# Simulating the acquisition of core semantic competences from small data

**Aurélie Herbelot**
Center for Mind/Brain Sciences
Dept. of Information Engineering and Computer Science
University of Trento
aurelie.herbelot@unitn.it

## Abstract

Many tasks are considered to be 'solved' in the computational linguistics literature, but the corresponding algorithms operate in ways which are radically different from human cognition. I illustrate this by coming back to the notion of semantic competence, which includes basic linguistic skills encompassing both referential phenomena and generic knowledge, in particular a) the ability to denote, b) the mastery of the lexicon, or c) the ability to model one's language use on others. Even though each of those faculties has been extensively tested individually, there is still no computational model that would account for their joint acquisition under the conditions experienced by a human. In this paper, I focus on one particular aspect of this problem: the amount of linguistic data available to the child or machine. I show that given the first competence mentioned above (a denotation function), the other two can in fact be learned from very limited data (2.8M token), reaching state-of-the-art performance. I argue that both the nature of the data and the way it is presented to the system matter to acquisition.

## 1 Introduction

Many tasks and datasets are considered solved problems in the computational linguistics literature. However, the data, training regimes and system architectures required to obtain top performance are unrealistic from the point of view of human cognition. Thus, state-of-the-art data-driven frameworks can be considered excellent engineering solutions to particular linguistic tasks, but they are not usable as 'models' of language acquisition, and thus of limited applicability to test hypotheses about human language.

This paper argues that core problems in computational linguistics should be 're-solved' – solved again – not as tasks, but as phenomena to *simulate*. This would involve a more careful attention to i) the type of data fed to the system; ii) the knowledge already in-built in its architecture; iii) the mode of learning implied by the training regime; iv) the specific features exploited by the learning process; and of course, v) the theoretical proposals explaining the phenomenon. Some of these desiderata have started being explored in the literature: the BlackBox NLP events, for instance, are currently fostering important discussions on the interpretability of artificial neural systems (Linzen et al., 2018, 2019). Still, the field remains far from satisfying all of them.

The work described in the following pages is a step towards the simulation of a particular phenomenon: the acquisition of core semantic competences. Its specific focus is on data: more particularly, the type and size of the corpus a system is exposed to. As we will see, talking about input data (desideratum i. above) naturally brings in questions about learning mechanisms (ii. and iii.), and about representation (iv.) Let us first note that an NLP system is typically exposed to at least hundreds of millions of words, if not billions. In contrast, a 3-year-old US child has only observed 25M words; a Mayan child of the same age will hear as little as 5M words (Cristia et al., 2017). In spite of the limited data they are exposed to, a child will reliably learn their language – this is referred to as the 'poverty of stimulus' in Chomsky's work.

If the stimulus is poor, we have to posit the existence of extra cognitive mechanisms to compensate for the lack of explicit linguistic evidence. For human syntax, Chomsky famously advocated the existence of an innate Universal Grammar. I argue that there is an equivalent question to be asked in machine learning: indeed, the architecture (ii.) and hyperparameters (iii.) of a system, as well as the specific representation of the input data (iv.), are 'innate' features which are important to make



*Proceedings of the 24th Conference on Computational Natural Language Learning*, pages 344–354
Online, November 19-20, 2020. 
https://doi.org/10.18653/v1/P17

explicit when describing a 'data-driven' system.

In what follows, I investigate a particular configuration of a semantic acquisition model. Specifically, I ask whether a particular type of input, based on individual grounded entities, can make up for data sparsity. Following this hypothesis, I propose a model nicknamed EVA (Entity Vector Aggregator)[1] and compare it to the behaviour of a character-based language model with no access to referential information. I perform a battery of tests including similarity, compatibility and acceptability judgements, as well as lexical relation categorisation, and demonstrate that when fed with the right data and the right representation, the model learns core semantic competences from as little as 2.8M words.

## 2 Semantic competence in the linguistic literature

The notion of linguistic *competence* was introduced by Chomsky *Aspects of the theory of syntax* (Chomsky, 1965): competence is 'knowing one's language', and it must be distinguished from performance, 'using one's language'. According to Chomsky, the study of linguistics is the study of competence. The linguist should try and elucidate the underlying structure of the *mental* phenomenon that leads to observable performance.

In syntax, competence is usually defined in terms of grammaticality. The semantic equivalent is more difficult to pinpoint, and various proposals have been made. We will focus on three major positions in this paper: semantic competence as mastery of a) the lexicon; b) reference; c) language use.

**1. Mastery of the lexicon:** following Chomskian grammar, Katz and Fodor (1963) propose that the goals of semantics can be obtained by *"subtracting grammar from the goals of a description of a language"* (p172). According to them, this subtraction results in elements of lexical semantics, including relations such as hyponymy or antonymy, as well as word senses. Semantic competence is then the ability to say that *The paint is silent* is not felicitous, that *The bill is large* is ambiguous, or again that *There are two chairs in the room* entails *There are at least two things in the room.*

**2. Ability to refer:** coming from formal semantics, Partee (1979) investigates the notion of a 'godly' speaker, who would have perfect ability to match words to extensions, and argues such a speaker might embody (intensional) semantic competence. She however also identifies logical issues with that notion, in particular with respect to propositional attitudes. In Partee (2014), she offers a compromise which recognises the important relation between linguistic constituents and external reality, but also admits that language users can be mistaken or simply ignorant when it comes to truth-theoretic judgements.

