On Distributional Semantics

Mark Steedman

August 2016
Outline

I: Distributional Theories of Content: Collocation vs. Denotation

II: Entailment-based Paraphrase Cluster Semantics (Lewis and Steedman, 2013a, 2014)

III: Multilingual Entailment-based Semantics (Lewis and Steedman, 2013b)

IV: Querying FreeBase

V: Extending the Semantics
The Problem of Content

• We have (somewhat) robust wide coverage parsers that work on the scale of Bn of words. They can read the web (and build logical forms) thousands of times faster than we can ourselves.

• So why can’t we have them read the web for us, so that we can ask them questions like “What are recordings by Miles Davis without Fender Rhodes piano”, and get a more helpful answer than the following?
Miles Davis • Get Up With It (CD, Album) at Discogs
https://www.discogs.com/Miles-Davis:Get-Up-With-It/8293374
★★★★★ Rating: 5 / 5 votes
Find a Miles Davis - Get Up With It price or release date. KansasDrums - Billy Cobham/Electric Bass - Michael Henderson/Electric Piano [Fender Rhodes]...

What are some great Fender Rhodes jazz albums/groups (pre...
https://www.quora.com/What-are-some-great-Fender-Rhodes-jazz-album
Bill Evans did an album in 1970 titled From Left to Right Playing the Fender Rhodes Electric Piano and the Steinway Piano. Mostly referred to by its...

Fender Rhodes electric piano - Bill Evans
www.billevanswebpages.com/rohodespecific.html
That he changed the sound of jazz piano from the mid-fifties on is well documented... he sometimes used is the seventies, the Fender Rhodes electric piano. ... he recorded with the Rhodes more than five studio albums, and a "live" now...

Live-Evil (Miles Davis album) - Wikipedia, the free encyclopedia
https://en.wikipedia.org/wiki/Live-Evil_(Miles_Davis_album)
Live-Evil is an album of both live and studio recordings by American jazz musician Miles Davis. ... John McLaughlin electric guitar; Keith Jarrett Fender Rhodes electric piano; Fender Contemp electric bass; Michael Henderson electric bass... Title:

Who was the first artist to record using a Fender Rhodes? - T...
ept-forum.com » » Misc » Classic & Modern Fender Rhodes Artists
25 Mar 2006 - 30 posts, 10 authors
Too Many Ways of Answering The Question

• The central problem of QA is that there are too many ways of asking and answering questions, and we have no idea of the semantics that relates them.
• Your Question: Has Verizon bought Yahoo?
• The Text:
  1. Verizon purchased Yahoo. ("Yes")
  2. Verizon’s purchase of Yahoo ("Yes")
  3. Verizon owns Yahoo ("Yes")
  4. Verizon managed to buy Yahoo. ("Yes")
  5. Verizon acquired every company. ("Yes")
  6. Yahoo may be sold to Verizon. ("Maybe")
  7. Verizon will buy Yahoo or Yazoo. ("Maybe not")
  8. Verizon didn’t take over Yahoo. ("No")
The Problem

• The **hard problem in semantics** is not the logical operators, but the **content** that they apply over.

• How do we define a **theory of content** that is **robust** in the sense of generalizing across linguistic form, and **compositional** in the sense of:
  – being **compatible with logical operator semantics** and
  – supporting commonsense inference?
Previous Work

• Many have tried to build a form-independent semantics by hand:

  – both in linguistics, as in the “Generative Semantics” of the ’70s and the related conceptual representations of Schank and Langacker;
  – and in computational linguistics, as in WordNet, FrameNet, Generative Lexicon, VerbNet/PropBank, BabelNet, AMR . . .
  – and in knowledge graphs such as FreeBase.
Previous Work

Such hand-built semantic resources are extremely useful, but they are notoriously incomplete and language-specific.

