Continuum models of semantics for language discovery

Deepali Semwal, Sunakshi Gupta and Amitabha Mukerjee
Department of Computer Science and Engineering
Indian Institute of Technology, Kanpur
{deepalis,sunakshi,amit}@cse.iitk.ac.in

Abstract

What if we had to work in language with a semantic model that was shared with robotics and vision? We consider the question of bootstrapping NLP based on an unified continuum semantics for actions. We consider a scenario with contrastive concepts – objects (*ball, square*), actions (*throw, roll*), colours, and agents. We first acquire the semantics, then map these to crowd-sourced adult Hindi commentaries without any parse or POS knowledge. Despite wide variations across commentaries, a small subset of high-confidence labels is acquired with a simple contrastive association measure. This seed knowledge is used to iteratively bootstrap larger syntactic patterns, starting with the noun phrase and going on to the NP VP complex. Using a syllabic model of the input, we also discover morphological structure and agreement ("chaukor fenk-A", "ball fenk-I"). Since the approach is agnostic to language, we can work just as well with narratives in another language; results are shown for English. With this work, we are also releasing the action videos and the Hindi / English corpora, part of the planned multi-lingual *Videobabel* corpus.

1 Introduction

Do we use different models when we throw a ball, see a ball thrown, or talk about throwing a ball? Today each of these fields - robotics, vision, and NLP, use machine learning approaches with isolated training sets, separate models and differing paradigms (robotics and vision are continuous, NLP semantics is often discrete). As NLP gets integrated into the whole agent, one should try to consider if it may be more efficient if we could share some of the semantic structure across such related situations. This idea is also motivated by the discovery of motor neurons which seems to relate

Secondly, suppose we have a unified model such as this, would it be useful for learning language? Clearly infants have some conceptual priors before they come to language, and there is good reason to believe that these are not modeled as crisp, boolean predicates. Can we use such a model to learn words, morphology, and syntax?

We discuss these two main contributions next.

Mapping concepts multi-modally

Consider an AI agent which is in the following situations:

(a) The agent recognizes [Sam throwing a ball to Jane]

(b) The agent understands the sentence "Sam threw the ball to Jane"

(c) The agent executes [throwing the ball to Jane].

Classical AI divides up the problem of intelligence into aspects that can be distinguished based on the input and output. Thus the task (a) above would typically be handled by computer vision. Researchers would take a large class of videos labelled as [throw], and would build a classifier function, f_{vis} that would distinguish it from other actions. For the language input in task (b), one would use a pre-trained parser and a semantic analyzer trained on a very large set of POS-, parse-,

D S Sharma, R Sangal and J D Pawar. Proc. of the 11th Intl. Conference on Natural Language Processing, pages 249–257,
Goa, India. December 2014.

and semantically- annotated sentences. Given these tools, it may be able to map the input to a formal structure such as `throw(agent:Sam, object: Ball, path: $p)^goal($p, Sita)`. Let us call designate this mapping process as f_{nlp}. For task (c) a robotic agent would operate in a space where it can evaluate the result of a throwing action. It would do many trials of throwing along different 3D paths (or at goals), and come up with a function f_{rob} that can throw the ball effectively along any target path.

Now we observe that these three functions are completely disjoint, and there can be no synergy in terms of correlating knowledge between these modalities. For example, consider an expert throwing robot - after lots of training, it has a very fine f_{rob}, and can throw balls very well. However, its f_{vis} is just about average. Now, when the agent is looking at Sam's throw, it cannot use it's own f_{rob} to determine if the throw is good or bad, or make suggestions for Sam to improve his throw; for that, it would need to be trained on the image data all over again.

This separation is even more critical when it comes to language. Given a language statement it cannot relate it to the seen or executed throw. The area in NLP called grounded language learning attempts to provide mechanisms for learning the grounded meaning of a symbol, but these are limited to linking up a recognizing function f_{vis} as the semantic map of the word "throw". (Kwiatkowski et al., 2011) But the actual throw depends on what is being thrown, and who is throwing it, and how - so the actual f_{vis} will either fail for other objects, or it will have to have a potentially infinite set of arguments to handle all sorts of contextual variations. That is, it runs into the frame problem.

Now consider these two sentences:

(a) Sam threw a glance at Sita.

(b) Sam threw the flashlight beam into the cave.

