Living a discrete life in a continuous world: Reference with distributed representations

Gemma Boleda¹  Sebastian Pado²  Nghia The Pham³  Marco Baroni⁴
¹Universitat Pompeu Fabra
²Institut für Maschinelles Sprachverarbeitung, Universität Stuttgart
³Center for Mind/Brain Sciences, University of Trento
⁴Facebook Artificial Intelligence Research, Paris
gemma.boleda@upf.edu, pado@ims.uni-stuttgart.de, thenghia.pham@unitn.it, mbaroni@fb.com

Abstract

Reference is the crucial property of language that allows us to connect linguistic expressions to the world. Modeling it requires handling both continuous and discrete aspects of meaning. Data-driven models excel at the former, but struggle with the latter, and the reverse is true for symbolic models.

We propose a fully data-driven, end-to-end trainable model that, while operating on continuous multimodal representations, learns to organize them into a discrete-like entity library. We also introduce a referential task to test it, cross-modal tracking. Our model beats standard neural network architectures, but is outperformed by some parametrizations of Memory Networks, another model with external memory.

1 Introduction

Language is both discrete and continuous, as exemplified by the phenomenon of reference (Frege, 1892; Abbott, 2010). When we refer to an object in the world using the noun phrase the mug I bought, we use content words such as mug, which are notoriously fuzzy or vague in their meaning (Van Deemter, 2012; Murphy, 2002). Once the referent for the mug has been established, however, it becomes a linguistic entity that we can manipulate in a largely discrete fashion, retrieving it and updating it with new information as needed (remember the mug I bought? My brother stole it; Kamp and Reyle (1993)). As recently discussed by Boleda and Herbelot (2016), data-driven methods such as distributional semantics and deep learning deal with continuous aspects of meaning very well (e.g., correctly categorizing a wide array of real-life mugs, or capturing the similarity between cups and mugs), but struggle with discrete aspects, such as the ability to create referents on the fly, update them with new information, and keep them distinct over time. Conversely, symbolic approaches such as Discourse Representation Theory (DRT, Kamp and Reyle (1993)) have been carefully crafted to capture discrete reference through constructs such as discourse entities, but do not handle continuous aspects well, which makes them brittle.

We present work in progress on a continuous model of reference that can emulate both continuous and discrete properties of language, learning from noisy data directly from reference acts. We further introduce an associated experimental task, cross-modal tracking, that addresses three of the main challenges of reference (Kamp, 2015): (i) Grounding language in the external world (e.g., linking the phrase the mug I bought to a particular physical mug), (ii) Combining visual and linguistically-conveyed information (you can recognize a mug by its visual properties, but you can store the information that I bought it even if you did not see me buy it), (iii) Tracking entities across time, keeping different entities distinct and recognizing when we are encountering the same entity again (my mug as of last night and my current mug are the same, and different from my sister’s mug).

2 Task and Data

Task. A simplified example of our cross-modal tracking task is shown in Figure 1. In the exposure phase, the model processes a sequence of exposures, each consisting of the image of an object together with one linguistic attribute that cannot be inferred from the image (e.g., the attribute instructed for the first barkeeper). Note that the object category (barkeeper, soldier) is not given. We present the same entities with different attributes at different time steps (e.g., the first barkeeper reap-
pears with the attribute amused). In the query phase, the model is presented with a purely linguistic query involving two attributes and an object (e.g., instructed and evaluated barkeeper), plus the set of all unique images presented before. The model must select the one image that satisfies the query. As (partially) shown in Figure 1, the input sequence contains distractors: A soldier sharing both attributes, a different barkeeper sharing only one of the attributes. The model must construct and update entity representations and keep them distinct based on their attributes (both visual and non-visual), addressing the three challenges listed above. Also, the model must handle both continuous features (categorization, by mapping across modalities to match images with nouns) and discrete aspects (identifying the one entity that matches the query).

The task is related to coreference resolution (Poesio et al., 2017), but focuses on identifying language-external objects rather than mentions of a referent in text; to Visual Question Answering (Antol et al., 2015), but it cannot be solved with visual information only; to Referring Expression Generation (Krahmer and Van Deemter, 2012), but it is identification rather than generation.