- So why not let machine learning do the work instead?
- Treat semantic primitives as hidden.
- Mine them from unlabeled multilingual text, using Machine Reading.
One (Somewhat*) New Approach

• Clustering by Collocation
  – Meanings are vectors (etc.)
  – Composition is via Linear Algebraic Operations such as vector addition, matrix multiplication, Frobenius algebra, packed dependency trees, etc.
  – Vectors are good for underspecification and disambiguation (Analogy tasks and Jeopardy questions), and for building RNN embeddings-based “Supertagger” front-ends for CCG parsers, and related transition models for transition-based dependency parsers

* Cf. the MDS “Semantic Differential” (1957), which Wordnet was developed by George Miller partly in reaction to.
For Example: Analogy via Word2Vec

• king - man + woman = ["queen", 0.7118192911148071], ["monarch", 0.6189674139022577],
  ["princess", 0.5902431011199951], ["crown prince", 0.5499460697174072],
  ["prince", 0.5377321243286133]]

• wine - France + Scotland = ["malt whiskey", 0.5816349983215332],
  ["whiskey", 0.5434308648109436], ["single malts", 0.5399404168128967],
  ["malt whiskeys", 0.5365753173828125], ["whiskeys", 0.5350099802017212]]

• right - good + bad = ["wrong", 0.548572838306427], ["fielder Joe Borchard", 0.47464582324028015],
  ["left", 0.46392881870269775], ["fielder Jeromy Burnitz", 0.45308032631874084],
  ["fielder Lucas Duda", 0.4393044114112854]]

• Bernanke - USA + Russia = ["Ben Bernanke", 0.6536909937858582],
  ["Kudrin", 0.6301712989807129], ["Chairman Ben Bernanke", 0.6148115396499634],
  ["Medvedev", 0.6024096608161926], ["Putin", 0.5873086452484131]]
Orthogonality in Vector Components

• “A is to B as C is to D” works best when the two components AB and BC are orthogonal i.e. independent, and if B and D are close anyway. Compare:
  
  – smaller - small + big = [['bigger', 0.7836999297142029], ['larger', 0.5866796970367427], ['Bigger', 0.5707237720489502], ['biggest', 0.5240510106086731], ['splashier', 0.5107756853103638]]
  
  – unhappy - happy + fortunate = [['incensed', 0.49339964985847473], ['displeased', 0.4742095172405243], ['unfortunate', 0.46231183409690857], ['frustrated', 0.4529050886631012], ['miffed', 0.445096492767334]]
  
  – Las Meninas - Velasquez + Picasso = [['Paul Cézanne', 0.6370980739593506], ['Pablo Picasso', 0.634435772895813], ['Renoir', 0.6213735938072205], ['Dubuffet', 0.619714617729187], ['Degas', 0.6172788143157959]]
  
  – kill - dead + alive = [['destroy', 0.4605627655982971], ['exterminate', 0.42368459701538086], ['overpower', 0.4220679700374603], ['survive', 0.3986499309539795], ['stymie', 0.39753955602645874]]
Factorization in Vector Components

• Mitchell and Steedman (2015) show that the orthogonality effect holds for a range of morpho-syntactic components, and that in general the cosine of vector differences is a strong predictor of performance on the word analogy task for CBOW, SkipGram, and GloVe.

⚠️ But this makes them look rather like old fashioned morpho-syntactic-semantic features male/female, active/inactive, etc.

• It is unclear how to apply logical operators like negation to vectors.

• Beltagy et al. (2013) use vectors to estimate similarity between formulæ in an otherwise standard logical approach.
Another (Somewhat) New Approach

• Clustering by Denotation:
  – Meanings are automatically-extracted hidden relations, identified by automatic parsing and recognition of Named Entities either in text or in knowledge graphs.
  – Semantic composition is via syntactic derivation and traditional Logical Operators such as $\neg$, $\land$, $\lor$, etc.
  – Denotations are good for inference of entailment from the text to an answer to your question.
  – They are directly compatible with negation, quantifiers, modality, etc.
II: Entailment-based Paraphrase Cluster Semantics

• Instead of traditional lexical entries like the following:

\[
\text{author} := N/PP[of] : \lambda x\lambda y. author'xy \\
\text{write} := (S\backslash NP)/NP : \lambda x\lambda y. write'xy
\]