Indeed, primate brains seem to be operating in a more integrated manner. Motor behaviour invokes visual simulation to enable rapid detection of anomalies, while visual recognition invokes motor responses to check if they match. Linguistic meaning activates this wide range of modalities (Binder and Desai, 2011). Such a cross-modal model also permits affordance and intentionality judgments, which our system can also achieve.

The first part of this work (section 2) is inspired by the observation that each time the robot throws the ball for motor learning, it also generates a visual trajectory from which it can learn a visual model of the action. Further, the motion parameters are correlated with the visual feedback - both lie on matched low-dimensional curved manifolds which can be aligned.

We develop a unified approach for modeling actions, based on visual inputs (say trajectories or paths) resulting from given motor commands. The model constitutes a low-dimensional manifold that discovers the correlations between visual and motor inputs. The model can be applied to either visual recognition or to motor tasks.

Bootstrapping a lexicon and syntax

For the language task (section 3), we consider the system which already has a rudimentary model for [throw], being exposed to a set of narratives while observing different instances of throwing. The narratives are crowdsourced from speakers for a set of synthetic videos which have different actions (throw or roll), agents ("dome" or "daisy" - male/female), thrown object or trajector (ball or square), colour of trajector (red or blue) and path (on target, or short, or long). We work with transcribed text, and not with direct speech, so we are assuming that our agent is able to isolate words in the input.

An important aspect of working with crowdsourced data is that for the very beginning learner, we need a more coherent input where similar phrases are used for similar situations. This is difficult, given the diversity of our crowdsourced input, so we first identify a small coherent subset (called the *family lect*) on which initial learning is done. This is then extended to the remaining narratives (the *multi-lect*).

The system works on subsets of the narratives for each semantically distinct category. The joint word-semantics probabilities are computed. To learn the label for [ball], we

contrast the frequency of a word being uttered when a [ball] is thrown or rolled, vs a [square]. Candidate labels are ranked based on the ratio of these joint probabilities - p(word,concept) / p(word,non-concept). The high-confidence matches - those significantly higher than the next match (abt 20%) are taken as the initial bootstrapping map (fig. 4). This partial lexicon is then used to learn a partial syntax, which is ploughed back to learn more lexemes (and also synonyms and alternations). When this interleaving stabilizes, we broaden the semantic context to learn other structures. Finally, we find that we are able to discover a good chunk of transitive verb syntax. The system is demonstrated on Hindi, where we are also able to discover some morphological agreement relations. Since we use no knowledge of langauge, the same approach also works for English.

This partial-analysis based approach is substantially different from other attempts at grounded modeling in NLP, which have focused on demonstrating the acquisition of syntax /morphosyntax (Madden et al., 2010), (Kwiatkowski et al., 2011), (Nayak and Mukerjee, 2012). One may call this approach *dynamic NLP*, since it keeps learning from every sentence, and does not generate a static model. Also, after an initial bootstrapping phase driven by this multi-modal corpus, learning can continue to be informed by text alone, a process well-known from the rapid vocabulary growth after the first phase of language acquisition in children (Bloom, 2000).

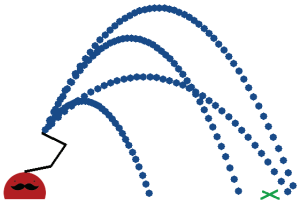

Figure 1: Set of trajectory images. Each trajectory is associated with the motor parameters of the throw.

2 Visuo-motor Pattern Discovery

Learning a few visuo-motor tasks are among our agent's very first achievements. Let us consider the act of throwing a ball. Our learner knows the motor parameters of the throw as it is being thrown - here we focus not on the sequence of motor torques, but just the angle and velocity at the point of release.

Each trajectory gives us an image (samples - fig. 1). We are given a large set of images (say, N=1080), each with 100×100 pixels. Each image can be thought of as a point in a 10^4-dimensional space. The set of possible images is enormous, but we note that if we assign pixels randomly, the probability that the resulting image will be a [throw] trajectory is practically zero. Thus, the subspace of [throw] images is very small.

Next we would like to ask what types of changes can we make to an image while keeping it within this subspace? In fact, since each throw varies only on the parameters (θ, v), there are only two ways in which we can modify the images while remaining locally within the subspace of throw images. This is the dimensionality of the local tangent space at any point, and by stitching up these tangent spaces we can model the entire subspace as a non-linear manifold of the same intrinsic dimensionality. The structure of this image manifold exactly mimics the structure of the motor parameters (the motor manifold). They can be mapped to a single joint manifold, which can be discovered using standard non-linear dimensionality reduction algorithms such as ISOMAP. In fig. 2, we show the resulting manifold obtained using a hausdorff distance metric (Huttenlocher et al., 1993): $(h(A,B) = max_{a \in A} min_{b \in B} \|a - b\|)$.

