot;signal&quot;: &quot;@nest&quot;,</td>
</tr>
<tr>
<td>11</td>
<td>&quot;Mention&quot;: &quot;grasp:Mention&quot;</td>
</tr>
<tr>
<td>12</td>
<td>}</td>
</tr>
<tr>
<td>13</td>
<td>}</td>
</tr>
</tbody>
</table>
@prefix robotContext: <http://emissor.org/robot/context/> .
@prefix xml: <https://www.w3.org/TR/2002/07/owlib> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@prefix prov: <http://www.w3.org/ns/prov#> .
@prefix gaf: <http://groundedannotationframework.org/gaf#> .
@prefix gaf: <http://groundedannotationframework.org/grasp> .
@prefix sem: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix gaf: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix prov: <http://www.w3.org/2002/07/owl#> .
@prefix wd: <http://www.wikidata.org/entity/> .
@prefix grasp: <http://groundedannotationframework.org/grasp> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@prefix nd: <http://www.w3.org/2001/XMLSchema#> .
@prefix prov: <http://www.w3.org/ns/prov#> .
@prefix gaf: <http://groundedannotationframework.org/gaf> .
@prefix gr: <http://groundedannotationframework.org/grasp/> .
@prefix gaf: <http://groundedannotationframework.org/gaf> .
@prefix prov: <http://www.w3.org/ns/prov#> .
@prefix xhtml: <http://www.w3.org/1999/xhtml#> .
@prefix prov: <http://www.w3.org/ns/prov#> .
@prefix gr: <http://groundedannotationframework.org/grasp/> .
@prefix prov: <http://www.w3.org/ns/prov#> .
@prefix xhtml: <http://www.w3.org/1999/xhtml#> .
@prefix prov: <http://www.w3.org/ns/prov#> .
@prefix gr: <http://groundedannotationframework.org/grasp/> .
@prefix prov: <http://www.w3.org/ns/prov#> .
@prefix xhtml: <http://www.w3.org/1999/xhtml#> .

```prolog
robotWorld: Instances {
    robotWorld: lani a gaf: Instance, robotMu: robot ;
    rdfs: label "lani" ;
    robotWorld: pills a gaf: Instance, robotMu: object ;
    rdfs: label "pills" ;
    gaf: denotedIn robotTalk: chat1, utterance2, char0=39 .
    rdfs: label "pills-277239" ;
    robotMu: id "277239"^^xsd:string ;
    gaf: denotedIn robotTalk: visual1, detection2, pixel10=3 ;
    eps: hasContext robotContext: context212127 .
    robotWorld: table a gaf: Instance, robotMu: object ;
    rdfs: label "table" ;
    gaf: denotedIn robotTalk: chat1, utterance2, char0=39 .
    rdfs: label "table-208510" ;
    robotMu: id "208510"^^xsd:string ;
    gaf: denotedIn robotTalk: visual1, detection2, pixel10=3 ;
    eps: hasContext robotContext: context212127 .
}
```

```prolog
robotTalk: Interactions {
    robotWorld: Netherlands a robotMu: location, sem: Place, robotMu: country ;
    rdfs: label "Netherlands" ;
    robotWorld: Gelderland a robotMu: location, sem: Place, robotMu: region ;
    rdfs: label "Gelderland" ;
    robotWorld: Apeldoorn a robotMu: location, sem: Place, robotMu: city ;
    rdfs: label "Apeldoorn" .
    robotTalk: chat1 a sem: Event, grasp: Chat ;
    rdfs: label "chat1" ;
    robotMu: id "1"^^xsd:string ;
    sem:hasSubEvent robotTalk: chat1, utterance2 .
    robotTalk: visual1 a sem: Event, grasp: Visual ;
    rdfs: label "visual1" ;
    robotMu: id "1"^^xsd:string ;
    sem:hasSubEvent robotTalk: visual1, detection2 .
    robotTalk: chat1, utterance2 a sem: Event, grasp: Utterance ;
    rdfs: label "chat1, utterance2" ;
    robotMu: id "2"^^xsd:string ;
    sem:hasActor robotFriends: lani .
    robotTalk: visual1, detection2 a sem: Event, grasp: Detection ;
    rdfs: label "visual1, detection2" ;
    robotMu: id "2"^^xsd:string ;
    sem:hasActor robotInputs: front-camera .
    rdfs: label "front-camera" .
    rdfs: label "lani" ;
    robotContext: home a robotMu: location, sem: Place ;
    rdfs: label "home" ;
    robotMu: id "251375"^^xsd:string ;
    robotContext: context212127 a eps: Context ;
    rdfs: label "context212127" ;
    robotMu id "212127"^^xsd:string ;
    eps: hasDetection robotWorld: pills-277239, robotWorld: table-208510 ;
    sem:hasBeginTimeStamp robotContext: context212127 .
    sem:hasEndTimeStamp robotContext: context212127 .
    eps: hasLocation robotContext: home .
    robotContext: 2021-03-12 a sem: Time, time: Date Time Description ;
    rdfs: label "2021-03-12" ;
    time: day "12"^^xsd:integer .
```

robotWorld:Claims {
  robotWorld: lani_sense_front-camera a gaf:Assertion, sem:Event;
  rdfs:label "lani_sense_front-camera" .
  robotWorld: lani_know_lani a gaf:Assertion, sem:Event;
  rdfs:label "lani_know_lani" ;
  owl:sameAs robotWorld:lani .
  rdfs:label "pills_locatedUnder_table" ;
  gaf:denotes robotWorld:pills; robotWorld:table ;
  grasp:wasAttributedTo robotFriends:lani ;
  grasp:hasAttribution robotTalk:pills_locatedUnder_table CERTAIN−POSITIVE−NEUTRAL−NEUTRAL .
}

robotTalk:Perspectives {
  robotTalk: chat1 Utterance2, char0−39 a gaf:Mention, grasp:Statement ;
  rdfs:label "chat1Utterance2_char0−39" ;
  rdf:value "I found them. They are under the table." ;
  prov:wasDerivedFrom robotTalk: chat1 Utterance2 ;
  gaf:contains Denotation robotWorld:pills; robotWorld:table ;
  grasp:wasAttributedTo robotFriends:lani ;
  grasp:hasAttribution robotTalk:pills_locatedUnder_table CERTAIN−POSITIVE−NEUTRAL−NEUTRAL .
}

robotWorld:Claims {
  robotWorld: lani_know_lani a gaf:Assertion, sem:Event;
  rdfs:label "lani_know_lani" ;
  owl:sameAs robotWorld:lani .
  rdfs:label "pills_locatedUnder_table" ;
  gaf:denotes robotWorld:pills; robotWorld:table ;
  grasp:wasAttributedTo robotFriends:lani ;
  grasp:hasAttribution robotTalk:pills_locatedUnder_table CERTAIN−POSITIVE−NEUTRAL−NEUTRAL .
}

robotWorld:Claims {
  robotWorld: lani_know_lani a gaf:Assertion, sem:Event;
  rdfs:label "lani_know_lani" ;
  owl:sameAs robotWorld:lani .
  rdfs:label "pills_locatedUnder_table" ;
  gaf:denotes robotWorld:pills; robotWorld:table ;
  grasp:wasAttributedTo robotFriends:lani ;
  grasp:hasAttribution robotTalk:pills_locatedUnder_table CERTAIN−POSITIVE−NEUTRAL−NEUTRAL .
}

robotWorld:Claims {
  robotWorld: lani_know_lani a gaf:Assertion, sem:Event;
  rdfs:label "lani_know_lani" ;
  owl:sameAs robotWorld:lani .
  rdfs:label "pills_locatedUnder_table" ;
  gaf:denotes robotWorld:pills; robotWorld:table ;
  grasp:wasAttributedTo robotFriends:lani ;
  grasp:hasAttribution robotTalk:pills_locatedUnder_table CERTAIN−POSITIVE−NEUTRAL−NEUTRAL .
}

robotWorld:Claims {
  robotWorld: lani_know_lani a gaf:Assertion, sem:Event;
  rdfs:label "lani_know_lani" ;
  owl:sameAs robotWorld:lani .
  rdfs:label "pills_locatedUnder_table" ;
  gaf:denotes robotWorld:pills; robotWorld:table ;
  grasp:wasAttributedTo robotFriends:lani ;
  grasp:hasAttribution robotTalk:pills_locatedUnder_table CERTAIN−POSITIVE−NEUTRAL−NEUTRAL .
}

robotWorld:Claims {
  robotWorld: lani_know_lani a gaf:Assertion, sem:Event;
  rdfs:label "lani_know_lani" ;
  owl:sameAs robotWorld:lani .
  rdfs:label "pills_locatedUnder_table" ;
  gaf:denotes robotWorld:pills; robotWorld:table ;
  grasp:wasAttributedTo robotFriends:lani ;
  grasp:hasAttribution robotTalk:pills_locatedUnder_table CERTAIN−POSITIVE−NEUTRAL−NEUTRAL .

Annotating anaphoric phenomena in situated dialogue

Sharid Loáiciga1  Simon Dobnik2  David Schlangen1
1Computational Linguistics, Department of Linguistics, University of Potsdam, Germany
2CLASP, Department of Philosophy, Linguistics and Theory of Science, University of Gothenburg, Sweden
{loaicigasanchez, david.schlangen}@uni-potsdam.de, simon.dobnik@gu.se

Abstract

In recent years several corpora have been developed for vision and language tasks. With this paper, we intend to start a discussion on the annotation of referential phenomena in situated dialogue. We argue that there is still significant room for corpora that increase the complexity of both visual and linguistic domains and which capture different varieties of perceptual and conversational contexts. In addition, a rich annotation scheme covering a broad range of referential phenomena and compatible with the textual task of coreference resolution is necessary in order to take the most advantage of these corpora. Consequently, there are several open questions regarding the semantics of reference and annotation, and the extent to which standard textual coreference accounts for the situated dialogue genre. Working with two corpora on situated dialogue, we present our extension to the ARRAU (Uryupina et al., 2020) annotation scheme in order to start this discussion.