Dataset. Our dataset for the task contains 40,000 sequences for training, 5,000 for validation and 10,000 for testing. Our basis is a set of 2,000 object categories and 50 associated ImageNet images per category, sampled from a larger dataset used in Lazaridou et al. (2015).

We build a set of attributes for each object by first extracting the 500 most associated, and thus plausible, syntactic neighbors for the category according to the DM resource (Baroni and Lenci, 2010). This excludes nonsensical combinations such as repair:dog. We further retain only (relatively) abstract verbs taking the target item as direct object. This is because (a) concrete verbs are likely to have strong visual correlates that could conflict with the image (cf. walk dog); and (b) referential expressions routinely successfully mix concrete and abstract cues (e.g., the dog I own). We remove all verbs with a score over 2.5 (on a 1–5 scale) in the concreteness norms of Brysbaert et al. (2014).

We then construct each sequence as follows. First, we sample two random categories, and three random objects (distinct images) for each category (total: six entities). We then sample three attributes compatible with both categories, giving us three attribute sets of size two (a1+a2, a1+a3, a2+a3). We create a completely balanced set of exposures by pairing up the three entities of each category randomly with the three attribute sets. Since this process gives us two exposures for each entity (one with the first attribute, one with the second), it yields a sequence of twelve exposures. The query is a random combination of a category and two attributes, guaranteed to match exactly one entity.

3 The DIRE Model

The core novelty of our model, DIRE (for Diistributed REference), is a method to dynamically construct an entity library. Inspired by Joulin and Mikolov (2015) and Graves et al. (2016), we simulate discrete memory-building operations in a differentiable continuous setup. The entity library structure is updated after reading an input exposure by either creating a new entity slot for the exposure, or adding the exposure contents to an existing entity slot. This decision is based on the similarity between the current input and the entities already in the library. This generic mechanism (§ 3.1) can be applied in any setting that accumulates information about entities over time. We explain how we use it for our cross-modal tracking task in § 3.2.

3.1 Building the DIRE Entity Library

The input to the model is a set of subsequent exposures $x_1, x_2, \ldots, x_n$ represented by vectors $u_1, u_2, \ldots, u_n$. At the $t$-th exposure, the entity library is updated to state $E_t$ as follows. The first exposure vector $u_1$ is added to the entity library as $E_1 = u_1^T$. For $u_{i+1}$, we first obtain a similarity profile by taking its dot product with the entity vectors already in the library: $s_i = E_{i-1}^T u_i$ (note that $s_i$ has $i - 1$ dimensions). The maximum similarity to an existing entity, $s_i^{\text{max}} = \max(s_i)$, cues whether $x_i$ is an instance of an already-seen entity.

While we developed DIRE, Henaff et al. (2016) proposed a similar architecture; we leave a comparison to future work.
We transform \( s_i^{\text{max}} \) into \( p_i^{\text{old}} \), the probability that exposure \( x_i \) corresponds to an “old” entity, as follows (with the scalar \( w, b \) parameters shared across all exposures for \( i > 1 \)): \( p_i^{\text{old}} = \sigma(w s_i^{\text{max}} + b) \).

The entity library is updated by “soft insertion” (Joulin and Mikolov, 2015) of the current exposure vector \( u_i \) into the library. Concretely, we add the vector to each entity in the library, weighted by the probability that the current exposure is an instance of that entity. For the \( i - 1 \) existing entities, this probability is obtained by distributing the \( p_i^{\text{old}} \) mass across them, according to their probability of being the matching entity, conditional on the exposure being old. The latter probability is estimated by softmax-normalizing the similarity profile \( s_i \) from above. The probability that \( x_i \) is new is obviously \( 1 - p_i^{\text{old}} \). This results in the following distribution, where || stands for concatenation:

\[
z_i = p_i^{\text{old}} (\text{softmax}(s_i))(1 - p_i^{\text{old}})
\]

(1)

The entity library is then updated as follows:

\[
U_i = z_i u_i^T
\]

(2)

\[
E_i = (E_{i-1}|0) + U_i
\]

(3)

Where we inserted a 0 vector of the same dimensionality as the \( u_i \) vectors at the bottom of the library, initializing a blank slot to store a new entity. As a consequence, the library in its end state will always contain as many entity vectors as exposures. However, those inserted for exposures of old entities (that is, when \( p_i^{\text{old}} \approx 1 \)) will be near-0, and they could easily be removed from the library, along the lines of Henaff et al. (2016).