**3. Distributional consistency:** Kripke (1972) argues for a 'causal theory of reference', which posits that people use words in the way that they have seen other people use it. Competent usage follows from simple exposure to performance data, without assuming fully competent extensional knowledge: For instance, having heard *Frege came to dinner* from some speaker, a competent listener might ask ***Who** is Frege?*, having understood that Frege is a person, but being unable to identify that person in the world (see also Putnam, 1975 for a related argument). Seen from a statistical perspective, this position boils down to an idea of distributional consistency, that is, the belief that speakers model their language use on others. A notion of acceptability derives from the theory (i.e. it would be incorrect to ask ***What** is Frege?*), but in a way that is different from the felicity conditions posited by Katz and Fodor (1963): while Katz and Fodor assume that felicity comes from the rules of the lexicon, the Kripkian account implies that it emerges from the language use following an initial reference act.

This paper starts from the assumption that all three definitions should be satisfied to speak of semantic competence. That is, I will posit that we need meaning representations that allow us to denote (to satisfy 2), for which we have descriptions or referring expressions by actual language users (to satisfy 3), and over which we can learn lexical relations (to satisfy 1). To achieve this, I will hypothesise a semantics based on *instances* (which can be aggregated into sets in a formal semantics fashion), but represented in terms of the statistical properties of language use. I will propose a representation which satisfies both requirements in §4.

## 3 (Small) data

The input data we will work with is a set of grounded 'utterances' extracted from annotations in the Visual Genome (VG) dataset (Krishna et al.,

[1]The code for EVA is freely available at https://github.com/minimalparts/EVA.

2017). This annotated set displays several important properties. First, it is *small* (around 2.8M tokens), so compatible in scale with the limited data a learner is exposed to. Second, while it does not quite correspond to the type of sentences a child might be exposed to, it has some similarities with a realistic 'early' linguistic diet: the simple image annotations can be regarded as utterances of the type *Look! The dog is playing with the ball.* Third, it encodes the particular representational aspects we want to investigate: it is anchored in a clear notion of grounded instances (the individual objects in an image) and corresponding language use (the human-generated captions/annotations associated with each bounding box).

The VG itself consists of a set of 108,077 images annotated with 5.4M region descriptions as well as 3.8M object referents,[2] 2.8M attributes and 2.3M relationships. All objects are associated with a unique identifier, meaning that we can use such identifiers as a set of object variables for the particular universe defined by the VG.

I follow the methodology introduced by Kuzmenko and Herbelot (2019), who extract information about VG instances and use it to create a 'set-theoretic' vector space. The example below shows a subset of the annotation for image ID 1, after some initial pre-processing of the data. I assume that each image corresponds to some 'situation', in the spirit of Young et al. (2014). So situation 1 contains a tall brick building, identified by variable 1058508, on which we find a black sign, identified by variable 1058507. *Object types* are recognisable through their suffix (e.g. $building.n$, $sign.n$), *attributes* consist of all other one-place predicates (e.g. $tall$, $made|of|bricks$); and *relationships* consist of all two-place predicates (e.g. $on$).

```
<situation id=1>
    ...
    <entity id=1058508>
        building.n(1058508)
        tall(1058508)
        brick(1058508)
        made|of|bricks(1058508)
        on(1058507,1058508)
    </entity>
    <entity id=1058507>
        sign.n(1058507)
        black(1058507)
        on(1058507,1058508)
    </entity>
    ...
</situation>
```

[2]In the VG, object referents are associated with WordNet synsets. For simplicity, I collapse all WordNet senses together, but this has hardly any effect on the size of the object referents' set which, including sense annotations, would amount to 1203 unique types vs 1188 when ignoring sense.

We can straightforwardly obtain shallow logical forms associated with each situation, e.g.:

$$\text{building.n}'(1058508), \text{tall}'(1058508), \text{brick}'(1058508),$$
$$\text{sign.n}'(1058507), \text{black}'(1058507),$$
$$\text{on}(1058507, 1058508)$$

For simplicity (and because each entity only occurs once in VG), I transform two-place predicates into two one-place predicates: e.g. $on(1058507, 1058508)$ becomes $on(1058507, \text{building.n}')$, $on(\text{sign.n}', 1058508)$, respectively denoting the set of things that are on buildings, and the set of things that signs are on.