1 Introduction

With the ease of combining representations from different modalities provided by neural networks, text and vision are coming together. There is a growing body of resources addressing a setting in which the visual context can be exploited to support a textual task, for example visual anaphora resolution. 1

Several corpora have been developed in the domain of vision and language (V&L), for example corpora of image captions (Lin et al., 2014; Young et al., 2014; Krishna et al., 2017), images and paragraph descriptions (Krause et al., 2017), visual question answering (Antol et al., 2015), visual dialogue (Das et al., 2017) and embodied question answering (Das et al., 2018). Through these the V&L research has progressively moved from sentence descriptions to descriptions involving utterances and conversations, therefore adding complexity to their semantic representations. In parallel to the corpora, V&L systems have been developed but of course these are limited by the complexity of the task for which the dataset has been collected. The end goal of the current research is to move to a more complex linguistic setting involving multi-party dialogue and visual representations that go beyond individual images.

Anaphora resolution has been studied both in the textual and situated dialogue domains (cf. Sukthanker et al. (2020) for an extensive survey of anaphora and coreference; (Kelleher et al., 2005; Seo et al., 2017; Kottur et al., 2018; Yu et al., 2019; Dobnik and Loáiciga, 2019)). In the textual domain, this has been formulated as a standard task with several corpora annotated uniformly for the most part, while in situated dialogue each corpus presents its own individual solution (cf. (Kelleher et al., 2005; Smith et al., 2011; Pustejovsky and Krishnaswamy, 2020)). With the increasing interest in the combination of V&L in deep learning applications, multimodal resources are increasingly used in the context of traditional textual natural language processing (NLP) tasks. As such, it makes sense to consider a common annotation strategy both for the textual and situated dialogue domains, basing it on the rich work of textual anaphora resolution standards. Doing so, we also hope to get new insights about the semantics of reference in natural language.

Situated reference resolution involves grounding linguistic expressions in perceptual representations (Harnad, 1990) or representations of actions (Roy, 2005). Anaphora resolution, traditionally a textual task, involves linking linguistic expressions referring to the same discourse entities (Stede, 2012). While challenging, the task is defined by the familiar nature of written texts: linear, planned and structured; defining thus the mechanisms and devices

1Also known as coreference resolution in the NLP domain, here we follow Poesio (2016) in our terminology.
found in them. In resources combining V&L, however, the textual part is often a dialogue or pairs of question-answers. As a result, the coreference devices differ from those found in texts and are closer to actual conversations in which people create reference to entities on the fly. This of course comes with its own challenges, but there are also some relations made easier since they can be grounded in the image.

As V&L come together, there is therefore an increased need for extending resources for the task of visual anaphora resolution. This means engaging with the challenges along two axes:

- **Dialogue:** built by two speakers who each have their own mental state and cognitive process but who are communicating through referring expressions which are projected in the same conversation. As conversations are linear (one cannot go back to the past or to the future) linguistic coreference is linear.
- **Shared physical context:** simultaneous access to an image or other perceptual context. Same as in dialogue, the speakers have different viewpoints of the scene and need to build their individual mental states representing the scene guided by visual attention. However, once a representation of a visual scene is built, reference can be made to its representations in a non-linear fashion.

We present our extension to the ARRAU (Poesio, 2004; Artstein and Poesio, 2006; Uryupina et al., 2020) annotation scheme by analysing two situated dialogue corpora: the Cups corpus (Dobnik et al., 2020) and the Tell-me-more corpus (Ilinykh et al., 2019), shown below in Figures 1 and 2 respectively. This exercise proved useful to pinpoint in what ways the purely textual document scenario is different from the domain of embodied interaction both in terms of the semantics of interaction and annotation practices.

The Cups corpus contains a conversation between two participants over an (almost) identical visual scene involving a table and cups where participants have different locations. Some cups have been removed from each participant’s view and they are instructed to discuss over a computer terminal in order to find the cups that each does not see. The ground truth of the visual scene is known as it has been artificially generated. It may take over an hour for the participants to solve the task and their activity results in free dialogue close to spoken conversations including phenomena such as clarifications, repairs, restarts and variable grammar. (The conversations are logged at a key-press level.) The Tell-me-more corpus consists of images accompanied with a short text of five complete sentences, collected by asking participants to describe the image to a friend, successively adding details in short constrained conversations. The genre of these texts is therefore mixed: in between standard text (as found in news text for example) and dialogue data which reflects the features found in conversations rather than written conventions.

These corpora are complementary as Cups gives us accurate visual ground truth information with free and unrestricted dialogue, while Tell-me-more offers a richer unrestricted image with short and task-constrained (pseudo-)dialogues.

In this paper, we discuss a number of cases from these corpora that challenge both standard language grounding annotations as well as standard anaphora annotation. This work points thus towards required future work in creating anaphora annotation schemes that can handle situated dialogue.

### 2 Related Work

Pointing to the inability of NLP tools to handle the textual part in situated dialogue, early works had described the need to ground the dialogue in the image in a manner informed by linguistics (Byron, 2003).

As content develops in a text, entities are introduced and re-mentioned, establishing discourse referents. The context is provided by the document and no extra-linguistic reference is needed for resolving the reference to an entity (Karttunen, 1969). In situated dialogue, on the other hand, the visual modality brings the extra-linguistic context as a source of referents. Here, resolving references to entities can be thus achieved by either looking at the picture or relating to the information that has been said previously in the discourse. Both of these processes happen simultaneously and therefore their interaction must be explained by theories of cognitive processing related to attention and memory (Kelleher and Dobnik). However, in order to understand both processes and their interaction we need to disentangle them. Extending the anaphora annotation paradigm is thus the best bet although not a lot of work exists in this area.
Textual coreference Annotated data for the coreference resolution task has mainly focused on news texts and concrete nouns, excluding reference to events and other coreferential relations such as bridging, deixis, and ambiguous items well documented in the linguistic literature but deemed infrequent or too difficult to process (Poesio, 2016). In contrast, there is a growing body of literature interested in phenomena beyond the nominal case (Kolhatkar et al., 2018; Nedoluzhko and Lapshinova-Koltunski, 2016), resulting in new annotated corpora (Lapshinova-Koltunski et al., 2018; Zeldes, 2017; Uryupina et al., 2020), although smaller in
size than OntoNotes (Pradhan et al., 2007), the largest and most used coreference corpus in the field.

Moreover, as a product of this year’s edition of the CRAC and CODI workshops, a shared task on anaphora resolution in dialogues has been proposed. This will undoubtedly result in additional corpora annotated with the standards used for the coreference resolution task.

Visual coreference Coreference work based on the popular VisDial dataset (Das et al., 2017) targets only a limited set of referential expressions, partly because it relies on automatic tools (Kottur et al., 2018; Yu et al., 2019), which are known to be problematic with this genre. With a focus in grounded human interaction, there are corpora whose textual part comprises question answer pairs (Antol et al., 2015; Goyal et al., 2017). Those, however, are short in nature, with few opportunities for re-mention of the different objects in the image and hence coreference. Last, corpora designed towards navigation and location involve considerable dialogue interaction between instruction giver and instruction follower which include examples of coreference. For example, the SCARE corpus (Stoia et al., 2008) provides natural interactions, it has been audio recorded and then transcribed, the conversations are long and there are frequent referring expressions (it is hard to understand transcribed dialogues on its own), but overall the size of the corpus is small. Thomason et al. (2019) present a corpus of 2050 short human-human interactions in a virtual environment collected with crowd-sourcing.

Referring expressions generation The goal in this area is to generate referring expressions over several turns of conversation in a natural and non-repetitive way to the same (or different) grounded objects following principles of communicative discourse (Takmaz et al., 2020). Here, the PhotoBook dataset (Haber et al., 2019) is used. Our work is complementary to these approaches as it focuses on the interpretative rather than generative aspects of reference and coreference.

3 The ARRAU annotation scheme

Deeply rooted in linguistic theory, the ARRAU corpus annotation scheme is particularly well-suited for annotating situated dialogue. Indeed, its annotation scheme was designed to accommodate different genres, including news, dialogue and narrative texts, and in consequence anaphoric phenomena beyond the nominal standard case typically found in other coreference corpora (Uryupina et al., 2020).

The dialogue genre has its own idiosyncrasies not covered by annotation schemes designed for news text, for example collaborative completions giving way for discontinuous markables (Uryupina et al., 2020), and more pronouns including deictics (Müller, 2007). The annotation scheme also includes guidelines for bridging reference, a much less studied type of reference but very commonly used in the Tell-me-more corpus discussed here. ARRAU is also known for containing annotations for both referring and non-referring expressions. Most coreference corpora focus on identity anaphora, meaning that only multiple mentions of the same discourse entity are annotated, leaving out those mentioned only once, also known as singletons. The large OntoNotes corpus, for instance, does not include annotations of singletons or expletives.

In the next section, we describe the general ARRAU annotation scheme along with our proposed adaptations. With the goal of moving towards general guidelines for the situated dialogue genre, the extensions we present target the common challenges of our two corpora.