3.2 Cross-modal Tracking with DIRE

DIRE can be applied to the cross-modal tracking task as follows. Given pre-trained image and verbal attribute representations, we first derive a multimodal representation \( u_i \) for each exposure \( x_i \). We then update the entity library by processing the representations as explained in Section 3.1. The linguistic query is mapped in the same multimodal space where entities live, and the most relevant entity is retrieved. Finally, the images the model has to choose from (candidate set) are also mapped in multimodal space, and the correct answer is picked based on their similarity with the retrieved entity.

Multimodal Mapping. The original visual and verbal representations are linearly mapped onto the multimodal space. We share the same \( V \) projection across all images (in the exposures as well as in the candidate set), a single \( A \) projection for the verbal attributes (in the exposures and in the query), and a matrix \( C \) for the category name in the query. Composite representations are obtained by summing the multimodal representations of their parts.

Query and Retrieval. To select the best entity match for the query, we compute a “soft retrieval” operation inspired by Sukhbaatar et al. (2015). We obtain a query-to-entity similarity profile by taking softmax-normalized dot products of the query vector with each vector in the entity library. We then sum the latter, weighting them by the corresponding scalar entries in the similarity profile. Note that if only one entity is significantly similar to the query (so that the corresponding entry in the similarity profile tends to 1, while all other entries tend to 0), this is equivalent to retrieving that entity.

Picking the Right Image. Having retrieved an entity vector as outlined just above, we compute its softmax-normalized dot products to the images shown at query time. This will give us a probability distribution over the candidate set, that we can compare to the gold answer to obtain a standard cross-entropy cost to optimize the model.

The whole architecture is differentiable, allowing end-to-end training by gradient descent; in particular, the cross-modal mapping is learned as the model learns to refer. At the same time, it emulates operations like insertion and retrieval of entity representations, that, in frameworks such as DRT, are performed entirely in symbolic terms, and are manually coded in the DRT system Boxer (Bos, 2008). The model is rather parsimonious, with parameters limited to three mapping matrices (V, B, C) and the bias and weight terms for \( p^{\text{old}} \).

4 Experiments

Experimental details. Images are represented by 4096-dimensional vectors produced by passing images through the pre-trained VGG 19-layer CNN of Simonyan and Zisserman (2015) (trained on the ILSVRC-2012 data), and extracting the corresponding activations on the topmost fully connected layer. Linguistic representations are given by 400-dimensional \textit{cbow} embeddings from Ba-

\footnote{Note that the input vectors for images are only visual, and those for nouns and attributes are only textual.}

\footnote{We use the MatConvNet toolkit, \url{http://www.vlfeat.org/matconvnet}}
roni et al. (2014), trained on about 2.8 billion tokens of raw text. We map to a 1K-dimensional multimodal space. The parameters of DIRE are estimated by stochastic gradient descent with 0.09 learning rate, 10 minibatch size, 0.5 dropout probability, and maximally 150 epochs (here and below, hyperparameter values as in Anonymous 2016).

As competitors, we train standard feed-forward (FF) and recurrent (RNN) networks which have no external memory, using two 300-dimensional hidden layers and sigmoid nonlinearities. We also implement the related Memory Network model (MemN; Sukhbaatar et al. (2015)). Like DIRE, MemN controls a memory structure, but stores each input exposure separately in the memory. At the same time, MemN can perform multiple “hops” at query time. Each hop consists in soft-retrieving a vector from the memory, where the probing vector is the sum of the input query vector and the vector retrieved in the previous hop (null for the first hop). Conceptually, DIRE attempts to merge different instances of the same entity at input processing time, whereas MemN stores each piece of input separately and aggregates relevant information at query time. MemN can thus use the query to guide the search for relevant information. At the same time, it does not optimize the way in which it stores information in memory. Another difference with DIRE is that MemN uses two sets of mapping matrices: one to derive the vectors used at query time, the other for the vectors used for retrieval. We employ the same hyperparameters for MemN (also multimodal vector size) as for our model.