The provided annotations together with the associated objects, attributes and relations can be taken to be a *partial* description of some subset of the real world (i.e. the subset encapsulated by the images). This can be illustrated by considering the following two instances of bear (objects referents 158539 and 1617277), together will all their annotated relations:

| | |
|---|---|
| 158539 | bear.n has(-,eye.n) has(-,claw.n) has(-,paw.n) has(-,mark.n) beside(grass.n,-)has(-,ear.n) on(-,land.n) has(-,leg.n) has(-,nose.n) |
| 1617277 | bear.n has(-,fur.n) has(-,nose.n) |

We see that two instances can be annotated with different degrees of granularity in the VG. The first instance above includes many more details about the physical appearance of the bear, although the second includes the relation 'has fur', which is missed by the first one. That is, we have two different 'experiences' of bears, associated with utterances which, in a realistic situation, could have come from the learner's carer (*'Look at the bear next to the grass, look at its claws!'*) This is a typical example of the 'poverty of the stimulus' effect: the performance data associated with those instances of bear is both *incomplete* (the linguistic data only describes part of the bears) and *inconsistent* (the two descriptions are very dissimilar).

In order to fully exploit the information in the VG, the annotated *attributes* and *relationships* are supplemented with a third type of linguistic information: simple extensional co-occurrences are computed, thus modelling an implicit logical *and* (the comma in the shallow logical form). I.e., if a bear occurs in an image under a cloudy sky, the model registers the co-occurrence of a bear entity with a sky entity. In what follows, I refer to such implicit relations under the general term of *situational co-occurrences*, to express the fact that the

extensional co-occurrence takes place within a single situation.

## 4 Models

In the field of computational linguistics, we often take models to be 'algorithms', independently of the data they are trained on, and often, independently from the assumptions that the algorithm is built upon. But as pointed out in the introduction to this paper, what is in the data, what is inbuilt in the algorithm, and how the data is presented to the learning process determines the extent to which one can speak of a scientific model of such or such phenomenon. Therefore, I will talk of a 'model' as a *combination* of a particular system / algorithm (with its specific assumptions) and a particular type of data.

In what follows, in the spirit of fixing the learning mechanism as much as possible, I present three models based on very similar algorithms (variants of skip-gram language models). I however vary the data input into the system, both in size and representation.

**Pretrained FastText (FT):** The first model under consideration is a pre-trained, state-of-the-art set of vectors, generated with FastText (Bojanowski et al., 2017). The system is a character-based language model and thus unsuitable for encoding extensions (that is, it will not satisfy the ability to refer in our set of semantic competences). However, it provides a helpful upper-bound for the tasks that language models excel at. The FastText vectors[3] were obtained from training over 16B tokens from a Wikipedia snapshot, the UMBC webbase corpus (Han et al., 2013) and statmt.org news dataset (Mikolov et al., 2018).

**FastText trained on VG (FTVG):** Being based on simple character ngrams, FastText is well suited to learning from smaller data (Mikolov et al., 2018). A FastText model is trained with default settings on a portion of the Visual Genome's 5.4M region descriptions. Such descriptions are short phrases or sentences of the type *man wearing red and black surf apparel* or *Red bus has advertisements that says 123 Current Account Santander*. From those descriptions, 2.8M tokens are used to match the size of the next system's background data (see 'EVA' below). FTVG differs from FT not only with respect to the size but also the presentation of its data: while FTVG is exposed to raw utterances like FT, those utterances are broken down by instance (the data contains one description per line, so a target word is only ever found in contexts that pertain to the same instance).

**EVA:** Finally, a third model is proposed. Nicknamed EVA (Entity Vector Aggregator), it is generated straight from the extensional information contained in the VG annotations (the attributes, relationships and situational co-occurrences described in §3). Before being fed to the skip-gram, the data is converted into a form akin to a set-theoretic vector space, using the procedure below.

First, let $P_L$ be the predicates in some logic and $U$ the entities in some universe. Let us define a vector space model by using some interpretation function $||.||$ to return the denotations of $P_L$:

$$||.|| : (P_L \cup U)^* \rightarrow ((P_L \times U) \rightarrow \{0, 1\})$$

An example of such a vector space is shown on the left of Fig 1. I will refer to it as an *entity matrix*: each predicate is associated with a point expressed in terms of a vector basis $U$ (so each dimension corresponds to an entity). The point is a straightforward representation of the extension of the predicate, and shows the entities that the predicate is true of. For instance, following the first row of the matrix, we find that the set of bears in our toy space is $\{x1, x2\}$.

We can then define an aggregation function $A_D$ which groups context elements by predicate (e.g. all objects that are bears are aggregated into a single bear$'$ vector by pointwise addition):

$$A_D : ((P_L \times U) \rightarrow \{0, 1\}) \rightarrow ((P_L \times P_L) \rightarrow \mathbb{N}_0)$$

This operation results in a vector space such as the one shown on the right of Fig 1. I will refer to it as a *predicate matrix*, since the basis is now made of the predicates in $P_L$.

An entity and predicate matrix are built for the VG, using the following restrictions. We ignore objects which are not annotated with any attribute or relation and would result in $\vec{0}$ vectors, thus obtaining around 2M entities. Further, the entity matrix is constructed for predicates with frequency over 100. The result of this pre-processing is a 2M $\times 8284$ matrix, where the predicates include 1188 object types, 798 attributes and 6283 relationships.