• —we seek a lexicon capturing entailment via logical forms defined as (conjunctions of) paraphrase clusters like the following:

\[
\text{author} := N/PP_{of} : \lambda x_{book}\lambda y_{person}.relation37'xy \\
\text{write} := (S\backslash NP)/NP : \lambda x_{book}\lambda y_{person}.relation37'xy
\]

• Such a “distributional” lexicon for content words works exactly like the naive lexicon (1) with respect to the semantics of quantification and negation.
Finding Typed Relation Expressions in Text

- We obtain the clusters by parsing (e.g.) Gigaword text with (e.g.) the CCG-based logical-form-building C&C parser, (Bos et al., 2004), using the semantics from Steedman 2012, with a lexicon of the first type (1), to identify expressions relating Named Entities such as Verizon, Yahoo, Scott, Waverley, etc.

- Nominal compounds for the same MUC named entity type are merged.

- Entities are soft-clustered into types according to a suitable method (Topic models, WordNet clusters, FreeBase types, etc.)

- These types are used to distinguish homonyms like the two versions of the born in relation relating PERSONS to DATES versus LOCATIONS.
Example

- Obama was born in Hawai’i.

(3) born := \((S\backslash NP)/PP[\text{in}] : \lambda x \lambda y. \begin{cases} x = \text{LOC} \land y = \text{PER} \Rightarrow \text{rel}49 \\ x = \text{DAT} \land y = \text{PER} \Rightarrow \text{rel}53 \end{cases} \) \ xy

\[
\begin{align*}
\text{Obama} & := \begin{cases} \text{PER} = 0.9 \\ \text{LOC} = 0.1 \end{cases} \\
\text{Hawai’i} & := \begin{cases} \text{LOC} = 0.7 \\ \text{DAT} = 0.1 \end{cases}
\end{align*}
\]

- The “Packed” Distributional Logical Form

(4) \( S : \left\{ \begin{array}{ll}
\text{rel}49 = 0.63 \\
\text{rel}53 = 0.27 \\
\text{hawaii’obama’}
\end{array} \right\} \)
**Directional Entailments**

- We now search for potential entailments between such typed relations, where for multiple pairs of entities of type $X$ and $Y$, if we find relation $A$ in the text we often also find relation $B$ stated as well.

Entailment is a directed relation: $X_{person}$ elected to $Y_{office}$ does entail $X_{person}$ ran for $Y_{office}$ but not vice versa.

- Thus we use an assymmetric similarity measure rather than Cosine.

- Lewis (2015); Lewis and Steedman (2014) apply the entailment graphs of Berant et al. (2012) to generate more articulated entailment structures.
Local Entailment Probabilities

- The typed named-entity technique is applied to (errorfully) estimate local probabilities of entailments:
  a. \( p(\text{conquer } x_{\text{country}} y_{\text{country}} \Rightarrow \text{invade } x_{\text{country}} y_{\text{country}}) = 0.9 \)
  b. \( p(\text{invade } x_{\text{country}} y_{\text{country}} \Rightarrow \text{attack } x_{\text{country}} y_{\text{country}}) = 0.8 \)
  c. \( p(\text{invasion (of } x_{\text{country}}) (\text{by } y_{\text{country}}) \Rightarrow \text{attack } x_{\text{country}} y_{\text{country}}) = 0.8 \)
  d. \( p(\text{invade } x_{\text{country}} y_{\text{country}} \Rightarrow \text{invasion (of } x_{\text{country}}) (\text{by } y_{\text{country}})) = 0.7 \)
  e. \( p(\text{invasion (of } x_{\text{country}}) (\text{by } y_{\text{country}}) \Rightarrow \text{invade } x_{\text{country}} y_{\text{country}}) = 0.7 \)
  f. \( p(\text{conquer } x_{\text{country}} y_{\text{country}} \Rightarrow \text{attack } x_{\text{country}} y_{\text{country}}) = 0.4 \)
  g. \( p(\text{conquer } x_{\text{country}} y_{\text{country}} \Rightarrow \text{conqueror (of } x_{\text{country}}) y_{\text{country}}) = 0.7 \)
  h. \( p(\text{conqueror (of } x_{\text{country}}) y_{\text{country}} \Rightarrow \text{conquer } x_{\text{country}} y_{\text{country}}) = 0.7 \)
  i. \( p(\text{bomb } x_{\text{country}} y_{\text{country}} \Rightarrow \text{attack } x_{\text{country}} y_{\text{country}}) = 0.7 \)
    (etc.)
Global Entailments