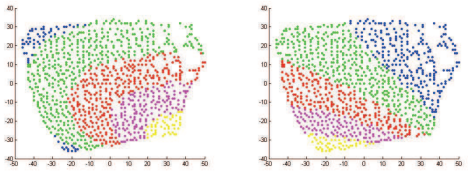

Figure 2: Variations in the manifold according to a) angle of projection, and b) velocity. (low values in yellow)

The same idea can be used to find correlations in any system involving a motion or a change of initial conditions, using this algorithm:

Algorithm 1 Visuo-motor law discovery Algorithm

No. of images	100	200	400	600	800	1000
SAE in velocity	3.69	3.57	3.85	2.71	2.38	1.93

Table 1: Sum Absolute Error (Velocity) falls as N increases

1: **Input:** Set of high dimensional images $\{I_1, I_2, I_3, ...I_N\}$, and corresponding control parameters.
2: **Step1:** Obtain a low dimensional embedding for the images using ISOMAP.
3: **Step2:** Train a regression model to acquire the mapping from the low dimensional (curved) coordinates to the control parameters.
4: **Step3:** For executing a new throw, use a (query) image with desired path. Find a linear interpolation J for this query image: $J = \sum_{j=1}^{k} w_j I_j$
5: **Step4:** Calculate the embedding points for the query image using the weights learnt in Step3. $\hat{Q} = \sum_{j=1}^{k} w_j \hat{q}_j$
6: **Step5:** Use the mapping learnt in Step2 to obtain the corresponding parameters for the query image J_i.

end

This is a generic algorithm for discovering patterns in visuo-motor activity. Initially, its estimate of how to achieve a throw are very bad, but they improve with experience. We model this process in table 1 - as the number of inputs N increases, the error in predicting a throw decreases - at N=100, the error is nearly twice that at N=1000.

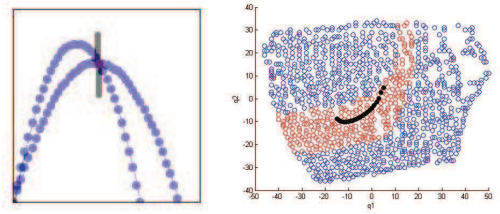

Figure 3: *Throwing darts.* a) Two trajectores that hit the board near the middle. b) 2D manifold of projectile images, showing band for "good throws", and interpolated curve for "best" throws.

The visuo-motor manifold model developed here is not task-specific, but can be applied for many tasks involving projectile motion, catching a ball, throwing at a basket, darts, tennis, etc. As an example we can consider darts - ignoring the lateral deviations, the successful trajectories are those that intersect the dartboard near its center (fig. 3a). This corresponds to a "good zone" in the latent space (central band in fig. 3b) Of these, one may wish to select those that are nearly orthogonal to the board at contact (short curved axis). Of course, actual task performance will improve with experience (more data points in the vicinity of the goal), resulting in better performance. This model only provides a starting point.

3 Bootstrapping Language

In order to learn language, we create a set of 15 situations involving the actions [throw] and [roll] (one of the agents is shown in fig. 1). The videos of these actions are then put up for commentary. Largely students on the campus network contributed; we obtain 18 transcribed narratives in Hindi. Later we also collected 11 English commentaries.

The compiled commentaries vary widely in lexical and constructional choices. An example description for of a video is "daisy throws a blue square". Another subject describes the same video as "now daisy threw the blue box which fell right on the mark" . Both these narratives varies in terms of lexical units used as well as the details incorporated in description.