4 Annotating situated dialogue

4.1 Mention identification and object detection

The first step is identifying the referring expressions or mentions to annotate. In ARRAU, all noun phrases are considered, marking the complete phrase with all its modifiers and not just its head. This includes noun phrases which are non-referring such as pleonastics and also noun phrases not re-mentioned later in the text. The mentions also include personal pronouns and demonstrative pronouns used as deictics (to refer back to non-nominal antecedents).

We also consider all noun phrases, including pronouns and deictics as mentions. For Cups, we created a simple NP chunker based on the regular
expression method (Bird et al., 2009) with moderate success: a manual annotation of one of the documents showed an error rate of about 30% (295 errors out of 1030 identified chunks). In contrast, for Tell-me-more we had annotators identify the NPs completely by hand.

Compared to ARAU, the noun phrases in these corpora are rather simple, without a lot of modifiers. However, this does not mean that mention identification is straightforward as complex noun phrases with embedded markables such as the blue cup with a white handle do arrive. Consider also the blue cup to the left of the red cup, where a particular cup is referred to by taking another cup as a landmark: is it the left or the red cup or the left of the red cup which should be considered for re-mention?

Akin to the mention identification, the image in the multimodal corpora is processed in order to detect objects. In Cups, we have the ground truth of the scenes from which participants views have been generated. All the objects and geometrically defined regions are assigned a predefined ID as shown in Figure 1. In Tell-me-more, the object labels are part of the underlying ADE20K data (Zhou et al., 2017), extracted using tools from Schlangen (2019). Here, an automatic object classifier may not detect all the objects in the scene or assign them different labels than participants use when referring to them in the dialogue.

4.2 Characterisation of the mention

The morphosyntactic properties of the mention are annotated, including gender (female, male, neutre), number (singular, plural, mass) and person (1st, 2nd, 3rd), and its semantic type (person, animate, concrete, space, time, plan (for actions), abstract, or unknown). We include all these categories used in ARAU.

In addition, we have also extended them in order to include a cardinality attribute. This accounts for a common strategy of grouping things in order to refer to them collectively. In other words, objects can be created dynamically as the dialogue progresses. For example, when a speaker refer to the blue ones, these are not all the blue cups in the scene but a particular set of blue objects that were grouped at that point of the dialogue and which can then be subsequently re-mentioned.

The cardinality attribute has the values unique and group. The first refers to objects represented by a single individual entity while groups refer to entities composed by several objects. Note that group is different from the mass number attribute in that mass nouns are usually singular. The value group refers to cases where the speaker decided to refer to a specific region of the image containing several entities together, for instance green curtains in sentence 4 in (1).

(1) 1. I see a picture of an entertainment room. 2. There is a round table in the foreground and a fussball table in the middle of the room, as well as a pool table further back. 3. There is a sitting area with chairs facing a television set. 4. The room has several windows with green curtains. 5. The floors are made of a brown tile.

4.3 Characterisation of the reference

As mentioned, ARAU covers a broad range of anaphoric relations including both non-referring and referring noun phrases. Distinguishing between these two is non-trivial, and research around ARAU have argued in favour of annotating both types (Poesio, 2016; Yu et al., 2020).

4.3.1 Non-referring

This includes mentions with a specific syntactic or semantic function: predication, expletive, idiom, incomplete or fragmentary expression, quantifier, and coordination. The last two are, by the authors own admission, controversial. Following ARAU, we annotate all types of non-referential mentions.

4.3.2 Referring

If a mention is identified as referring, then its information status needs to be annotated as discourse-new or discourse-old; discourse-old information needs to point to an antecedent.4 This distinction signals whether an entity is mentioned a first or subsequent time, shaping the reader’s discourse model of that particular discourse entity (Stede, 2012).

---

4 An antecedent can always be annotated as ambiguous if a clear entity cannot be identified for a particular mention.
Referring mentions yield coreference chains – the sequence of mentions pointing to a same entity in a text – a central construct in the coreference resolution domain. Built on top of the document as a unit, this notion relies on and in turn informs theories about accessibility hierarchy and salience of entities (Ariel, 1988, 2004; Grosz et al., 1995).

These theories are based on the observation that some forms are used to introduce entities and some others to refer to them: some entities are discourse-new and some are discourse-old. In situated dialogue, the image provides an additional context and source of referents, but it does not follow that the status of subsequent mentions is old. In the example (2) below, the fact that the discourse starts with *It* is licensed by the image and this source of reference should be accounted for differently in the annotation than a genuine discourse-old case such as the *it* in sentence 2.

(2) 1. It’s a well-lit kitchen with stained wooden cupboards. 2. There’s a microwave mounted over the stove, which has a red tea kettle on it. 3. The appliances are black and stainless steel in the kitchen. 4. The countertops look like they’re black granite. 5. The window has sunlight streaming in and it’s very brightly light.

In order to address these cases in the *Tell-me-more corpus*, we consider them discourse-old. Very importantly, in order to keep them distinct from genuinely old information in the discourse, we introduced a new value task for the antecedent (hence a discourse-old entity can have an antecedent which is a phrase, a segment, or the task). Our reasoning is that although the pronoun *It* does not have an antecedent in the text, it appears in the first position of the first sentence because the speaker was probably referring back to the *the image* in the instructions “Describe the image to a friend...”.

In dialogue as found in the *Cups corpus*, on the other hand, references can be established either relative to utterances of a particular speaker or across utterances of different speakers, and in situated dialogue, references can also be established to the objects in the scene. This leads to another notable extension to the annotation scheme: the grounding of the entities to the image (Section 4.4).

4.3.3 Bridging

An understudied referential relationship also included in the *ARRAU* guidelines is bridging, i.e. an associative relationship between two mentions (Versley et al., 2016). When the status of a mention is either *new* or *old*, it is possible to annotate if the mention is a related object of some other entity. Here we follow the simplified scheme from Artstein and Poesio (2006):

- **Part**: “An object that stands in a part-of relation to an object previously mentioned.”
- **Set**: “Relations that hold between a set and its elements, or between a set and a subset.”
- **Other**: “Expressions containing the word other and referring to a second object of the same type as an object already mentioned.”
- **Miscellaneous**: “Clear cases of bridging references that do not fall into any of the categories above.”

The *Tell-me-more corpus* is rich in examples of bridging. Since the corpus uses pictures of different rooms in a house, after a room is introduced, typically a series of objects belonging to that room follow, creating many opportunities for using a bridging reference mechanism. For instance, image your surprise if the second sentence of example (3) started with *the toaster* instead of *the bed*. Coherence will be immediately broken.

(3) 1. This is a bedroom with a twin sized bed in it. 2. The bed has a blue bag laying on it and a green bad on the floor at the foot of the bed. 3. There is a nightstand aside of the bed with a water bottle on it. 4. There is an arched closet space on one wall and an arched shelving area too. 5. There is a small lamp attached to the wall at the head of the bed.

4.4 Grounding and referentiality

In spoken discourse people try their best to ground the references so they make sure they understand each other. To do so, they rely on the mechanisms of memory and attention (Kelleher and Dobnik). Memory controls how long objects referred to and objects perceived are cognitively salient in the mind of an agent, while attention controls the ratio of information that becomes salient coming from perception vs the amount of information coming from cognitive control of an agent (Lavie et al., 2004).

Most entities annotated as concrete references can be grounded to the image easily. Following the *ARRAU-trains* annotation closely, we have added an attribute on-image with values yes/no. If the value is yes, then the attribute bounding-box with values yes/no needs to be annotated as well. The idea here is to distinguish between grounded entities detected by the object detector, and those that although visible do not have a bounding box or predefined ID.

This last scenario can be difficult, such as *base
of the tub in example (4), where the object detector failed to recognise the target object. We observed, however, that this happens when the speakers refer to parts of the objects, and then the bridging annotation scheme can be smoothly applied.

(4) 1. This is a picture of a bathtub. 2. The tub is white. 3. The wall and base of the tub are brown. 4. The door appears to be glass. 5. There is a handrail on the side wall.

For bridging references, if a mention which is visible is in a part-of relation with another object which does have a bounding box, then we ground it to that object as well.

This process of referring to sub-objects is also fairly common in Cups. For example, participants refer to the cups handles and tops that we did not identify earlier.

Last, the image also allows for typically semantic properties to be used to refer back to the objects: colour, shapes, sizes. These can be genuinely referential (a form of ellipsis) or used in attributive manner. Compare for example white in the second sentence of (4), with (5) below.

(5) P1: closest to me, from left to right red, blue, white, red
P2: ok, on your side I only see red, blue, white

Note that in the case of mentions annotated as groups, we ground all the elements belonging to the group. However, deciding which elements exactly the speaker had in mind can be ambiguous. In (6) from Cups the speakers refer to rows of objects even though these are not arranged in strict geometric lines. Hence, what objects are included in a row is contextually defined and not always clear.

(6) P2: ok, so your next row
P2: you said there 's a takeaway cup somewhere marooned all alone
P1: Okay. So we have that row I described with the now found red cup. Then a takeaway cup that is between that row and the next. It’s very much in the middle of the two rows.

Moreover, we observe references to different regions of the image, and these references change dynamically throughout the conversation, e.g. my left, your right, the first row. In the Cups corpus, we have split the scene into equal rectangular regions that are splitting the table into a grid as shown in Figure 1c. However, the grid nature of the sub-regions and their granularity are frequently insufficient as participants do not split the table to sub-regions in a grid-like manner but relative to the current focus on the scene and the topological arrangements of objects. In the example, “the empty space in the second row of objects close to you” an empty space has been designed as a new region which does not correspond to our projected grid-like regions. The references such dynamic objects must be resolved by the hearer and misunderstanding may occur, depending on the complexity and ambiguity of the scene.