[3] Freely available at https://fasttext.cc/docs/en/english-vectors.html.

| | x1 | x2 | x3 | x4 | x5 | x6 |
|---|---|---|---|---|---|---|
| bear' | 1 | 1 | 0 | 0 | 0 | 0 |
| white' | 1 | 0 | 0 | 0 | 0 | 0 |
| black' | 0 | 1 | 0 | 0 | 0 | 0 |
| tree' | 0 | 0 | 1 | 1 | 1 | 1 |
| old' | 0 | 0 | 1 | 1 | 0 | 0 |
| young' | 1 | 1 | 0 | 0 | 1 | 1 |

| | bear' | white' | black' | tree' | old' | young' |
|---|---|---|---|---|---|---|
| bear' | 2 | 1 | 1 | 0 | 0 | 2 |
| white' | 1 | 1 | 0 | 0 | 0 | 1 |
| black' | 1 | 0 | 1 | 0 | 0 | 1 |
| tree' | 0 | 0 | 0 | 4 | 2 | 2 |
| old' | 0 | 0 | 0 | 2 | 2 | 0 |
| young' | 2 | 1 | 1 | 2 | 0 | 2 |

Figure 1: **Left:** an entity matrix, showing the entities that a predicate is true of. **Right:** the corresponding predicate matrix, after aggregation with function $A_D$. The first row is simply the pointwise addition of the first two columns in the entity matrix (the two bear entities).

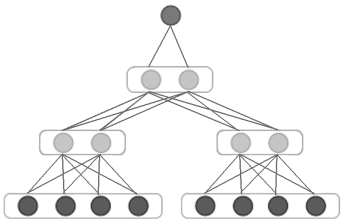
We can compare the figures above to the size of the large FT pretrained model by counting the number of unique tokens in the Visual Genome data, where 'unique' means that the token – whether object type, attribute or relationship – appears with a specific entity. Since two-place predicates are transformed into two one-place predicates, the token is incremented for each argument separately (e.g. *tree(3787077)* is one token but *parked-on(1058515,1058539)* gives two tokens). This comes to 1,590,861 tokens for one-place predicates (object type and attributes) and 1,224,582 tokens for two-place predicates (relationships), thus around 2.8M tokens in total.[4] So EVA is exposed to around 5700 *less* data than FT. It however has the advantage of being grounded in a clear notion of entity, thus matching the type of situated speech that forms most of a child's diet (Clark, 2009). Further, the corpus size is in line with the number of tokens that a child might get directed at them in around a year of early life.

The challenge for both FTVG and EVA is to deal with the poverty of the stimulus. It is worth recalling that Landauer and Dumais (1997) suggested a solution to the problem which involved the use of Principal Component Analysis (PCA) as a dimensionality reduction method over a distributional matrix. The use of PCA was meant to capture the main axes of variance over the limited data given to the model, allowing for fast generalisation. Later models of distributional semantics, in particular neural architectures (e.g. the skip-gram of Mikolov et al., 2013), do not explicitly mention dimensionality reduction as a way to successfully generalise over insufficient data, but the intuition remains implicit in the choice of dimensionality of the embedding layers. FTVG can rely on this mechanism, as well as its character-ngram model. EVA needs its own way to surmount the issue, and because it encodes extensional information, it has to deal with the poverty of the stimulus not only at the level of the linguistic input, but also at the denotation level. (As we have seen before, the VG is in no way an exhaustive and accurate representation of the world.) In other words, we want EVA to learn word embeddings at reduced dimensionality like its competitors, but from co-occurring *extensions*.

The implementation of EVA's embedding function is extremely simple and does away with some of the hyperparameters used by the original skip-gram model. It takes a predicate matrix of size $m \times m$, as would be produced by the aggregation function $A_D$, subsamples the counts in that matrix (lowering very frequent counts), and performs a prediction task. That task consists in predicting whether a 'target' predicate (from the rows of the matrix) and a 'context' predicate (from the columns of the matrix) have been seen together in the description of a unique, grounded entity. A 'positive' example for the target bear′ might be the predicate brown′ (some bear entity has been seen to be brown). A 'negative' sample for the same target bear′ might be transparent′. Positive samples are taken in shuffled order from the subsampled matrix, while negative samples are randomly chosen amongst the zero values of the matrix. As in the original skip-gram with negative sampling, embeddings for each predicate are first randomly initialised and fine-tuned in the process of doing the prediction task. A dimensionality of 300 gives optimal results in preliminary experiments over our development set. Therefore, results shown in the next sections are for this dimensionality only.

We use two main approaches for testing representations: a) for the similarity task, as is standard in the literature, we directly compute the relative position of embeddings in the space using the cosine metric; b) for other tasks, we use the vectors as input to a very simple feedforward neural net architectures, learning dedicated weights for each task

[4]Note that in the original VG annotation, region descriptions are not fully aligned with object / attribute / relation annotations and typically contain more information. So the data given to FTVG and EVA may present slight variations.



$$
\begin{aligned}
a_{11} &= RELu(x_{11}W_{11} + b_{11}) \\
a_{12} &= RELu(x_{12}W_{12} + b_{12}) \\
a_2 &= RELu([a_{11}, a_{12}]W_2 + b_2) \\
\hat{y} &= a_2 W_3
\end{aligned}
$$

Figure 2: A simple neural architecture.

over frozen background representations. The architecture used across tasks is shown in Fig. 2. It consists of an input taking two word vectors, mapped to a hidden layer with reduced dimensionality. All hidden layer representations are then concatenated and passed through a second hidden layer which is then fed to the output layer. The output layer may consist of a single node or several, depending on whether the task requires regression or classification. A RELu non-linearity is applied to the input layer and first (concatenated) hidden layer. A softmax is used on the last layer for classification tasks.