As described earlier, we first select a more coherent subset of narratives - those that have a more consistent vocabulary from - and identify these as the *family lect.*

Given the [throw] model, the system can identify the act of throwing, and also the agent who throws, the object thrown, and its path. Further, we assume similar capabilities (not implemented) for [roll], and also the ability to discriminate a square from a circle, red from blue, and the two agents - [Dome] with a moustache (fig. 1, male), and [Daisy] with a ponytail. Based on these distinctions, it tries to find words that differ in their usage between the two contrasting situations. (Note: we use square brackets to indicate a concept, the semantic pole of a linguistic symbol) This contrastive approach has been suggested as a possible strategy applied by child learners (Markman, 1990). Given two contrasting concepts c_1, c_2, we compute the empirical joint probabilities of word (or later, n-gram) ω_i and concept c_1, c_2, and compute their contrast score:

$$S_{\omega_i, c_1} = \frac{P(\omega_i, c_1)}{P(\omega_i, c_2)}$$

We also compare the corpus of narratives

Schema	Top Hindi Lexemes	Score	Freq	ratio
[circle]	बॉल (ball) [ball]	19.9	19	1.43
	गयी (gayi) [went]	13.9	13	
[square]	चौकोर (chaukor) [square]	25.1	24	5.00
	गिरा (giraa) [fell]	5.01	9	
[daisy]	डेज़ी (daisy)	27.5	29	10
	उछाली (uchaali) [toss]	2.75	3	
[dome]	डोम (dome)	13.5	36	1.76
	पहले (pehley) [before]	7.67	6	
[throw]	पहले (pehley) [before]	6.95	6	1.27
	फेंकी (phenki) [throw]	5.48	10	
[roll]	सरकाया (sarkaayaa) [roll]	16.2	15	2.01
	सरकायी (saraayi) [roll]	8.05	7	
[red]	लाल (laal) [red]	10.9	31	3.53
	रुकी (ruki) [stopped]	3.08	2	
[blue]	नीली (neeli) [blue]	14.6	14	1.37
	नीले (neeley) [blue]	10.7	10	

Figure 4: High confidence lexeme discovery : Contrastive scores and dominance ratio for top lexemes in. Highlighted squares show high-confidence lexemes (ratio more than twice)

here with that from a larger unannotated corpus CIIL/IITB Corpus for Hindi. We look for words that are more frequent in the input domain than in a general situation. This rules out many frequent words like है,के *(hai,ke)* [is, of]. The small set of high confidence words - whose contrastive probability is more than twice the next best match - are highlighted in Fig. 4.

3.1 Interleaving of Word / Syntax learning

Once the system has a few grounded lexemes, we proceed to discovering syntactic constructions. At the start, we try to learn the structure of small contiguous elements. One assumption we use here is that concepts that are very tightly bound (e.g. object and its colour) are also likely to appear in close proximity in the text (Markman, 1990). Another assumption (often called *syntactic bootstrapping*), is used for mapping new phrases and creating

लाल चौकोर (laal chaukor) [red square]
लाल रंग का चौकोर
(laal rang ka chaukor) [red coloured-GEN square]

Table 2: Initial constructions learned from the trajector delimited strings

लाल बॉल (laal ball) [red ball]
नीले रंग का चौकोर
(blue coloured-GEN square) [blue square]

Table 3: Syntax for trajector - iteration 1, Family-lec

equivalence classes or contextual synonyms. This says that given a syntactic pattern, if phrase $p1$ appears in place of a known phrase $p0$, and if this substitution is otherwise improbable (e.g. the phrase is quite long), then $p0, p1$ are synonyms (if in the same semantic class) or they are in the same syntactic lexical category.

We find that one type of trajector (e.g. "ball") and its colour attribute (e.g. "lAl", [red]) have been recognized. So the agent pays more attention to situations where these words appear. Computationally this is modelled by trying to find patterns among the strings starting and ending with (delimited by) one of these high-confidence labels (e.g. "red coloured ball"). This delimited corpus consistes of strings related to a known trajector-attribute complex. Within these tight fragments, we show that standard grammar induction procedures are able to discover preliminary word-order patterns which can be used to induce broader regularities. We compare two available unsupervised grammar induction systems - Fast unsupervised incremental parsing (Seginer, 2007) and ADIOS(Solan et al., 2005); results shown here adopt the latter because of a more explicit handling of discovered lexical classes.

The initial patterns learned for the trajector in this manner are shown in Table 2. These are generalized using the phrase substitution process to yield the new lexeme बॉल (ball), [ball] (Table. 3). This is done based on the family-lect (Figure 4), and the filtered sub-corpus is used to learn patterns and equivalence classes.