Last, in the Cups corpus objects may be re-referred to again in different parts of a dialogue, potentially creating very long distance relationships between mentions. However, we generally restrict these to the scope of the dialogue games for which some parts of the corpus are also annotated.

4.5 The annotation process

Our annotation is implemented using the MMAX tool (Müller and Strube, 2006) for compatibility with the ARRMA MMAX schemes. An example of the annotator interface is presented in Figure 3. Besides the authors, three student assistants have been involved as annotators until now. We expect to release a first version of the annotation later during the year. This will include proper inter-annotator agreement metrics in order to evaluate the adequacy of the proposed schema.

4.6 Unaddressed challenge: speakers’ cognitive state

Contrary to a Gricean-based analysis of spoken discourse, coherence-based theories of discourse do not traditionally take the cognitive state of the speaker as a necessary element to text interpretation (Bender and Lascarides, 2019). In situated dialogue, however, although the image can be treated as the ground truth of the situation, the speaker’s cognitive state has to be considered by the hearer, in order to disambiguate the utterances. In other words, the hearer makes a model of the beliefs, desires and intentions associated with the utterance. This is exemplified in the following excerpt from
Figure 3: Example of annotation in the MMAX tool. Coreferential links are shown with the green lines in the bottom right. The annotator has simultaneous access to the image and the text while annotating all specified attributes in the annotation scheme.

**Cups** where both participants do not see one of the two red cups close by, but each a different one. They mistakenly believe that there is only one missing red cup and this dis-alignment of their beliefs gradually leads to increasingly diverging cognitive states.

(7) P2: there is an empty space on the table on the second row away from you
P2: between the red and white mug (from left to right)
P1: I have one thing there, a white funny top
P2: ok, i’ll mark it.
DIALOGUE_STATE: B found O-25.
P1: and the red one is slightly close to you
P1: is that right?
P1: to my left from that red mug there is a yellow mug
P2: hm...
P2: can’t see that and now i’m confused
DIALOGUE_STATE: B cannot see O-29.
P2: describe the second row away from you like you see it
P1: only one thing there, a white funny top
P2: aha, so it’s closer to you than those i call “the second row”
P1: behind that, there is a yellow, red, white and blue
P1: from my left to right
P1: yes, that must be it!
P1: so what do you see in the “second row” from my perspective?
P2: i see a red, then space, then white and blue (same as katie’s”)
P2: no yellow

P2: is it on the edge of the table?
P2: on your left
P1: ok, yes!
DIALOGUE_STATE: inconsistent

5 Conclusions

Different V&L resources provide with an opportunity to explore the notion of discourse entity and (co)reference in grounded context. Since the nature of contexts defined by the tasks in which the corpora were collected varies considerably we get an opportunity to study the phenomena over these contexts and get a more complete picture of reference. Extending the coreference annotation to the V&L domain is essential to understand the relationship between reference and coreference. Work around textual coreference has defined the task with insufficient consideration of the semantic aspects involved in the interpretation of anaphoric phenomena; whereas work from the V&L community assumes that coreferential information can be inferred latently. By extending the coreference annotation scheme to rich situated dialogue corpora, we make explicit the relations at play between the text and the image. The same mechanisms that humans adopt to solve coreference in the textual domain should underlay results in the V&L domain.
Indeed, reference is underspecified in both modalities; any kind of information extraction from these domains will benefit from mechanisms that resolve this underspecification: capturing coreference is a door to capturing coherence. Furthermore, a rich annotation scheme that is portable between tasks and contexts, leads to the development of corpora allowing the training of data driven systems for the V&L domain and social robotics.

References


Incremental Unit Networks for Multimodal, Fine-grained Information State Representation

Casey Kennington
Department of Computer Science
Boise State University
caseykennington
@boisestate.edu

David Schlangen
Department of Linguistics
University of Potsdam
david.schlangen
@uni-potsdam.de

Abstract

We offer a sketch of a fine-grained information state annotation scheme that follows directly from the Incremental Unit abstract model of dialogue processing when used within a multimodal, co-located, interactive setting. We explain the Incremental Unit model and give an example application using the Localized Narratives dataset, then offer avenues for future research.

1 Introduction

Human experience is profoundly multimodal. As people explore the world they are organizing perception, action, and thought in a complex social environment (Smith and Gasser, 2005). Tied directly to this multimodal experience is human language, primarily spoken language (Fillmore, 1981), and a growing body of literature across several disciplines make a strong case that language learning and language meaning is grounded in rich multimodal (even embodied), interactive, and enactive experience (Pulvermüller, 1999; Barsalou, 2008; Smith and Samuelson, 2009; Di Paolo et al., 2018; Bisk et al., 2020). Despite this, current state-of-the-art language models such as BERT (Devlin et al., 2018) are trained only using static text, and while it is clear that such models are powerful and useful for many tasks, they are clearly missing important multimodal semantic knowledge (Rogers et al., 2020; Bender and Koller, 2020). We argue that what is needed is a semantic model that is learned not only from text, but has knowledge of multiple modalities and that the model operates in a setting similar to how language is acquired for humans: multimodal, co-located, interactive spoken dialogue:

multimodality: A model of semantic meaning of language must ground into not just vision, but other modalities such as taste, touch, proprioception, and even affect. This is as much a modeling challenge as an engineering challenge, because each modality requires sensor hardware (e.g., cameras for vision) and methods for fusing the sensor information from different modalities.

co-location: Multimodal systems have multiple sensors that sense things like objects, events, and the interlocutor who has knowledge about the environment, language used to denote objects, and uses cues such as gaze and gestures in communication.

spoken interaction: Semantic meaning is learned and used in coordination with members of a particular language community (Clark, 1996) and spoken interaction is the setting where children learn language. Moreover, spoken language differs dramatically from written text in that spoken language contains communicative artifacts such as hesitations, false starts, repetitions, repairs, and coordination of turn-taking. Furthermore, people produce and understand language sequentially, not as complete and fully grammatical units (Tanenhaus and Spivey-Knowlton, 1995).

Taken together, these requirements imply technical and modeling challenges. Technical challenges include using multiple sensors and articulators, fusing their information streams, temporally aligning input and output. Modeling challenges include binding information from the sensors, learning meaningful patterns in a noisy setting, and representing the states of the sensors and unfolding interaction.

In this paper, we don’t formulate a semantic model, but focus rather on a representation with a fine-grained information state update approach using the Incremental Unit abstract model of spoken dialogue. We explain the Incremental Unit model in the next section, including how multi-
modal how information is represented, then offer a simple scheme for using Incremental Units as a basis for developing multimodal semantic models.

2 The Incremental Unit Framework

The Incremental Unit (IU) framework (Schlangen and Skantze, 2011) is an abstract, conceptual approach for incremental processing for spoken dialogue. The IU framework consists of a network of processing modules, each of which play a different role in an unfolding dialogue, all of which work together to create the fine-grained information state. Modules take input data on their left buffers, process the input, then produce output on their right buffers. A critical part of the IU framework is how the data are packaged and processed. The data are packaged as the payload of incremental units (IUs) which are passed between modules—each IU holds a discrete amount of information.

Another critical part of the framework is that the IUs themselves are interconnected via same level links (SLL)—allowing the linking of IUs as a growing sequence—and grounded-in links (GRIN) which allow that sequence to convey what IUs directly affect another IU. Ideally, IUs (e.g., produced from a sensor or processing module) can be guaranteed to be correct, but often an IU that has been outputted to the next module needs to be updated in light of new information. To make this possible, the framework makes use of three operations: IUs can be added to the IU network, but can be later revoked, and also committed when a module can guarantee that an added IU will not be revoked.

![Figure 1: Example of SLL, and Add, Revoke and Commit operation for an incremental speech recognizer.](image)

Figure 1 shows an example of how a speech recognition module would process incrementally, typically word-by-word. It takes a continuous audio signal as input from a microphone and produces discrete word IUs as output. As the utterance *I will leave now* is uttered, the speech recognizer outputs words as they are recognized at the word level and adds them to the IU network. The recognizer mis-recognized the word *live*, but in light of new information from the unfolding utterance, revoked *live* and replaced it with *leave*. Horizontal arrows show SLLs; i.e., how the IUs are related to each other temporally, and at the end of the utterance when the recognizer knows it will no longer revoke, it marks all of the IUs as committed. IUs also contain information about their creation time.

It’s important to distinguish at this point the networked IU modules or processors that pass IUs to each other and the network of IUs themselves. For example, a speech recognizer might pass its transcribed speech as IUs with payloads of word strings to a part-of-speech module that produces a part-of-speech for each word as payloads of part-of-speech strings, which are then the input of a language understanding component that operates on both the words and parts-of-speech to produce some kind of semantic abstraction of the unfolding utterance. Thus the three processors—speech recognizer, part-of-speech tagger, and language understanding—are separate modules, but each use the add, revoke, commit operations to alter the shared network of IUs. The IU framework, including the operations, can be used as a fine-grained model of the dynamics of the creation of the information state of an agent in a situated interaction, comprising both its world model and its discourse model, and the interaction between them.