Results. Table 1 shows that our model, DIRE, outperforms the standard networks (FF and RNN) by a large margin, confirming the importance of a discrete memory structure in the reference tracking task. If we make the MemN architecture completely comparable to our model (with one matrix and one hop, MemN-1m-1h), our model achieves higher results (0.64 for DIRE-1m, 0.59 for MemN-1m-1h), which indicates that the basic architecture of the model holds promise. However, MemN outperforms DIRE when using two matrices, two hops (0.67 MemN-2m-1h/MemN-1m-2h vs. 0.65 DIRE-2m), or both (0.69 MemN-2m-2h). For MemN, this seems to be the upper bound, as increasing to three hops greatly harms results (see last row).

Further analysis suggests that DIRE successfully addresses two of the three challenges set out in the introduction: It learns to (i) ground linguistic expressions in images, (ii) combining visual and linguistically-conveyed information. Only for 8% of the datapoints does the model pick an image of the wrong category, and these are cases where confounders have visually similar or related categories to the target (cottage-chalet, youngster-enthusiast, witch-potion). Moreover, the similarity of the exposure to the query goes to near-zero when the attribute is wrong, even when the category is the same. However, (iii) the entity creation mechanism still needs to be fine-tuned, as currently DIRE creates a new entity vector for almost each exposure. More work is needed for this crucial part of the model.

|               | Random BL | FF    | DIRE-1m | MemN-1m-1h | MemN-2m-1h | MemN-1m-2h | MemN-2m-2h | MemN-1m-3h | MemN-2m-3h |
|---------------|-----------|-------|---------|------------|------------|------------|------------|------------|------------|
|               | 0.17      | 0.27  | 0.64    | 0.59       | 0.67       | 0.67       | **0.69**   | 0.30       | 0.30       |

Table 1: Tracking results (accuracy on test set).

5 Discussion

Providing a continuous model of reference that can emulate discrete reasoning about entities is an ambitious research programme. We have reported on work in progress on such a model, DIRE, which, unlike Memory Networks, aims at making decisions as to how to store the information at input processing time, in a way that aids further reasoning, namely, organizing it by entity. Results suggest that merging complementary aspects of DIRE and MemN could be fruitful.

Our project is related to several areas of active research. Reference is a classic topic in philosophy of language and linguistics (Frege, 1892; Abbott, 2010; Kamp and Reyle, 1993; Kamp, 2015); emulating discrete aspects of language and reasoning through continuous means is a long-standing goal in artificial intelligence (Smolensky, 1990; Joulin and Mikolov, 2015); grounding language in perception (Chen and Mooney, 2011; Bruni et al., 2012; Silberer et al., 2013), as well as reference and co-reference (Krahmer and Van Deemter, 2012; Poesio et al., 2017) are important subjects in Computational Linguistics. Our programme puts these different strands together.
Acknowledgments: We thank Iasonas Kokkinos, Germán Kruszewski, and other members of the CLIC lab of the University of Trento and the FAIR-Paris team for useful feedback. Also Angeliki Lazaridou for kindly providing us with the image vectors we use. This project has received funding from the European Research Council (ERC) under the European Union’s Horizon 2020 research and innovation programme (grant agreement No 715154; AMORE); Marie Skłodowska-Curie grant agreement No 655577 (LOVe); DFG (SFB 732, Project D10); and Spanish MINECO (grant FFI2013-41301-P). This paper reflects the authors’ view only, and the EU is not responsible for any use that may be made of the information it contains.

References

Barbara Abbott. 2010. Reference. Oxford University Press, Oxford, UK.

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. 2015. VQA: Visual Question Answering. In Proceedings of ICCV. Santiago de Chile, Chile.

Marco Baroni, Georgiana Dinu, and Germán Kruszewski. 2014. Don’t count, predict! a systematic comparison of context-counting vs. context-predicting semantic vectors. In Proceedings of ACL. Baltimore, MD, pages 238–247.