The system now knows patterns for [red square], say, and it now pays attention to situations where the pattern is almost present,

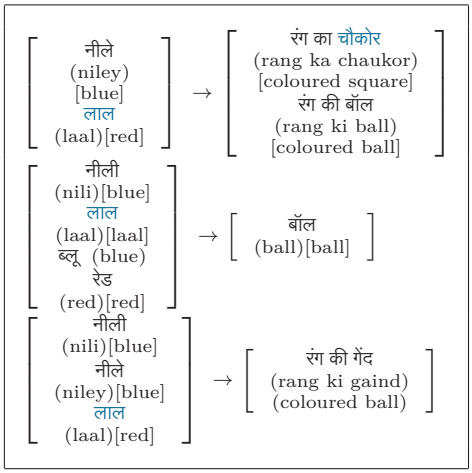

Table 4: Learned constructions pertaining to trajectors : The coloured units are the initially grounded lexemes

except for a single substitution. It can look into other semantic classes as well, (e.g. [blue square], [red ball]). In most of these instances (Fig. 4) we already have partial evidence for these units from their contrastive scores. Now if we discover new substitution phrases $p1$ in the position of $p0$, referring to a concept in the same semantic class (e.g. [ball] for [square]), and if $p1$ is already partially acceptable for [ball] based on contrastive probability, then $p1$ becomes an acceptable label for this semantic concept. This process iterates - new lexemes are used to induce new patterns, and then further new lexemes, until the patterns stabilize. This is then extended to the entire corpus beyond the small family lect; results are shown in Table 4.

The table captures a reasonable diversity of Noun Phrase patterns describing coloured objects. Note that words like "red" and "ball" have also become conventionalized in Hindi. We also observe that the token *niley* appears in the 4-word pattern *niley rang kaa chaukor* and is highly confident even from a single occurrence; this reflects the *fast mapping* process observed in child language acquisition after the initial grounding phase (Bloom, 2000). As the iteration progresses, these patterns are used for further enhancing the learners inventory of partial grammar.

3.2 Verb phrases and sentence syntax

Having learned the syntax for a trajector, this part of the input is now known with some con-

SCHEMA	Top scoring Hindi lexical units / cluster	score	ratio
[throw]	पहले (pehley) [before]	6.89	1.49
	गिरा, गिर, गिरी (gira, gir, giri) [fall]	4.6	
[roll]	सरकायी, सरकाया (sarkaaya, sarkaayi) [roll]	23.66	7.78
	वहीं (wahin) [there]	3.04	

Figure 5: *Verb learning.* Recomputed contrast scores after morphological clustering. Note that "threw" now appears as a high-confidence label.

[AGT] ने [TRJ] → [सरकाया (sarkAyA) / सरकायी (sarkAyI)]
[AGT] -NOM rolls [TRJ]

Table 5: Initially acquired sentence constructions

fidence, and the learner can venture out to relate the agent to the action and path. In this study, we failed to find any high-confidence lexemes related to path, hence we were not able to bootstrap that aspect. In the following we restrict ourselves to patterns for the semantic classes [agent], [action], [trajector].

At this stage, the agent notes that many of the words seem rather similar (e.g. "sarkAyA", "sarkAyI" (H); or "throwing", "thrown" (E)). A text-based morphological similarity analysis reveals several clusters with alterations at the end of words (Fig. 5). To quantify this aspect, we consider normalised Levenshtein distance and perform a morphological similarity analysis. Since our input is text, we limit ourselves to analysis based on the alphabetic patterns as opposed to phonemic maps. Similar words are clustered using a normalized similarity index .Thus we have twelve type instances of **(◌ा -aa) - (◌ी ii)**, and seven types for *(◌ा -aa) - (◌े -e)*. We find that these variants - e.g. "sarkaayaa", "sarkaayii" - appear in the same syntactic and semantic context. These clusters are now used to further strengthen the lexeme and action association.

[[AGT] ने [TRJ] ([AGT] ne [TRJ])] → [फेंका (phenka) फेंकी (phenki)]

[AGT] -NOM throws [TRJ]

Table 6: Sentence syntax discovered - iteration 1 - FL

[[AGT] ने [TRJ] ([AGT] ne [TRJ])] → [फेंका (phenka) फेंकी (phenki) [throws] सरकायी (sarkaayi) सरकाया (sarkaaya) [rolls]]

[AGT] ने नीला चौकोर फेंका
([AGT] ne nila chaukor phenka)
[[AGT] [throws a blue square]

Table 7: Learned constructions over trajector phrases, The coloured units are the initially grounded lexemes

Again, we use an iterative process, starting with grounded unigrams, moving a level up to learn simple word-order patterns, learning alternations and lexical classes through phrase substitution, and so on to acquire a richer lexicon and syntax. The learner has the concept of agent and has associated the words "daisy" and "dome". One action word is known ("sarkaaya", roll) while the word for "throw" is not discovered due to lexical variations. (Fig. 5). We now filter the corpus with these known words and try to discover the verb phrase syntax. Here the known trajector syntax (table 4) is considered as a unit (denoted as **[TRJ]**, and the concept of agent ([daisy] or [dome]) is denoted **[AGT]**. Results of initial patterns, obtained based on the known action lexeme, are shown in Table 5.