The training regime for all three models is as follows. A single grid search is performed over the hyperparameter space, using 200 iterations of Bayesian optimisation[5] with early stopping. For EVA, I follow results by Kuzmenko and Herbelot (2019) showing that linguistic phenomena are not all modelled by the same feature types in the VG. The validation data is used to select the best combination of feature types for a task (attributes, relations, situational co-occurrences), running the hyperparameter optimisation over all possible combinations. For all models, the five best hyperparameter sets are then selected according to validation results, and their stability is checked by performing 10 extra validation runs on each set, yielding 10 models per combination. The 10 models corresponding to the best average score are then applied to the test set and an average score is reported over the test data, together with standard deviation.

[5]This step uses the package available at https://github.com/fmfn/BayesianOptimization. Hyperparameters are optimised in the following ranges: learning rate and regularisation, $[0.001 - 0.01]$; epochs, $[100 - 500]$; minibatch size, $[16 - 1024]$; size of hidden layer, from 100 to initial vector size for FastText, and EVA; and in the range $50 - 100$ for FTVG.

## 5 Evaluation procedure

The entity matrix is evaluated in terms of the three aspects of competence we discussed in §2: knowledge of core lexical relations (Katz and Fodor, 1963), knowledge of 'acceptable' use of a term (again, Katz and Fodor, 1963, but also Putnam, 1975; Kripke, 1972), and of course, ability to retrieve the extension of a term (Partee, 1979).

**Lexical relations:** the models are evaluated on three different datasets encoding different aspects of lexical knowledge, namely the relation of *similarity*, the ability of the model to *classify specific relations* such as hyponymy or meronymy, and finally the relation of *incompatibility*. First, *Similarity* is evaluated against SimLex-999 (Hill et al., 2015), a set of 999 pairs meant to capture similar rather than merely related items. The second test is to evaluate the ability of the model to distinguish between particular *relations*, as encoded in the BLESS dataset (Baroni and Lenci, 2011). BLESS contains 26554 pairs annotated for hyponymy, meronymy, co-hyponymy, attribute and event relations (an additional class is included for the absence of relation and is marked as 'random'). Finally, the models are fed the *incompatibility* dataset of Kruszewski and Baroni (2015). This dataset contains 17973 word pairs associated with a compatibility judgement elicited from human annotators, on a scale from 1 to 7. So for instance, the pair *airplane-baby* has a mean score of 1 (fully incompatible), *dessert-vegetable* a score of 3 (somewhat compatible) and *airplane-jet* a score of 6.6 (close to full compatibility). All datasets are pre-processed to only keep the instances containing words present in the VG corpus, thus reducing the size of each available resource. The three models are evaluated on the same data.

The overall number of tested instances is shown for each dataset in Table 1, as well as the splits between training, validation and test sets. Note that SimLex-999 is evaluated in the standard fashion, by computing cosine distance between vectors in the space, with no further training involved. The data is nevertheless split into validation and test sets to allow for the selection of the best set of features for EVA at validation stage (out of the attributes, relationships and situational co-occurrences). To

| Dataset | # Instances post-filtering | Train | Val | Test |
|---|---|---|---|---|
| SimLex-999* | 169 | - | 100 | 69 |
| BLESS | 1764 | 1200 | 300 | 264 |
| Compatibility | 2074 | 1500 | 300 | 274 |
| Acceptability | 1030 | 700 | 200 | 130 |

Table 1: Number of instances left in datasets after filtering against VG vocabulary. Splits into train, validation and test sets are shown. Due to the small number of instances in SimLex-999, systems are evaluated 10 times on that dataset, using 10 random splits.

confirm robustness of the reported results, systems are run over 10 random splits of the 169 instances in the dataset, and average correlations are reported.

**Acceptability:** there are various datasets for acceptability / plausibility judgements (e.g. Vecchi et al., 2017; Wang et al., 2018), but one is needed which contains a fair number of concrete nouns, to match the VG data. The compound dataset of Graves et al. (2013) fulfils this requirement: it consists of 2160 compound nouns annotated by humans on a scale of 0 to 4, made of 500 concrete nouns. Half of the compounds are attested collocations like *television chef*, while the others are unattested, like *bike barn* or *book puppy*. Again, the data is filtered to keep only the pairs containing words included in the VG dataset.

Let us note here that the acceptability task is interestingly different from learning the incompatibility relation, whilst sharing some aspects with it. The nouns tested for incompatibility in the previous section (e.g. *zebra - woman*) represent labels which may or may not denote the same sets: the task is extensional in nature. The acceptability task, on the other hand, tests to what extent a speaker might generate a plausible interpretation for a given compound noun. This involves inferring a tacit relation between the nouns. So for instance, *lawn guy* is judged fairly acceptable by humans (average score of 3.464 out of 4), presumably because a lawn guy might be the guy who is standing on the lawn, or the guy who normally mows the lawn, etc.