- The local entailment probabilities are then used to construct an entailment graph using integer linear programming with a prior $p = 0.25$ with the global constraint that the graph must be closed under transitivity.

- Thus, (f) will be included despite low observed frequency, while other low frequency spurious local entailments will be excluded.

- Cliques within the entailment graphs are collapsed to a single paraphrase cluster relation identifier.

- The entailment graph is Boolean, rather than probabilistic.
A simple entailment graph for relations between countries.
Lexicon

- The lexicon obtained from the entailment graph

  \[\text{attack} := (S\setminus NP) / NP : \lambda x \lambda y \lambda e. \text{rel}_1 x y e\]
  \[\text{bomb} := (S\setminus NP) / NP : \lambda x \lambda y \lambda e. \text{rel}_1 x y e \land \text{rel}_4 x y e\]
  \[\text{invade} := (S\setminus NP) / NP : \lambda x \lambda y \lambda e. \text{rel}_1 x y e \land \text{rel}_2 x y e\]
  \[\text{conquer} := (S\setminus NP) / NP : \lambda x \lambda y \lambda e. \text{rel}_1 x y e \land \text{rel}_2 x y e \land \text{rel}_3 x y e\]
  \[\text{conqueror} := VP_{pred}/PP_{of} : \lambda x \lambda p \lambda y \lambda e. p y \land \text{rel}_1 x y e \land \text{rel}_2 x y e \land \text{rel}_3 x y e\]

- These logical forms support correct inference under negation, such as that \textit{conquered} entails \textit{attacked} and \textit{didn’t attack} entails \textit{didn’t conquer}
Entailment

• Thus, to answer a question “Did X conquer Y” we look for sentences which subsume the conjunctive logical form $rel_2 \land rel_1$, or satisfy its negation $\neg rel_2 \lor \neg rel_1$.

 располагаем, что $invasion-of$ is a paraphrase of $invade = rel_2$, we also know $invasion-of$ entails $attack = rel_1$. 

Examples from Question-Answer Test Set

- Examples:

<table>
<thead>
<tr>
<th>Question</th>
<th>Answer</th>
<th>From Unseen Sentence:</th>
</tr>
</thead>
<tbody>
<tr>
<td>What did Delta merge with?</td>
<td>Northwest</td>
<td>The 747 freighters came with Delta’s acquisition of Northwest</td>
</tr>
<tr>
<td>What spoke with Hu Jintao?</td>
<td>Obama</td>
<td>Obama conveyed his respect for the Dalai Lama to China’s president Hu Jintao during their first meeting</td>
</tr>
<tr>
<td>What arrived in Colorado?</td>
<td>Zazi</td>
<td>Zazi flew back to Colorado. . .</td>
</tr>
<tr>
<td>What ran for Congress?</td>
<td>Young</td>
<td>. . . Young was elected to Congress in 1972</td>
</tr>
</tbody>
</table>

- **Full results** in Lewis and Steedman (2013a) and Lewis (2015)
III: Multilingual Entailment Cluster Semantics

- If we can find entailments including paraphrases by observing local entailments between statements in English of relations over typed named entities, there is no reason we shouldn’t consider statements in other languages concerning named entities of the same types as nodes in the same entailment graph.

- Thus from French *Shakespeare est l’auteur de* Mesure pour mesure, and knowledge of how French named entities map to English, we should be able to work out that *être l’auteur de* is a member of the *write* cluster.