Next, we interleave this syntactic discovery with lexical discovery, permitting also bigram substitutions. This gives us the more general results of Table 6. Again the system iterates over the corpus till the discovery of new patterns converges (Table 7). An interesting observation is that the Hindi data finds a phrase in the TRJ position - नीला चौकोर (nilaa chaukor) [blue square]. This had not been learned in the trajector iteration, since *nilaa* was less frequent.

Objects	Abstract	Fauna
बॉल, गेंद	दृष्टि	नाग
नाखूनों (naakhunon) *[nails]*	(drishti)	(naag)
बम *[bomb]*	*[glance]*	*[Cobra]*
रस्सा (rassa) *[rope]*		

Table 8: Trajector classes for Hindi

Thus we see that with this approach we are able to acquire several significant patterns. These patterns apply to only a single action input, and for a very limited set of other participants. But it would be reasonable to say that the agent may observe similar structures elsewhere - e.g. in a context involving hitting, say, if we have the sentence "Daisy hit Dome" then the agent may use the syntax of [AGT] [verb] [TRJ] to extend to this context and guess that "hit" may be a verb and "Dome" the object of this action (which it knows from the semantics). Thus, once a few patterns are known, it becomes easier to learn more and more patterns, which is the fast mapping stage we have commented upon earlier.

4 Expanding the selection set for the verbs classes

In the grounded phase, we discovered that objects like [ball] can be thrown or pushed. Thus, the verbs "फेंकी (phenki)" would select for trajectors such as "ball" or गेंद gaind.

As the learner matures, she acquires a richer ontology of actions and objects, and is of course exposed to large amounts of language, mostly without direct grounding. In the next phase, we consider how this process enables an expansion of the selection set for these verbs. For this purpose, we consider the already familiar syntactic patterns. We use Hindi wordNet as our knowledge base and analyse the new situations with already familiar verbs and syntaxes. We here consider the objects that our known verbs take as arguments in a bigger Hindi CIIL/IITB corpus.

A total of 29 sentences for Hindi are extracted by filtering for the verb forms learned for the actions in the grounded phase ("फेंकी" (phenki) etc .) While the syntax patterns for these new sentences are much more complex, we expect the trajector term to appear as the noun that is closest to the verb; based on the syntax learned we look at nouns before the verb for Hindi language.

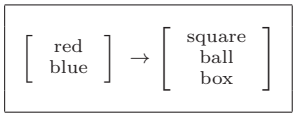

Table 9: Learned trajector constructions

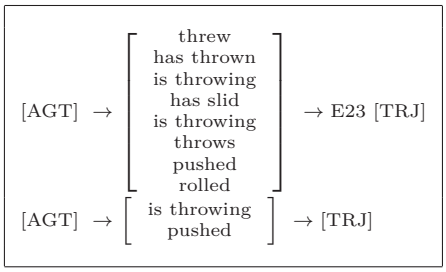

Table 10: Learned constructions over trajector phrases. E23 is the equivalence class learned for *a, the*

We discover sentences such as "स्टोव वाले बाबू ने मोतीभाई के चेहरे पर एक सहानुभूतिपूर्ण दृष्टि फेंकी" *[the stove man threw a sympathetic glance at Motibhai's face]*, where the token दृष्टि (drishti) *[glance]* appears in the place of [trajector]. Here it is as the object which belongs to *perception class* and hence an abstraction (see Table 8).

5 Acquiring another language: English

Here we collected 11 commentaries, which also vary widely. Again, starting with a "family" corpus, we obtain a small set of high-confidence labels (Figure 6, 7). At the bigram discovery stage, *"threw"* and *"rolled"* are found to be substitutable by *has thrown, is throwing*; and *has slid* respectively. Interestingly in expanding the corpus with wordnet knowledge, we find that words such as "glance", "flashlight" etc. also appear as throw-able in English, paralleling the Hindi usage.