Multimodal Example Following Kennington et al. (2014), Figure 2 shows an example of modules and IUs created by a multimodal system co-located with a human interlocutor. For this example, the system is tasked with learning about objects. In this specific turn of the interaction, the interlocutor utters *this is my phone* accompanied by a display of the phone and a deictic pointing gesture. The system has two sensors, a camera and a microphone. The microphone feeds continuous audio to the automatic speech recognizer (ASR), which transcribes the utterance into word IUs. Those are outputted to a part-of-speech (POS) tagger that produces part-of-speech IUs. Those in turn are outputted to the semantic parser (SEM) which produces a semantic abstraction over the utterance; the semantic parser uses both words and parts-of-speech to produce the under-specified semantic parse IUs. Those are given to a natural
Figure 2: Example of a system made up of two modalities (i.e., audio and vision), camera and microphone sensors, and processing modules. An interlocutor says *this is my phone* accompanied by a deictic gesture to the phone; the modules process the scene and audio; the DM (dialogue manager) makes a decision to ask a clarification question which is rendered by the NLG as *is that a phone?*. The modules create the IUs, which are connected to each other via same-level links (solid lines) and grounded in links (dashed lines), the latter denote the IUs that played a role in that IU’s creation. For example, the bottom IU for NLU needed information from IUs created by the ObjRec, GestRec, and SEM modules. The full network constitutes a multimodal meaning representation.

Prior Work  
As a theoretical model, the IU framework formed the basis for a model of temporally aligning different sensor modalities; Kennington et al. (2017a) showed that timestamp information in the IUs can be used to inform modules to add IUs to the IU network at the same time, thereby giving downstream modules information about an event that may have happened, even if the sensors produced processing delays. Buß and Schlangen (2011) leveraged the IU operations for an incremental dialogue manager that could make self-corrections (e.g., if the system began an utterance, but a revoke meant that the utterance should change, the system would self-correct), and Lison and Kennington (2017) used the IU operations to inform a neural conversation model. The IU framework has also been the inspiration for several spoken dialogue system architectures, and several implementations based on the IU framework have been developed. InproTK (Baumann and Schlangen, 2012) is the most commonly used (written in Java), and was extended to incorporate modalities beyond just speech (Kennington et al., 2017b). More recently, ReTiCo (Michael and Möller, 2019) was developed (written in Python) and extended to incorporate multiple modalities, evaluated in a multimodal robotic system (Kennington et al., 2020).

Using a network (or a graph) to represent meaning has received recent attention, yet has a long history. Koller et al. (2019) provides an overview of several formalisms, including Abstract Meaning Representation (Banarescu et al., 2013), a particular representation that has seen adoption in the community. However, these graph-based semantic representations are focused only on representing sentences, not multimodal information, and does abstract away from the dynamics of creating the network.

3 The IU Framework for Fine-grained Information State Representation

In this section, we sketch a scheme for the IU network as a representation of a fine-grained information state. The scheme follows the IU approach to processing live speech; all annotations are packaged as IUs with links between them, all *add op-*
operations are accounted for (and revoke operations under live annotation conditions), each operation is timestamped, and the creation time of each IU is timestamped. We don’t specify how the modalities or modules interact with each other, the goal here is to focus on the information state.

We give an example in Figure 3 using a sample from the Localized Narratives dataset (Pont-Tuset et al., 2020). The dataset consists of images described by annotators. Descriptions have speech and mouse pointer modalities that are later temporally aligned. Speech is automatically transcribed as the annotators speak, but annotators are tasked with hand-transcribing their descriptions after they are complete. The dataset on its own has multimodal annotations, though it’s unclear how they would work in a live interaction with a system.

The IU network annotation in Figure 3 shows locations of mouse pointer (x,y coordinates), words, and added part-of-speech tags and semantic abstraction similar to that in Figure 2. The SLL and GRIN links are also present, and additional links between the speech and pointer modalities are depicted. What is not depicted in the figure are the add and revoke operations that enable the network to grow as an interaction unfolds in real time, though it is obvious that all IUs in the figure were created through an add operation. In the case where a perfect transcription exists, only add operations are necessary, but a live interaction would require the ability to revoke erroneous words then add correct ones in real time, in alignment with the movements of the mouse pointer. Timestamp information is not present in the figure; time generally flows downward as IUs are added to the network.

The scheme can be applied during the data collection process. This requires some up-front effort to setup each individual module to operate incrementally. For the Localized Narratives dataset, incremental text can come from ASR or typed text, and the other annotations from respective modules. Annotated data can be represented in any format, e.g., JSON. This scheme highlights the importance of annotating data that is representing a fine-grained information state collected in a multimodal, co-located, and spoken interactive task. Such a representation is potentially useful for a formal representation of situated conversation and embodiment.

4 Conclusion and Future Work

In this paper, we outlined an IU network-based approach to representing multimodal states within the requirements of multimodality, co-location, and interactive speech. Implicit in this representation is the requirement that the system is modular, though it is potentially possible to represent the IU network in an end-to-end neural architecture. The modalities explored here were only a minimal example of what the network could potentially handle—added modalities enrich the semantic representation. For example, we have used the IU framework to represent audio, visual, and internal robot state modalities in prior work (Kennington et al., 2020). We leave formalizing semantic operations, such as compositionality, meaning derived from handling uncertainty or requests for clarification, and global decoding strategies in the IU network semantic representation for future work.
Acknowledgements We appreciate the feedback from the anonymous reviewers.

References


Teaching Arm and Head Gestures to a Humanoid Robot through Interactive Demonstration and Spoken Instructions

Michael Connolly Brady  
DFKI, Saarland Informatics Campus  
66123, Saarbrücken, Germany  
michael.brady@dfki.de

Han Du  
DFKI, Saarland Informatics Campus  
66123, Saarbrücken, Germany  
han_h.du@dfki.de

Abstract
We describe work in progress for training a humanoid robot to produce iconic arm and head gestures as part of task-oriented dialogic interaction. This involves the development of a multimodal dialogue manager and corresponding system architecture for non-experts to ‘program’ the robot through speech and vision. Using this system, videos of gesture demonstrations are collected. Motor positions are extracted from the videos to specify motor trajectories, where collections of motor trajectories are used to produce robot gestures following a Gaussian mixtures approach. Concluding discussion considers how learned representations may be used for gesture recognition by the robot, and how the core system may mature into a robust system to address language grounding and semantic representation.

1 Introduction
A conventional way of programming robots to make iconic gestural movements is to animate movements as sequences of static motor positions. This method is slow and tedious and an easier method is sought. Ideally, people should be able to teach a robot how to make new gestures through visual demonstration and verbal instruction, as they might teach another person how to make a new arm and head gesture. Such a multimodal interactive approach is one of today’s current challenges in robotics. Perhaps one reason that multimodal interaction with robots is problematic relates to the compartmentalization of research specialties. Speech engineers are generally not experts at computer vision and motor control. Likewise, robotics engineers and computer vision engineers tend to treat speech and language as a ‘black box’ problem best left to speech and language technologists. The result is that language, vision, and motor control tend to be segregated during software planning and implementation. It is left to the robot interaction engineer to cobble these segregated modalities together into a cohesive software framework. The broad aim of our project is to pragmatically address this challenge by developing a processing architecture where communicative information across modalities can be more integrated. Teaching a robot how to produce gestures through visual demonstration and spoken dialogue is a task that is well suited for addressing the challenge.

Robot Learning from Demonstration (LfD), sometimes also referred to as “robot programming by demonstration,” “teaching by example,” or “imitation learning” is an established approach for training robots through vision. As alluded to above however, one issue with LfD is that LfD practitioners generally fail to incorporate the power of verbal instruction, see (Ravichandar et al., 2020). We posit that with the relatively recent advent of Deep learning and related breakthroughs in computer vision, artificial speech recognition, and related technologies, the time is ripe to integrate natural verbal instruction with LfD.

LfD and training by example has a rich history and is a popular research area in modern robotics, for example see: (Calinon and Billard, 2007; Argall et al., 2009; Koenig et al., 2010; Calinon et al., 2010; Lee, 2017; Zhu and Hu, 2018; Ravichandar et al., 2020). LfD sidesteps more traditional and tedious methods of manually specifying motor control or where math and computer programming expertise is required. The essence of LfD is that robot movements may be acquired by having a person act out the movements to be learned (either through telepresence, kinetically, or visually), and transposing those movements into representations that a robot may use in combination with the robot’s knowledge and internal processing to then produce the movement. It is important to note that LfD is not merely a ‘record and replay’
technique. Generalization is required so that, for example, starting and ending positions of the movements are not pre-determined. Exact trajectories as well as amplitudes of movements may vary insofar as the task demands, and resulting movements should be robust in the face of changing environmental conditions and actuator imprecisions. For our present purpose, the idea is also to avoid exact monotonous repetitions, and to develop robust representations that may also be used for perceiving learned gestures.

Interacting with robots through natural language is another popular area of research. E.g. see: (Cantrell et al., 2010; She et al., 2014; Gemignani et al., 2015; Misra et al., 2018; Liu and Zhang, 2019; Kruijff-Korbayova et al., 2020). Perhaps the most popular domain for linguistic information transfer between people and robots is in giving travel or route instructions, such as in the spoken guidance of robotic wheelchairs, for a review see: (Williams and Scheutz, 2017).

It is important to note that speech communication also contains non-linguistic cues, both vocal (e.g. laughter, affect, tone) and non-vocal (e.g. gestures, eye gaze, face expressions, environmental context). For related review, see: (Mavridis, 2015; Devillers et al., 2020). In addition to the linguistic signal, these and related cues should be readily available for incorporation into interaction designs.