- We use cross-linguistic paraphrase clusters to re-rank Moses n-best lists to promote translations that preserve the cluster-based meaning representation from source to target.
Experiment: Reranking SMT Translations

- For a source (French) sentence that can be dependency-parsed to deliver a cluster-semantic logical form:

- We Moses-translate (to English) taking the 50-best list and parsing (with C&C) to produce cluster-semantic logical forms.

- If the logical form of the top ranked translation is different from that of the source, we choose whatever translation from the remainder of the n-best list has the logical form that most closely resembles the source cluster semantics.
Reranking SMT

- Example:
  
  **Source:** Le Princess Elizabeth arrive à Dunkerque le 3 août 1999
  
  **SMT 1-best:** The Princess Elizabeth is to manage to Dunkirk on 3 August 1999.

  **Reranked 1-best:** The Princess Elizabeth arrives at Dunkirk on 3 August 1999.

- Fluent bilingual human annotators are then asked to choose between the one-best Moses translation and the cluster-based alternative.

<table>
<thead>
<tr>
<th></th>
<th>Percentage of Translations preferred</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-best Moses</td>
<td>5%</td>
</tr>
<tr>
<td>Reranked best</td>
<td>39%</td>
</tr>
<tr>
<td>No preference</td>
<td>56%</td>
</tr>
</tbody>
</table>
Reranking SMT

- Many cases of “no preference” were where Moses and the preferred translation were similar strings differing only in attachment decisions visible only in the logical form.

- No parallel text was used in these experiments.

- This is good, because SMT has already used up all of the available parallel text (Och, 2007)!

- Full results in Lewis and Steedman (2013b).
IV: Querying Freebase without QA pairs

• Reddy et al. (2014):
  – Rather than inducing a semantic parser from Freebase questions and answers. . .
  – Take a parser that already builds logical forms and learn the relation between those logical forms and the relations in the already denotational knowledge graph.

• Specifically:
  – First automatically map the linguistic logical forms into graphs of the same type as the knowledge graph;
  – Then learn the mapping between the elements of the semantic and knowledge-base graphs.
The Knowledge Graph

- Freebase is what used to be called a Semantic Net

- Cliques represent facts.

- Clique q represents the fact that Obama’s nationality is American

- Clique m represents the fact that Obama did his BA at Columbia
Cameron directed *Titanic* in 1997.

The logical form language is preprocessed to be homomorphic to the freebase language.
Mapping from logical forms to FreeBase paths

- Cameron directed *Titanic* in 1997.

\[
\begin{align*}
\text{directed.arg1}(E, \text{cameron}) & \quad \text{film.directed-by.arg2}(m, \text{yperson}) \\
\land \text{directed.arg2}(F, \text{titanic}) & \quad \land \text{film.directed-by.arg1}(m, x_{\text{film}}) \\
\land \text{directed.in}(G, 1997) & \quad \land \text{film.initial-release-date.arg1}(n, x_{\text{film}}) \\
& \quad \land \text{film.initial-release-date.arg2}(n, w_{\text{year}})
\end{align*}
\]
The Nature of the Mapping

- In the parser logical form, we need to replace
  - Entity variables with Freebase entities (e.g. *Cameron* with CAMERON)
  - Edge labels with Freebase relations (e.g. *directed.arg1* with *film.directed_by.arg2*)
  - Event variables with factual variables (e.g. *E* becomes *m* and *F* becomes *n*)

 אבל יש *O(k+1)^n* גרנדור גרפים אפשריים לכל הפורמט הלוגי (כולל קולחלות קשתות)

A Better Way to Query Knowledge Graphs

- Treat the paths relating named entities in the graph itself as logical forms of a rather stilted natural language, collecting local entailment probabilities, e.g:

  \[ p(directed_{\text{year}}x_{\text{film}}y_{\text{person}} \Rightarrow film.directed-by.arg2(m,y_{\text{person}}) \land film.directed-by.arg1(m,x_{\text{film}}) \land film.initial-release-date.arg1(n,x_{\text{film}}) \land film.initial-release-date.arg2(n,y_{\text{year}})) = 0.9 \]

- Then include such typed relations in the global entailment graph using ILP, where they will end up in the same paraphrase cluster as directed/3

- Finally, redefine the semantics of the original parser yet again, so that every occurrence of cluster identifiers like directed/3 in the entailment semantic conjunctions is replaced by the corresponding path expression.
A Better Way to Query Knowledge Graphs

- This can be done for any language for which the entailment semantics has been trained.