**Extensions:** reference is deterministically encoded in EVA. To make this clear, the next section provides illustrative examples of composition over VG categories. It also shows how referents are retrieved by the model and how dimensions can be aggregated to quantify over instances of subkinds.

## 6 Results

This section contains results obtained on the validation and test portions of our datasets (see Table 1 for data splits).

Table 2 shows how EVA's performance on the validation sets depends on the combination of VG feature types used in training. Various observations can be made with regard to the results, starting with the most striking effect: SimLex-999 is extremely sensitive to data type. The similarity dataset shows correlations between $0.14$ (when using situational co-occurrence only) and $0.39$ (when using attributes and relations). In general, it is clear that using situational co-occurrences is detrimental to the performance of the system. This is to be expected, since the similarity evaluation is geared towards identifying taxonomic siblings (e.g. *cat, dog*: kinds that are structurally similar) rather than related items (e.g. *cat, meow*: kinds or events that might co-occur in the same situations).

Other datasets are less affected by feature selection but still show a preference for certain inputs. Notably, BLESS performs at its best when using situation information. This is perhaps due to the distinctions that the model has to perform between classes such as taxonomic siblings, meronyms and 'other' relations. Meronymy, in particular, requires to distinguish between items that simply co-occur in a situation (*cat* and *garden*) and those that co-occur but are also part of a relation (*cat* has-a *paw*). Finally, relations seem crucial to get best performance on the incompatibility dataset.

Moving to the test set, we only retain the models with highest performance on the validation data (*Att+Rel* for SimLex-999, *Sit* for BLESS, *Att* for acceptability and *Att+Sit* for incompatibility). Overall results are provided in Table 3 for all three models (FT, FTVG and EVA), and discussed below.

**Lexical relations:** On the *similarity* task (SimLex-999), EVA outperforms FTVG by 10 points and lags behind the huge pre-trained FT by only one point. The *classification of lexical relations* (BLESS) is achieved by all systems with high accuracy, without significant differences. Finally, performance on *incompatibility* is slightly over state-of-the-art level for both systems trained on the VG. EVA gives the best overall score, outperforming pretrained FT by two points. In other words, the system built on denotations is the overall winner when considering lexical

| | Att | Rel | Sit | Att+Rel | Att+Sit | Rel+Sit | Att+Rel+Sit |
|---|---|---|---|---|---|---|---|
| SimLex ($\rho$) | 0.33 $\pm$ 0.04 | 0.38 $\pm$ 0.04 | 0.14 $\pm$ 0.05 | **0.39** $\pm$ 0.04 | 0.16 $\pm$ 0.05 | 0.25 $\pm$ 0.05 | 0.24 $\pm$ 0.04 |
| BLESS (acc.) | 0.89 $\pm$ 0.00 | 0.89 $\pm$ 0.00 | **0.92** $\pm$ 0.01 | 0.91 $\pm$ 0.00 | 0.91 $\pm$ 0.00 | 0.91 $\pm$ 0.00 | 0.91 $\pm$ 0.00 |
| Accept. ($\rho$) | **0.50** $\pm$ 0.01 | 0.47 $\pm$ 0.01 | 0.47 $\pm$ 0.01 | 0.48 $\pm$ 0.02 | 0.48 $\pm$ 0.01 | 0.49 $\pm$ 0.02 | 0.46 $\pm$ 0.01 |
| Incompat. ($\rho$) | 0.42 $\pm$ 0.02 | 0.45 $\pm$ 0.03 | 0.43 $\pm$ 0.01 | **0.47** $\pm$ 0.04 | 0.42 $\pm$ 0.02 | 0.45 $\pm$ 0.02 | 0.45 $\pm$ 0.03 |

Table 2: EVA performance on validation set, for different combinations of feature types. The figures shown are averaged over 10 runs.

| | Corpus size | SimLex $\rho$ | BLESS acc. | Incompatibility $\rho$ | Acceptability $\rho$ | Reference |
|---|---|---|---|---|---|---|
| FT | 16B | **0.39** $\pm$ 0.08 | **0.87** $\pm$ 0.01 | 0.43 $\pm$ 0.04 | **0.59** $\pm$ 0.02 | $\times$ |
| FTVG | 2.8M | 0.28 $\pm$ 0.12 | 0.86 $\pm$ 0.01 | 0.44 $\pm$ 0.06 | 0.58 $\pm$ 0.01 | $\times$ |
| EVA | 2.8M | 0.38 $\pm$ 0.10 | **0.87** $\pm$ 0.01 | **0.45** $\pm$ 0.04 | 0.56 $\pm$ 0.02 | ✓ |

Table 3: Test results on all datasets.

competence, despite being trained on very scarce data.

**Semantic acceptability:** This time, we see that FTVG slightly outperforms EVA ($\rho = 0.58$ vs $\rho = 0.56$), possibly by virtue of being a language model and thus more suited to encoding word usage, in the sense of 'distributional consistency' (see §2). It is nevertheless striking that minimal training over data which encodes no surface information achieves very reasonable performance, in the range of pretrained FT ($\rho = 0.59$). This can be taken as confirmation that acceptability *can* be learned successfully from an extensional representation.