2 Method

The robot this work uses is “VoxHead,” a 3D printed humanoid robot (Brady, 2016; Devillers et al., 2020). Figure 1 displays the robot. The robot serves as a life-sized and relatively low cost platform for interactive social robotics research. The robot has motors for mouth, eye cameras, and facial expressions. For the present work we do not concern ourselves with facial motors. Instead, focus is on general head, neck, and arm movements. In total there are sixteen degrees of freedom in the head, neck, and arms that we work with. Specifically we use: head tilt, head turn, neck tilt, neck turn, and for each arm: arm raise-lower, arm left-right, arm rotate, elbow bend, wrist rotate, and wrist bend. Hands with individual fingers or grippers are also not used here.

2.1 Control Architecture

Figure 2 depicts the general software plan. Sensory input to the robot is handled by a series of perception modules. A perception module may run on it’s own mini-computer as e.g. an end-to-end DNN, or may run on a remote server, such as with an ASR engine. A countless number of perceptual processing modules may in theory be included, a few of which are portrayed here. For the present purpose of simplicity, only a speech-to-text ASR percep-
tion module (Amazon Transcribe), and a skeletal tracking perceptual module (to be described in Section 2.2) are used. Input from these two sources is received by an Interaction Manager (IM). The IM collects sensory input based on a control signal from the Core Cognitive Architecture (CCA). Sensory input that is requested by the CCA feeds to an Information Store (IS), for cognitive processing. The IM also relays commands from the CCA to be executed by various production modules. Like with the perception modules, a countless number of production modules may be included, a few are portrayed, and for the present purpose only the two highlighted modules (speech synthesizer, and head and arms motors controller) are considered here.

The CCA is very much a work in progress. Skeletal tracking information is read by a Task Manager (TM), within the CCA for data processing (see Section 2.2), while linguistic representations and semantic gestures are read in by the Multimodal Dialogue Manager (MDM). Some multimodal dialogue managers have been proposed over the years, e.g.: (Wahlster, 2006; Sanders and Holzapfel, 2008; Peternel et al., 2014; Ondáš and Juhár, 2015). In developing the MDM, there are a variety of topics in human-robot communication to address. For a review, see: (Breazeal et al., 2004; Tellex et al., 2011; Ajoudani et al., 2018; Gluck and Laird, 2019).

We take inspiration from the above cited multimodal dialogue managers in combination with a more recently implemented open-source dialogue manager called VOnDa, (Kiefer et al., 2019). Dialogue management using VOnDa is founded on the information state based approach (Traum and Larsson, 2003). The information state contains the robot’s state, including dialogue as well as domain specific information. Here, the information state may be extended by additional [multimodal] contextual knowledge. VOnDa’s information state is represented as extended OWL ontologies and managed using a semantic repository and reasoner called HFC (Krieger and Willms, 2015). With VOnDa, changes in the robot’s information state trigger a declarative rule system with statistical selection to generate a dialog act in response to the situation. A dialogue act generally results in
the output of text (to be converted to speech), but may also be realized as motor control directives, and other modalities, such as affective cues for a text-to-speech synthesizer. For the MDM we are also pursuing how to incorporate a construction grammar approach with ontologies for language learning. See: (Steels, 2004; Oliva et al., 2012; Linderes and Laird, 2017). We are also considering how our MDM may integrate with a V oX M L approach (Pustejovsky and Krishnaswamy, 2016).

Output from the MDM is combined with output from the TM to assemble a control signal by the Interaction Planner (IP), to be interpreted and executed by the IM. This signal is implemented using an extensible markup protocol. The IM runs locally on the robot and is designed to be very fast, mainly handling interrupts and conflict resolution. Meanwhile, the CCA may be hosted on a super machine or distributed across machines with unlimited processing power. Though the control signal from the CCA via the IP is dynamically generated, stand-alone or static control scripts may be used in place of the CCA. This allows the IM and its processing modules to be tested in the absence of the CCA. This also allows the IM to be developed as a stand-alone Robot Operating System (ROS) package, to be used with other cognitive architectures. The use of static control scripts in place of the CCA converts our system architecture into a menu-driven dialogue system. That is, with static control scripts the IM may be regarded as something of a multimodal V oX M L interpreter.

Consider the following scenario. A human trainer named John begins a learning session by saying something along the lines of “okay robot, let’s learn a new gesture.” With this, the robot is triggered to enter ‘gesture learning mode’ and when the robot is ready with its front camera recording, the robot responds with some variation of “okay, John, I’m ready.” John then performs the body gesture that he wants the robot to learn. For example, let us consider a gesture to indicate ‘stop’ - the gesture a police officer might use when directing traffic and signaling a car to stop (as in Figure 1, bottom left). While performing the gesture, John may give a verbal description, such as “lift your hand like this, palm up and fingers stretched, and extend the arm forward.” Once John has finished producing the gesture, he then says: “that’s it,” and the robot acknowledges this by saying “okay,” or something analogous. The video recording of the gesture is then saved and processed into a labeled representation as described in Section 2.2.

After processing and maybe after multiple examples of the desired gesture have been recorded, the robot should be ready to produce the gesture. In this case, the robot says something amounting to: “shall I perform the gesture now?” and John may respond with feedback indicating “yes” or “no,” prompting the robot to then execute the gesture or not. If there was a problem during processing, the robot may ask John to repeat the gesture. Once the robot has performed the gesture, the robot then asks: “was that okay?” and John may verbally respond “yes, good” while nodding his head ‘yes’ and-or giving a ‘thumbs up’ gesture. Or John may indicate ‘no, let’s try again’ while shaking his head ‘no’ and giving a ‘thumbs down’ hand gesture (assuming yes/no head and hand gestures have been acquired by the robot). Either a verbal command or a visual command should be enough for the interaction to proceed. The robot might then say ‘what does this gesture mean?’ John would then explain the meaning of the gesture and the robot would store the gesture with a semantic label (e.g. ‘stop’).

2.2 Gesture Acquisition

When in ‘gesture recording mode,’ the robot records a video of the person’s complete motion. Each motion or gesture is stored in a buffer as a video example. The trainer (or multiple different trainers) can record the same motion multiple times, and the repetitions are stored as new examples under the same class. We use OpenPose (Cao et al., 2019) for its current superior performance in extracting 2D skeletal information from the recorded video examples. For representing and reconstructing 3D motions from the 2D poses, we deploy a dilated fully convolutional model (Pavillo et al., 2019) to estimate a 3D skeletal pose at each sampled frame. Each pose is represented as a set of Cartesian joint positions. Sequences of the extracted 3D positions are transformed into estimated motor positions for a single video example, and are saved as a motor trajectory. A motor trajectory takes the form of a matrix. The columns of the matrix correspond to motor channels of the robot, and rows of the matrix correspond to the passage of time. If a user is satisfied with a gesture reproduced by the robot, the video sample of the gesture may be discarded, and only the motor trajectory needs to be saved.
Though the robot can produce a gesture based on a single example, it is better to generalize the motion under the same gesture label, assuming there are multiple examples for the same class of gesture. This is done to reflect naturalness that real people perform the same motion with a rich repertoire of variations. In order to capture these variations, we apply a mixture of Gaussians (Min and Chai, 2012) to generalize the distribution of the motion examples $P(x)$ for each gesture. This is done following Equation 1.

$$P(x) = \sum_{k=1}^{K} \phi_k N(\mu_k(x), \sigma_k(x))$$

(1)

One issue in combining multiple motor trajectories is that each motion example may have a different length, meaning the number of frames could vary. To address this, we define a canonical timeline and time normalize all motion examples in the same class to this canonical timeline. The resulting statistical motion model provides a compact way to represent each gesture as a set of discrete examples. With statistical motion models, gestures can be represented in a continuous manifold space. In the gesture production phase, if the robot is asked to perform a gesture (e.g. ‘stop’) without any additional constraints, our model can sample a random motion to be close to the examples with high likelihood. For the gestures with additional constraints, for instance, if the direction of the robot arm is specified, or the robot starts from an unusual initial pose, our model can formulize it as an optimization problem to find the best match in a continuous motion space. Following Equation 2.

$$\arg \max_{x} P(x|c)$$

(2)

where $c$ is a set of constraints, which can be target positions or orientations, and even some high level constraints. Furthermore, if an end effector position is specified, the statistical motion model can be coupled with inverse kinematics and-or a visual guidance system. Our system does not simply produce deterministic motions from examples, but is enabled to produce similar motions with new variations. In addition, our motion model can be continuously tuned by adding new examples.

It should be noted that in estimating motor positions from Cartesian 3D joint data using inverse kinematics, there is ‘motor bleed over.’ This relates to how people’s skeletons differ in size and proportion to each other and to the robot’s skeleton. It is thus difficult to isolate desired robot motor movements for system calibration. An improved method for motor position estimation from skeletal data is desired and is a focus of current efforts.

3 Discussion

We have introduced the infrastructure of an interactive speech-vision-motor system for training a life-sized humanoid robot to produce desired arm and head gestures. The system interfaces a rudimentary cognitive architecture with an interaction manager for robot control. We use an LfD technique combined with spoken instructions and dialogue for training a robot to produce gestures. We lastly turn to consider the relationship between perception and action, the language grounding problem, and semantic representation.

There is an intimate relationship between perception and action. The research industry surrounding the mirror neuron hypothesis reifies this (Hickok, 2014) In light of this, our current work also includes the development of a gesture recognition algorithm that depends on production learning. The time-normalized motor trajectories of a class from Section 2.2 define a centroid motor trajectory for the class. We call this centroid a gesture prototype. In short, a motor trajectory to be categorized is template-matched against the stored inventory of gesture prototypes using a multidimensional dynamic time warping algorithm (Müller, 2007). The best match is taken as the gesture’s category.