- The parsers can now be used to query the knowledge graph directly.

- Because they are entailment-supporting on the basis of exposure to a much wider range of text than just the knowledge graph, they will be able to give entailed answers that are not there in the graph, such as that Obama ran for office on the basis of facts that are, such as that he was elected.

- This also provides a way of dealing with the notorious incompleteness of the knowledge graph. (Lincoln's son Tad lacks a nationality in Freebase,)
An Even Better Way to Query Knowledge Graphs

- Building the knowledge graph directly from the parser, in the multilingual entailment cluster-based semantics, by machine reading.

- Harrington and Clark (2009); Harrington (2010) show that such Semantic Nets can be queried and updated efficiently at the scale we can now build them using the spreading activation of Collins and Loftus (1975).

⚠️ In the old days, the received wisdom was that SA didn’t work: either the decay rate was too high, so too few nodes were activated, or it was too low, so the whole network woke up.
• From Harrington and Clark (2009):

Figure 5. Average time to add a new node to the network vs. total number of nodes
Building Entailment-Semantic Nets Using SA

- The above proposal promises to solve two separate problems of classical semantic nets, namely: query form mismatch; and complexity of search and update.
V: Extending the Natural Semantics

A simple entailment graph for relations over events does not capture relations of causation and temporal sequence entailment.
Timestamped Data

• We have done pilot experiments with timestamped news, using the University of Washington NewsSpike corpus of 0.5M newswire articles (Zhang and Weld, 2013).

• In such data, we find that statements that so-and-so is visiting, is in and the perfect has arrived in such and such a place, occur in stories with the same datestamp, whereas is arriving, is on her way to, occur in preceding stories, while has left, is on her way back from, returned, etc. occur in later ones.

• This information provides a basis for inference that visiting entails being in, that the latter is the consequent state of arriving, and that arrival and departure coincide with the beginning and end of the progressive state of visiting.

• We can use it as the input to a neo-Reichenbachian semantics of temporality
Reconnecting with Logical Operator Semantics

- Some handbuilt lexical entries for auxiliary verbs (closed-class words):

  \[
  \text{has} := (S \setminus NP)/VP_{en} : \lambda p_E \lambda y. \text{consequent-state} \ (p_Ey) R \land R = \text{NOW}
  \]

  \[
  \text{will} := (S \setminus NP)/VP_{b} : \lambda p_E \lambda y. \text{priors} \Rightarrow \text{imminent-state} \ (p_Ey) R) \land R = \text{NOW}
  \]

  \[
  \text{is} := (S \setminus NP)/VP_{ing} : \lambda p_E \lambda y. \text{progressive-state} \ (p_Ey) R \land R = \text{NOW}
  \]

Reconnecting with Logical Operator Semantics

• Some potentially learnable lexical entries for implicative verbs:

\[
\text{tried} := (S\backslash NP)/VP_{to} : \lambda p_E \lambda y. rel_{try} p_E y R \land \text{rel}_{want} p_E y R \\
\land \text{preparatory-activity}(p_E y) y R \land R < \text{NOW}
\]

\[
\text{failed} := (S\backslash NP)/VP_{to} : \lambda p_E \lambda y. rel_{try} p_E y R \land \text{rel}_{want} p_E y R \\
\land \text{preparatory-activity}(p_E y) y R \land \neg p_E y R \land R < \text{NOW}
\]

\[
\text{managed} := (S\backslash NP)/VP_{to} : \lambda p_E \lambda y. rel_{try} p_E y R \land \text{rel}_{want} p_E y R \\
\land \text{preparatory-activity}(p_E y) y R \land p_E y R \land R < \text{NOW}
\]
Conclusion I: Denotation-based

- Learning over denotations, defined as relations over typed named entities, allows us to build entailment into lexical logical forms for content words via conjunctions of paraphrase clusters.