Marco Baroni and Alessandro Lenci. 2010. Distributional Memory: A general framework for corpus-based semantics. Computational Linguistics 36(4):673–721.

Gemma Boleda and Aurélie Herbelot. 2016. Formal distributional semantics: Introduction to the special issue. Computational Linguistics 42(4).

Johan Bos. 2008. Wide-coverage semantic analysis with Boxer. In Proceedings of the 2008 Conference on Semantics in Text Processing. Association for Computational Linguistics, pages 277–286.

Elia Bruni, Gemma Boleda, Marco Baroni, and Nam Khanh Tran. 2012. Distributional semantics in technicolor. In Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics. Jeju Island, Korea, pages 136–145.

Marc Brysbaert, Amy Beth Warriner, and Victor Kuperman. 2014. Concreteness ratings for 40 thousand generally known English word lemmas. Behavior Research Methods 46:904–911.

David Chen and Raymond Mooney. 2011. Learning to interpret natural language navigation instructions from observations. In Proceedings of AAAI. San Francisco, CA, pages 859–865.

Gottlob Frege. 1892. Über Sinn und Bedeutung. Zeitschrift für Philosophie und philosophische Kritik 100:25–50.

Alex Graves, Greg Wayne, Malcolm Reynolds, Tim Harley, Ivo Danihelka, Aignieszka Grabska-Barwińska, Sergio Gómez Colmenarejo, Edward Grefenstette, Tiago Ramalho, John Agapiou, Adrià Puigdomènech Badia, Karl Moritz Hermann, Yori Zwols, Georg Ostrovski, Adam Cain, Helen King, Christopher Summerfield, Phil Blunsom, Koray Kavukcuoglu, and Demis Hassabis. 2016. Hybrid computing using a neural network with dynamic external memory. Nature 538(7626):471–476.

Mikael Henaff, Jason Weston, Arthur Szlam, Antoine Bordes, and Yann LeCun. 2016. Tracking the World State with Recurrent Neural Networks. https://arxiv.org/abs/1612.03969.

Armand Joulin and Tomas Mikolov. 2015. Inferring algorithmic patterns with stack-augmented recurrent nets. In Proceedings of NIPS. Montreal, Canada.

Hans Kamp. 2015. Entity Representations and Articulated Contexts: An Exploration of the Semantics and Pragmatics of Definite Noun Phrases. Ms. University of Stuttgart.

Hans Kamp and Uwe Reyle. 1993. From Discourse to Logic: Introduction to Model-theoretic Semantics of Natural Language, Formal Logic and Discourse Representation Theory. Kluwer, Dordrecht.

Emiel Krahmer and Kees Van Deemter. 2012. Computational generation of referring expressions: A survey. Computational Linguistics pages 173–218.

Angeliki Lazaridou, Nghia The Pham, and Marco Baroni. 2015. Combining language and vision with a multimodal skip-gram model. In Proceedings of NAACL. Denver, CO, pages 153–163.

Gregory Murphy. 2002. The Big Book of Concepts. MIT Press, Cambridge, MA.

Massimo Poesio, Roland Stuckardt, and Yannick Versley. 2017. Anaphora Resolution: Algorithms, Resources, and Applications. Springer. In press.

Carina Silberer, Vittorio Ferrari, and Mirella Lapata. 2013. Models of semantic representation with visual attributes. In Proceedings of ACL. Sofia, Bulgaria, pages 572–582.

Karen Simonyan and Andrew Zisserman. 2015. Very deep convolutional networks for large-scale image recognition. In Proceedings of ICLR Conference Track. San Diego, CA. Published online: http://www.iclr.cc/doku.php?id=iclr2015:main.
Paul Smolensky. 1990. Tensor product variable binding and the representation of symbolic structures in connectionist networks. *Artificial Intelligence* 46:159–216.

Sainbayar Sukhbaatar, Arthur Szlam, Jason Weston, and Rob Fergus. 2015. End-to-end memory networks. In *Advances in Neural Information Processing Systems* 28, Montréal, Canada, pages 2440–2448.

Kees Van Deemter. 2012. *Not exactly: In praise of vagueness*. Oxford University Press.