Plans are to develop our system to addresses the symbol grounding problem (Harnad, 1990; Steels, 2003; Cangelosi, 2010; Misra et al., 2016). Establishing a socially situated and embodied system for interactive gesture learning was but a first step. Semantic meaning must be grounded in experience, where different modalities (speech, vision, motor feedback) are integrated. Interactive audio-visual-motor recordings from our system may be used for machine learning approaches, e.g. (Santín et al., 2020) to train multi-modal speech recognizers. In order for meaning to emerge, the robot must ‘understand’ its own output. By pursuing a paradigm where gesture recognition is based on the robot’s representations for gesture production, our hope is to depict representations to be one and the same for perception and production. In viewing speech as a problem of motor control, speech cognition becomes grounded in the robot’s experience.
Acknowledgments

We would like to thank our colleagues for discussions and three anonymous reviewers for their feedback. This work is part of the research project XAINES, funded by grant No. 01W20005 of the German Ministry for Education and Research (BMBF). Please find more information on XAINES here: https://www.dfki.de/en/web/research/projects-and-publications/projects-overview/projekt/xaines

References


Building a Video-and-Language Dataset with Human Actions for Multimodal Logical Inference

Riko Suzuki
suzuki.riko@is.ocha.ac.jp
Koji Mineshima
minesima@abelard.flet.keio.ac.jp
Daisuke Bekki
bekki@is.ocha.ac.jp

1 Ochanomizu University, Tokyo, Japan
2 The University of Tokyo, Tokyo, Japan
3 Keio University, Tokyo, Japan

Abstract

This paper introduces a new video-and-language dataset with human actions for multimodal logical inference, which focuses on intentional and aspectual expressions that describe dynamic human actions. The dataset consists of 200 videos, 5,554 action labels, and 1,942 action triplets of the form (subject, predicate, object) that can be translated into logical semantic representations. The dataset is expected to be useful for evaluating multimodal inference systems between videos and semantically complicated sentences including negation and quantification.

1 Introduction

Multimodal understanding tasks (Johnson et al., 2017; Suhr et al., 2017, 2019) have attracted rapidly growing attention from both computer vision and natural language processing communities, and various multimodal tasks combining visual and linguistic reasoning, such as visual question answering (Antol et al., 2015; Acharya et al., 2019) and image caption generation (Vinyals et al., 2015), have been introduced. With the development of the multimodal structured datasets such as Visual Genome (Krishna et al., 2017), recent studies have been tackling a complex multimodal inference task such as Visual Reasoning (Suhr et al., 2019) and Visual-Textual Entailment (VTE) (Suzuki et al., 2019; Do et al., 2020), a task to judge if a sentence is true or false under the situation described in an image.

The recently proposed multimodal logical inference system (Suzuki et al., 2019) uses first-order logic (FOL) formulas as unified semantic representations for text and image information. The FOL formulas are structured representations that capture not only objects and their semantic relationships in images but also those complex expressions including negation, quantification, and numerals. When we consider extending the logical inference system between texts and images to that between texts and videos, it is necessary to handle the property of video information: there are dynamic expressions to capture human actions and movements of things in videos more than in images.

As an example, consider a video-and-language inference example in Figure 1. This video consists of SCENE 1, where the sentence The woman puts on her outerwear is true, and SCENE 2, where the sentence The woman takes off her outerwear is true. Note that the entire video represents richer information as expressed by the sentence the woman tries to put on her outerwear. To judge whether this sentence is true, it is not enough to simply combine two actions, putting on outerwear and taking off outerwear. To capture this dynamic aspect of human action, it is necessary to take into account the information expressed by intentional phrases such as trying to put on outerwear.

Towards such a complex multimodal inference between video and text, we build a new Japanese video-and-language dataset with human actions. We annotate videos with action labels written in triplets of the form (subject, predicate, object), where object can be empty (indicated by ϕ). Action labels contain not only basic expressions such as ⟨person, run, ϕ⟩ and ⟨person, hold, cup⟩,
but also expressions including intentional phrases such as \( \text{⟨person, try to eat, food} \rangle \). An advantage of using triplets \( \text{⟨subject, predicate, object} \rangle \) is that a triplet itself can serve as the semantic representation of a video and can be translated into logical formulas (see Section 3). This paper introduces a method to create a video-and-language dataset involving aspectual and intentional phrases. We collect a preliminary dataset labeled in Japanese for human actions. We also analyze to what extent our dataset contains various aspectual and intentional phrases. Our dataset will be publicly available at https://github.com/rikos3/HumanActions.

2 Related Work

There have been several efforts to create human action video datasets in the field of computer vision. Charades (Sigurdsson et al., 2016) contains 9,848 videos of daily activities annotated with free-text descriptions and action labels in English. Charades STA (Gao et al., 2017) is a dataset built by adding sentence descriptions with start and end times to the Charades dataset. For Japanese video datasets, STAIR Actions (Yoshikawa et al., 2018) is a dataset that consists of 63,000 videos with action labels. Each video is about 5 seconds and has a single action label from 100 action categories. Action Genome (Ji et al., 2020) is a large-scale video dataset built upon the Charades dataset, which provides action labels and spatiotemporal scene graphs.

VIOLIN (Liu et al., 2020) introduces a multimodal inference task between text and videos: given a video with aligned subtitles as a premise, paired with a natural language hypothesis based on the video content, a model needs to judge whether or not the hypothesis is entailed by the given video. The VIOLIN dataset mainly focuses on conversation reasoning and commonsense reasoning, and the dataset contains videos collected from movies or TV shows.

Compared to the existing datasets, our dataset is distinctive in that action labels are written in structured representations \( \text{⟨subject, predicate, object} \rangle \) and contain various expressions such as continue to eat and try to close that support complex inference between videos and texts.

3 Semantic Representations of Videos

Suzuki et al. (2019) proposed FOL formulas as semantic representations of text and images. They use the formulas translated from FOL structures for images to solve a complex VTE task. We extend this idea to semantic representations of videos.

FOL structures (also called first-order models) are used to represent semantic information in images (Hürlimann and Bos, 2016). An FOL structure for an image is a pair \( (D,I) \) where \( D \) is a domain consisting of all the entities occurring in the image, and \( I \) is an interpretation function that describes the attributes and relations holding of the entities in the image (Suzuki et al., 2019).

To extend FOL structures for images to those for videos, we add to FOL structures a set of scenes \( S = \{s_1,s_2,\ldots,s_n\} \) that makes up a video, ordered by the temporal precedence relation. This structure may be considered as a possible world model for standard temporal logic (Venema, 2017; Blackburn et al., 2002). Thus, a video is represented by \( (S,D,I) \), where \( S \) is a set of scenes linearly ordered by the temporal precedence relation, \( D \) is a domain of the entities, which is constant in all scenes, and \( I \) is an interpretation function that assigns attributes and relations to the entities in each scene. We assign personal IDs \( (d_1,d_2,\ldots,d_n) \) to people appearing in each scene. Since the purpose of our dataset is to label human actions, we assign IDs to people, but not to non-human objects.

To facilitate the annotation of the attributes and relations holding of the entities in each scene, we use triplets of the form \( \text{⟨subject, predicate, object} \rangle \) given to each scene \( s_i \) as action labels, where object may be empty. This form itself can be seen as a semantic representation of videos. Furthermore, it can also be translated into an FOL formula, in a similar way to the standard translation of modal logic to FOL (Blackburn et al., 2002). The following examples show a translation from triplets in scenes into FOL formulas.

(1) \( s_1 : \langle d_1, \text{run, } \phi \rangle \)
\[ \Rightarrow \text{run}(s_1, d_1) \]

(2) \( s_2 : \langle d_1, \text{hold, pillow} \rangle \)
\[ \Rightarrow \exists x(\text{pillow}(s_2, x) \land \text{hold}(s_2, d_1, x)) \]

Here each predicate has an additional argument for a scene variable. (1) means that the entity \( d_1 \) runs in scene \( s_1 \); (2) means that the entity \( d_1 \) holds a pillow in scene \( s_2 \).

Each triplet can be translated into an FOL formula by using this method and thus serve as a se-
mantic representation of a video usable in the semantic parser and inference system for the VTE task presented in Suzuki et al. (2019). Though it is left for future work, the dataset in which each scene of a video is annotated with triplets will be useful to evaluate the VTE system for videos.

4 Data Collection

4.1 Video Selection

We selected videos from the test set of the Charades dataset (Sigurdsson et al., 2016). The Charades dataset contains videos drawing daily activities in a room such as drinking from a cup, putting on shoes, and watching a laptop or something on a laptop. Each video is collected via crowdsourcing: workers are asked to generate the script that describes daily activities and then to record a video of that script being acted out.

We select videos where multiple persons appear from the Charades test set to cover various actions within human interaction such as touching someone’s shoulder or handing something. These actions are expected to be described in expressions involving various linguistic phenomena. To collect videos where multiple persons appear, we selected 200 videos whose descriptions include phrases another person, another people, and they. Figure 2 shows a video example involving human interaction.