Note that our lexical categories differ widely from syntactic categories, since they are influenced considerably by semantics. It is quite possible that human language users also use such mixed categories. At the same time, several traditional structures (e.g. Adj-N (red ball, laal chaukor),Art-N (E23 ball),

Top English lexemes	score	Freq	ratio
ball	28.7	27	3.99
near	7.19	6	
square	17.5	17	3.59
box	4.87	4	
daisy	23.6	22	7.63
right	3.09	2	
dome	29.3	29	6.00
before	4.88	4	
threw	14.8	14	1.91
throwing	7.72	7	
rolled	8.12	7	1.12
slid	7.28	6	
red	8.10	23	3.95
slightly	2.05	1	
blue	5.67	22	1.43
which	3.95	7	

Figure 6: High Confidence units

Top scoring English lexical units / cluster	score	ratio
threw	14.86	2.36
Throwing, thrown	6.29	
rolled	8.07	1.11
slid	7.27	

Figure 7: High Confidence clusters

etc are also discovered (Table 9, 10). It also discovers agreement between the unit "chaukor", [`square, M`] and verbs ending in -aa. However, following the usage-based approach (Tomasello and Tomasello, 2009), we would be inclined to view these constructions as reducing the description length needed to code for the strings arising in this context.

Thus the system has learned that the set of objects (and words) selected by an action such as "throw" may be broader than the initial set. This process actually broadens the

semantics of throw itself, from the initial interpretation as a physical action, to something broader. This broader semantics is actually reflected in alternate word senses. One task which we do not attempt here is to discover this semantics as an extension of the original semantics in the physical sense; this would also be an important part of core NLP, but it is quite a challenging topic in itself.
This work however lies in the space of vision and action, and one may consider this process in terms of discovering similarities between different actions in videos, an area that is otherwise well-researched, but yet to reach this level of analysis (e.g. (Efros et al., 2003)).

6 Conclusion

The above analysis provides a proof-of-concept that a system starting with very few priors can, a) combine motor, visual (and possibly other modalities) into an integrated model, and b) use this rudimentary concept knowledge to bootstrap language. We also note that such a system can then learn further refinements to this concept space using language alone.

From here work needs to proceed in two directions. First would be to demonstrate scalability by including actions other than [throw]. One of our main claims is that neither the semantics nor the linguistic components had any kind of annotation, so the training data set needed for this should be relatively easy to generate compared to tree banks and semantically annotated data. The two linguistic corpora and the videos are being released as part of a multi-lingual, action centric corpus that we call *Videobabel*. It is plausible that with increasing availability of such unannotated multi-modal corpora, along with motor-enabled models of action, would permit the rapid scaling of conceptual and linguistic models.

References

Jeffrey R Binder and Rutvik H Desai. 2011. The neurobiology of semantic memory. *Trends in cognitive sciences*, 15(11):527–536.

Paul Bloom. 2000. *How Children Learn the Meaning of Words*. MIT Press.

Alexei A Efros, Alexander C Berg, Greg Mori, and Jitendra Malik. 2003. Recognizing action at a distance. In *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, pages 726–733. IEEE.

Daniel P. Huttenlocher, Gregory A. Klanderman, and William J Rucklidge. 1993. Comparing images using the hausdorff distance. *IEEE PAMI*, 15(9):850–863.

Tom Kwiatkowski, Luke Zettlemoyer, Sharon Goldwater, and Mark Steedman. 2011. Lexical generalization in ccg grammar induction for semantic parsing. In *Proc. EMNLP*, pages 1512–1523.

C. Madden, M. Hoen, and P.F. Dominey. 2010. A cognitive neuroscience perspective on embodied language for human-robot cooperation. *Brain and Language*, 112(3):180–188.

Ellen M Markman. 1990. Constraints children place on word meanings. *Cognitive Science*, 14(1):57–77.

Sushobhan Nayak and Amitabha Mukerjee. 2012. Grounded language acquisition: A minimal commitment approach. In *Proc. COLING 2012*, pages 2059–76.

Yoav Seginer. 2007. Fast unsupervised incremental parsing. In *ACL Ann. Meet*, volume 45, page 384.

Zach Solan, David Horn, Eytan Ruppin, and Shimon Edelman. 2005. Unsupervised learning of natural languages. *PNAS*, 102(33):11629–34.

Michael Tomasello and Michael Tomasello. 2009. *Constructing a language: A usage-based theory of language acquisition*. Harvard University Press.