**Extensions:** To complete the above results, let us recall that EVA encodes reference by default, since the raw entity matrix (before aggregation and dimensionality reduction) captures how predicates are associated with entities. Denotations are therefore returned fully deterministically. To illustrate this, I give here an example of basic intersective composition in the VG model. Given the entity matrix, set intersection is simply expressed as pointwise multiplication. For instance, the denotation of the phrase *brown bear* can be obtained by multiplication of the *bear* and *brown* entity vectors. The operation returns brown bear entities in the Visual Genome with their other properties, as exemplified below:

| | |
|---|---|
| 5460844 | bear.n.01, brown, large, adult, big, with(-,bear.n.01) |
| 5464728 | bear.n.01, brown, furry, shaggy, fuzzy, splashing, posing, big, in(-,water.n.01) |
| 4868617 | bear.n.01, brown, wearing(-,jean.n.01), on(-,pillow.n.01), holding(baby.n.01,-) |

...

Given the entity matrix, it is possible to multiply any number of vectors to obtain denotations for, say, 'playing brown bears', 'playing white bears', or 'cute teddy bears', and inspect the corresponding subspaces (that is, the basis made of the individuals in the denotations). In those subspaces, only the vectors corresponding to annotated properties for the respective sets are non-zero. For instance, there are five playing white bears in the VG, forming a 5-dimensional subspace with 38 non-zero property vectors. Quantification can be defined for particular restrictors (subsets of playing white bears) and scopes (the property vectors) by aggregating individuals into a 1-dimensional basis representing a subkind and reading set overlap relations off the normalised version of that basis.

To illustrate this, let us consider the three subkinds 'playing white bears', 'playing brown bears' and 'cute teddy bears'. For each subkind, having applied intersective composition to the vectors in the entity matrix by pointwise multiplication, we obtain a denotation vector which can be aggregated using $A_D$. The result of such operation is shown in Fig 3, with some relevant property vectors. Following normalisation, the weight of a vector along a dimension can be read as the probability of an instance of the set represented by the dimension to have the property of the vector. So for instance, there is a $0.6$ probability that a playing white bear is in water, versus a $0.36$ probability for playing brown bears in the VG data. While being preliminary, such observations about the behaviour of the EVA representations indicate that it could encode a number of important set-theoretic properties, making it properly compatible with formal semantics approaches.

## 7 Conclusion

This paper made the case for solving existing tasks with models more in line with cognitive reality, ar-

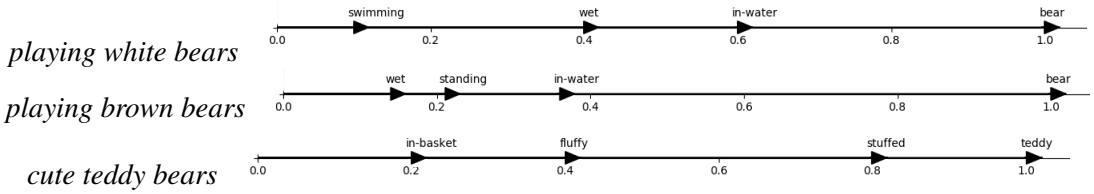



Figure 3: Result of intersective composition over EVA vectors.

guing in particular for the use of smaller corpora. Coming back to the points made in our introduction, I summarise the contributions of the model with respect to desirable aspects of a language acquisition 'simulation'.

First, we have seen that systems trained on 2.8M tokens of VG data are competitive with a large language model pretrained on 5700 times more data when tested on core lexical tasks. On the similarity task, the reference-based system EVA considerably outperforms a language model trained on the same amount of data, emerging as the best 'small data' model.

Notably, EVA exploits a very specific presentation of the data, capitalising on its access to individual instances and its ability to choose the semantic information relevant for solving a given task. Recent work has argued that language modeling is not enough for Natural Language Understanding, and in particular that the relation between language and world(s) matters to comprehension (Bender and Koller, 2020). The results presented here support this view: not only are instances the basic building blocks of reference, but they might also be crucial to support the acquisition of lexical competences.

With respect to the 'innate' mechanisms of the new model presented here, several shortcomings must be pointed out. First and foremost, EVA assumes the availability of a denotation function – some oracle able to map words to entities. This is of course something that children actually have to learn in the process of acquisition, and which probably proceeds in parallel with the training of other semantic competences. Ideally, this assumption should be relaxed in future versions of the model to understand how much the system learns when its reference module is imperfect (for instance, by linking EVA to an object recognition system from the Language and Vision literature).

Further, the learning mechanisms involved in EVA may be too generic. We have used simple co-occurrence prediction for the acquisition of word semantics and non-linear regression/classification for task-specific competences, which goes well with claims that language can be acquired via generic cognitive functions. But the actual training regime used by those mechanisms may not be as plausible as it could be. In particular, it is unclear how much supervision is involved in the human acquisition of skills such as lexical relation recognition or acceptability judgements (see e.g. Saxton, 2000 on the amount of negative input received by children from their carer). It is for instance dubious to argue that meronymies should be learned in a supervised fashion, with the learner being explicitly told that an ear or a whisker is 'part of' a cat. This aspect will have to be investigated further before claiming plausibility of the model.