4.2 Annotation

We annotate each video with ⟨subject, predicate, object⟩ triplet format as action labels that represent human-object activities. We also annotate each action label with a start and end time to locate the activity accurately. We ask two workers to freely write predicates and object names that describe human activities to collect various expressions. Using this format the workers can freely decide the span of each scene and thus annotate a video with action labels more easily and flexibly. In Section 4.5 below, we will explain how to convert the triplet action format with start and end times to FOL structures extended with scenes as presented in Section 3.

Subject We assign personal IDs (d₁, d₂, d₃, . . .) to people in order of appearance in the video. If multiple persons appear for the first time in the same scene, we assign personal IDs to people appearing in order from left to right.

Predicate In a triplet, predicate contains various expressions such as aspectual and intentional phrases for describing dynamic human actions in videos, those phrases that do not usually appear in captions for static images. The following examples show characteristic predicates of videos.

- predicates for utterance and communication (e.g. speak, talk, tell, ask, listen)
- predicates for intention and attitude (e.g. try to eat, try to close).
- aspectual predicates (e.g. start talking, continue to eat)

We allow workers to use not only a transitive or intransitive verb but also verb phrases for predicates such as try to V and continue to V to collect diverse aspectual and intentional phrases.

Object The object in a triplet contains an object name or personal ID. If the item in predicate is an intransitive verb, object is empty. For instance, in Figure 3, the object for the predicate hold is pillow and the object is empty for the predicate run.

Figure 3: A man is running while holding a pillow. Action labels are ⟨d₁, hold, pillow⟩ and ⟨d₁, run, ϕ⟩.

4.3 Validation

In this work, we ask three workers to either annotate or merge action labels. All of the workers are native speakers of Japanese. We merge and confirm action labels in the following steps: (1) merge action labels made by two workers and arrange them in ascending order of start times, (2) watch videos by three workers to see if an action label is correct, and (3) if action labels duplicate, select one action label.

Regarding duplicated action labels, the labels and their start and end time are determined according to the agreement of three workers. Consider the following duplicate case.

Figure 2: Example video for the action of touching someone’s shoulder from the Charades dataset.
Table 1: A comparison of our dataset with existing datasets

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Videos</th>
<th>Average time (sec)</th>
<th>Average of action labels</th>
<th>Action categories</th>
<th>English</th>
<th>Japanese</th>
</tr>
</thead>
<tbody>
<tr>
<td>Charades (Sigurdsson et al., 2016)</td>
<td>1942</td>
<td>30</td>
<td>6.8</td>
<td>157</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>ActionGenome (Ji et al., 2020)</td>
<td>9848</td>
<td>30</td>
<td>170</td>
<td>157</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>STAIR Actions (Yoshikawa et al., 2018)</td>
<td>102462</td>
<td>5-6</td>
<td>1.0</td>
<td>100</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Ours</td>
<td>200</td>
<td>30</td>
<td>27.77</td>
<td>1942</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

Table 2: Predicates for utterance, intention and aspect

<table>
<thead>
<tr>
<th>Predicate</th>
<th>Freq.</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>Utterance</td>
<td>138</td>
<td>1.28%</td>
</tr>
<tr>
<td>(σ₁)</td>
<td>0:10-0:13</td>
<td>d₁, hold, clothes</td>
</tr>
<tr>
<td>(σ₂)</td>
<td>0:11-0:14</td>
<td>d₁, hold, clothes</td>
</tr>
<tr>
<td>(σ₃)</td>
<td>0:11-0:15</td>
<td>d₁, hold, outerwear</td>
</tr>
</tbody>
</table>

| Attitude      | 51    | 0.98%                           |
| (σ₁)          | 0:10-0:13 | d₁, stop walking               |
| (σ₂)          | 0:11-0:14 | d₁, stop walking               |
| (σ₃)          | 0:11-0:15 | d₁, stop to hang              |

In this case, (σ₁) and (σ₂) are duplicates in that subject, predicate, and object are the same while the start time and end time are different. If the third worker judges that (σ₂) is more adequate than (σ₁), we merge (σ₁) and (σ₂) and obtain the action labels below.

| Aspect        | 8     | 0.15%                           |
| (σ₁)          | 0:10-0:13 | d₁, continue to build(1)        |
| (σ₂)          | 0:11-0:15 | d₁, continue to build(1)        |

Figure 4: Annotation example of a video labeled with various types of predicates. Here s₁, . . . , s₁₁ are scenes linearly ordered by the temporal precedence relation.

Table 3: Top 10 frequent action labels. Action labels are written in form of predicate_object or predicate.

<table>
<thead>
<tr>
<th>Action label</th>
<th>Freq.</th>
<th>Rate(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>行くwalk</td>
<td>288</td>
<td>5.19</td>
</tr>
<tr>
<td>立つstand_floor</td>
<td>221</td>
<td>3.98</td>
</tr>
<tr>
<td>立ち止まるstop walking</td>
<td>102</td>
<td>1.84</td>
</tr>
<tr>
<td>立つstand</td>
<td>96</td>
<td>1.73</td>
</tr>
<tr>
<td>見るsee</td>
<td>81</td>
<td>1.46</td>
</tr>
<tr>
<td>話すtalk</td>
<td>81</td>
<td>1.46</td>
</tr>
<tr>
<td>笑うlaugh</td>
<td>71</td>
<td>1.28</td>
</tr>
<tr>
<td>食べる_eat_food</td>
<td>54</td>
<td>0.97</td>
</tr>
<tr>
<td>飲む_drink_beverage</td>
<td>48</td>
<td>0.86</td>
</tr>
<tr>
<td>保持_copthold_cup</td>
<td>47</td>
<td>0.85</td>
</tr>
</tbody>
</table>

4.4 Collection Statistics

Table 1 shows that despite its size, our dataset contains more action categories than other previous datasets. About 65% of total action labels are action labels that appear only once. This indicates that there are a wide variety of expressions.

The dataset contains characteristic expressions of videos such as walk, talk, and stop walking. Table 2 shows the frequency and examples of three types of predicates, i.e., utterance, intentional, and aspectual predicates. The distribution of characteristic predicates of videos in our dataset was: 2.49% predicates for utterance, 0.98% predicates for intention and attitude, and 0.15% aspectual predicates. One possible reason for the low frequency of aspectual predicates is that Charades contains 30-second videos, which might be too short to describe multiple actions involving aspectual phrases. It would be expected to increase the number of aspectual predicates if we annotate
longer videos such as the VIOLIN dataset (Liu et al., 2020), which is left for future work. The number of overlaps of action categories between ours and STAIR Actions (Yoshikawa et al., 2018) is 28. These results indicate that our dataset contains more diverse action categories compared to other datasets.

Table 3 shows frequent action labels in our dataset. Our dataset contains not only predicates for utterance, intention, and aspect, but also punctual verbs (e.g. stop walking and turn on) and durative verbs (e.g. sit and wait).

4.5 Conversion to FOL structures

The triplet action forms with start and end points used in the annotation can be converted to FOL structures extended with scenes presented in Section 3. In the extended FOL structures, each scene is linearly ordered by the temporal precedence relation and is uniquely characterized by the set of all the attributes and relations holding in it.

As an illustration, consider the example in Figure 4. In this case, we can separate the entire video into 11 scenes as shown in Figure 4. Accordingly, in the extended FOL structure, we have \( S = \{ s_1, \ldots, s_{11} \} \). Here the first scene, \( s_1 \), consists of the following: the predicate run holds of the entity \( d_1 \), the predicate sit holds of the pair \( (d_2, x_1) \) where \( x_1 \) is an entity which is a table. In terms of the interpretation function \( I \) relativized to a scene, we have \( I_{s_1}(\text{run}) = \{ d_1 \} \), \( I_{s_1}(\text{sit}) = \{ (d_2, x_1) \} \) and \( I_{s_1}(\text{table}) = \{ x_1 \} \). Similarly, we can extend the interpretation function \( I \) to the other scenes.

While the triplet format is suitable for the annotation of various action labels, the semantic representation in the form of FOL structures with scenes can be directly used in model checking and theorem proving for the VTE system developed in Suzuki et al. (2019). Our annotation format is flexible enough to be adapted in such applications.

5 Conclusion

We introduce a video-and-language dataset with human actions for multimodal inference. We annotate human actions in videos in the free format and collect 1,942 action categories for 200 videos. Our dataset contains various action labels for videos, including those predicates characteristic of videos such as predicates for utterance, predicates for intention and attitude, and aspetual predicates. In future work, we analyze recent action recognition models using Action Genome (Ji et al., 2020) with our dataset. We will also work on building a multimodal logical inference system between texts and videos.

Acknowledgment

This work was partially supported by JST CREST Grant Number JPMJCR20D2, Japan. Thanks to the anonymous reviewers for helpful comments. We would also like to thank Mai Yokozeki and Natsuki Murakami for their contributions.

References


Author Index

Baez Santamaria, Selene, 56
Baier, Thomas, 56
Bekki, Daisuke, 102
Brady, Michael, 95
Calixto, Iacer, 32
Dobnik, Simon, 45, 78
Du, Han, 95
Frank, Anette, 1, 32
Gatt, Albert, 32
Ginzburg, Jonathan, 21
Ilinykh, Nikolai, 45
Kennington, Casey, 89
Kim, Taewoon, 56
Krause, Lea, 56
Kruijt, Jaap, 56
Loáiciga, Sharid, 78
Luecking, Andy, 21
Mineshima, Koji, 102
Parcalabescu, Letitia, 1, 32
Schlangen, David, 78, 89
Suzuki, Riko, 102
Tonelli, Sara, 11
Trost, Nils, 1
Trotta, Daniela, 11
Vossen, Piek, 56
Yanaka, Hitomi, 102