- The individual conjuncts are potentially language-independent.

- Mining them by machine reading remains a hard task, for which we have no more than proof-of-concept!

- The lexical conjunctions are projected onto sentential logical forms including traditional logical operators by the function words and CCG syntax.

- The sentential logical forms support fast inference of common-sense entailment.
Conclusion II: Collocation-based

- **Learning over Collocations**, represented as a vector space with reduced dimensionality, also represents meanings in terms of *hidden components*.

- Projection by vector addition remains *a hard baseline to beat!*

- By *superimposing a number of distinct collocations*, they remain the most powerful mechanism known for *resolving ambiguity*, as in the use of embeddings and LSTM in parser models.

- When Firth (1957/1968):179 made his oft-cited remark about knowing a word by the company it keeps, he was actually talking about *disambiguation*.
Thanks to:

- Johan Bos (Groningen), Steve Clark (Cambridge), James Curran (Sydney), Brian Harrington (Toronto), Julia Hockenmaier (Illinois), Mirella Lapata, Mike Lewis (Washington), Reggy Long (Stanford), Jeff Mitchell (UCL), Siva Reddy and Nathan Schneider (Georgetown).

- And to http://rare-technologies.com/word2vec-tutorial/#app for running Word2Vec, Congle Zhang and Dan Weld for NewsSpike, and to Google and ERC GramPlus for support.
Conclusions: For Philosophy of Language

• Under more traditional semantic theories employing eliminative definitions these entailments would have been thought of as belonging to the domain of inference rather than semantics, either as meaning postulates relating logical forms or as “encyclopædic” general knowledge.

• Carnap (1952) introduced meaning postulates in support of Inductive Logic, including a model of Probability, basically to keep the model small and consistent.

• Like Katz and Fodor (1963); Katz and Postal (1964); Katz (1971), we are in effect packing meaning postulates into the lexicon.

• This suggests that our semantic representation expresses an a pragmatic empiricist view of analytic meaning of the kind advocated by Quine (1951).
It can also be viewed as a grammar-based statistical model of “meaning as use” (Wittgenstein, 1953).
Conclusions: For Psychology

• Do children acquire the meaning of words like “annex” and “conquer” by building entailment graphs?
• I suggest they do, and that this is the mechanism for what Gleitman (1990) called syntactic bootstrapping of the lexicon—that is:
  – Once children have acquired core competence (by semantic bootstrapping of the kind modeled computationally by Kwiatkowski et al. 2012 and Abend et al., 2016), they can detect that “annex” is a transitive verb meaning some kind of attack without knowing exactly what it means.
  – They can then acquire the full meaning by piecemeal observation of its entailments and paraphrases in use.

This is a major mechanism of cultural inheritance of concepts that would otherwise in many cases take more than an individual lifetime to develop.
Conclusions: For Cognitive Science

- These terms compile into a (still) language-specific Language of Thought (Fodor 1975, passim), which is roughly what adult speakers do their thinking in.
- To the extent that the cliques or clusters in the graph are constructed from multilingual text, this meaning representation will approximate the hidden language-independent “private” Language of Mind which the child language learner accesses.
- However, very few terms in any adult logical form correspond directly to the hidden primitives of that Language of Mind. (red and maybe attack might be exceptions.)

🌟 Even those terms that are cognitively primitive (such as color terms) will not be unambiguously lexicalized in all languages.
Some conceptual primitives, such as that things can only be in one place at a time, probably predate human cognition, and are unlikely to be discoverable at all by machine reading of the kind advocated here.

- These properties are hard-wired into our minds by 600M years of vertebrate evolution.
- These are exactly the properties that Artificial Intelligence planning builds into the representation via the “Closed World Assumption” and the STRIPS dynamic logic of change.
- Computational Linguistics should learn from AI in defining a Linear Dynamic Logic for distributional clustered entailment semantics.
References


Lewis, Mike and Steedman, Mark, 2014. “Combining Formal and Distributional