Finally, the data used for our experiments is currently anchored in a visual dataset, and is therefore focused on concrete entities. Linguistic competence involves mastery of abstract vocabulary, as well as reference to 'possible worlds', which can be different from the universe we perceive. It remains to be seen how the competences acquired over purely perceptual data can be usefully brought into skills that involve abstraction and modality.

## Acknowledgments

I thank Ann Copestake and Katrin Erk for reading an early draft of this paper, as well as the participants to the GeCKo workshop in Barcelona for their helpful comments. I would also like to thank the anonymous reviewers for their helpful suggestions and comments. Finally, I gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan V GPU used for this research.

## References

Marco Baroni and Alessandro Lenci. 2011. How we blessed distributional semantic evaluation. In *Proceedings of the GEMS 2011 Workshop on GEometrical Models of Natural Language Semantics*, pages 1–10. Association for Computational Linguistics.

Emily M Bender and Alexander Koller. 2020. Climb-

ing towards NLU: On meaning, form, and understanding in the age of data. In *Proceedings of the 58th annual meeting of the Association for Computational Linguistics (ACL 2020)*, Seattle, United States.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Noam Chomsky. 1965. *Aspects of the theory of syntax*. MIT Press.

Eve V Clark. 2009. *First language acquisition*. Cambridge University Press.

Alejandrina Cristia, Emmanuel Dupoux, Michael Gurven, and Jonathan Stieglitz. 2017. Child-directed speech is infrequent in a forager-farmer population: A time allocation study. *Child development*.

William W Graves, Jeffrey R Binder, and Mark S Seidenberg. 2013. Noun–noun combination: Meaningfulness ratings and lexical statistics for 2,160 word pairs. *Behavior research methods*, 45(2):463–469.

Lushan Han, Abhay L Kashyap, Tim Finin, James Mayfield, and Jonathan Weese. 2013. Umbc_ebiquity-core: Semantic textual similarity systems. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 1: Proceedings of the Main Conference and the Shared Task: Semantic Textual Similarity*, volume 1, pages 44–52.

Aurélie Herbelot and Eva Maria Vecchi. 2015. Building a shared world: Mapping distributional to model-theoretic semantic spaces. In *Proceedings of EMNLP2015*, Lisbon, Portugal.

Felix Hill, Roi Reichart, and Anna Korhonen. 2015. Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*, 41(4):665–695.

Jerrold J Katz and Jerry A Fodor. 1963. The structure of a semantic theory. *Language*, 39(2):170–210.

Saul A Kripke. 1972. Naming and necessity. In *Semantics of natural language*, pages 253–355. Springer.

Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. 2017. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*, 123(1):32–73.

Germán Kruszewski and Marco Baroni. 2015. So similar and yet incompatible: Toward the automated identification of semantically compatible words. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 964–969.

Elizaveta Kuzmenko and Aurélie Herbelot. 2019. Distributional semantics in the real world: building word vector representations from a truth-theoretic model. In *Proceedings of the 13th International Conference on Computational Semantics (IWCS 2019)*. Association for Computational Linguistics.

Thomas K Landauer and Susan T Dumais. 1997. A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychological review*, pages 211–240.

Tal Linzen, Grzegorz Chrupała, and Afra Alishahi, editors. 2018. *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*. Association for Computational Linguistics, Brussels, Belgium.

Tal Linzen, Grzegorz Chrupała, Yonatan Belinkov, and Dieuwke Hupkes, editors. 2019. *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*. Association for Computational Linguistics, Florence, Italy.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Tomas Mikolov, Edouard Grave, Piotr Bojanowski, Christian Puhrsch, and Armand Joulin. 2018. Advances in pre-training distributed word representations. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*.

Barbara H Partee. 1979. Semantics – mathematics or psychology? In *Semantics from different points of view*, pages 1–14. Springer.

Barbara H Partee. 2014. The history of formal semantics: Changing notions of linguistic competence. https://udrive.oit.umass.edu/partee/Partee2014Harvard.pdf. 9th Annual Joshua and Verona Whatmough Lecture, Harvard.

Hilary Putnam. 1975. *Philosophical Papers: Mind, Language, and Reality*, volume 2. Cambridge University Press.

Matthew Saxton. 2000. Negative evidence and negative feedback: Immediate effects on the grammaticality of child speech. *First Language*, 20(60):221–252.

Eva M Vecchi, Marco Marelli, Roberto Zamparelli, and Marco Baroni. 2017. Spicy adjectives and nominal donkeys: Capturing semantic deviance using compositionality in distributional spaces. *Cognitive science*, 41(1):102–136.

Su Wang, Greg Durrett, and Katrin Erk. 2018. Modeling semantic plausibility by injecting world knowledge. In *Proceedings of the 2018 Conference of the*

*North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 303–308.

Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. 2014. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2:67–78.