Introduction to Semantic Role Labelling

Behrang QasemiZadeh
zadeh@phil.hhu.de
Computational Linguistics Department, HHU – DRAFT

October 2018–January 2019

Language Resources for Semantic Role Labeling

Language resources for Semantic Role Labeling:

FrameNet

PropBank

VerbNet

. . .

Proposition Bank (PropBank)

The main intention behind developing PropBank (and its companion NomBank) was to provide an **annotated corpus**, which can be used for training supervised semantic role labeling systems.

In doing so, PropBank tries to develop and adopt a 'theory neutral approach' (we discuss its drawbacks).

It annotates arguments of verbs sense by sense using a set of coarse labels in the form of `Arg0`, `Arg1`, ... `Arg5`.

Additionally, adjuncts are annotated using a set of `ArgMs` (Argument Modifiers): LOCation, EXTent, ADVerbial, CAUse, ... (see the list at the end).

PropBank: Example

In PropBank, instances of the verb **build** are arranged into 4/5 sensel-like-categories, e.g. (note syntactic relations of `Args` to `rel`):

Sense 1: Mostly as to construct, manufacture

[ARG0 Honda 's plant in Marysville , Ohio ,] was gearing up to [rel build] [ARG1 1990 model Accords] , a Honda spokesman said.

[ARG0 You] [rel built] [ARG1 your career] on [ARG2 prejudice and hate]. **** This seems out of place**

[ARG1 double-deck freeways] [rel built] [ARGM-TMP today] with [ARG2 the heavily reinforced concrete and thicker columns required after the Sylmar quake]

PropBank: Example (contd.)

Sense 2: Mostly as in progressive development/grow

[ARG1 Pressures] began to [rel build].

[ARG0 The great silver clouds on the horizon] [rel build] [ARG1 themselves] [ARGM-LOC on the pale water].

[ARG1 The fiscal 1990 measure] [rel builds] on [ARG2 a pattern set earlier this year by House and Senate defense authorizing committees]

PropBank Corpus

As shown in the examples, `Args` are consistently annotated disregarding syntactic forms. For instance, `Arg1` (proto-patient) in:

[ARG1 Pressures] began to [rel build].

[ARG0 The great silver clouds on the horizon] [rel build] [ARG1 themselves] [ARGM-LOC on the pale water].

The first version of PropBank provided this type of annotations for a sub-corpus of Penn Treebank of size 1M words (a portion of Wall Street Journal sections) and for over than 4000 `rels` (verb senses).

PropBank Corpus (contd.)

Since then, PropBank annotations have been extended to larger corpora (OntoNotes Project) and for several languages other than English.

OntoNotes 5.0 is available for free from LDC, we also have LDC licenses for PropBank/WSJ/PTB for HHU affliates (means you can access the data).

For languages other than English, look at the links at `https://propbank.github.io/` (mostly part of OntoNotes).

PropBank Framesets

PropBank also provides an index of the predicates that it annotates in the so-called frameset format:

Each verb form has a frameset, e.g., the verb *build*. In the frameset, different meanings/senses of the verb are listed under different rolesets.

The id of the roleset identifies the meaning/category (e.g., Roleset id: build.01), which is followed by a short description of the meaning (e.g., construct for build.01), and a set of links to other language resources (if available/possible).

PropBank Framesets (contd.)

For each **roleset**, we have a **Roles** section, and a number of **Examples**.

Roles section list a mapping (and description) of `Args` to semantic role labels (*where it is available):

Arg0-PAG: builder (vnrole: 26.1-1-agent)

Arg1-PRD: construction (vnrole: 26.1-1-product)

Arg2-VSP: material, start state (vnrole: 26.1-1-material)

Arg4-PRD: end state (vnrole: 26.1-1-product)

PAG: Proto-Agent, PPT: Proto-Patient, VSP: Verb-Specific, PRD: Secondary predication: refers to or modifies another role, (see the list of functional tags in the PropBank annotation guidelines).

PropBank Framesets (contd.)

Example sections list various combination of arguments and their syntactic realization (as to my understanding, this is only a an inverted index of the corpus and may not contain all possible combinations):

Example: basic
Arg0: Stephen Wozniak and Steven Jobs
Rel: built
Arg1: the Apple I
Argm-loc: in a garage

Example: theoreti. . .
Arg0: John
Rel: built
Arg1: his house
Arg2: of straw

Browse the framesets from here:
`http://verbs.colorado.edu/propbank/framesets-english/`.

PropBank Limitations

PropBank has been [successfully] employed in several [attempts towards] semantic role labeling, see e.g., Carreras and Màrquez (2004) and its citations [and it has worked to a great extent!].

However, these efforts have been hindered by the core design principle behind PropBank.

PropBank chooses to employ a 'theoretically-neutral' approach by annotating semantic arguments of verbs (as seen), that is, PropBank's `Arg` labels have verb-specific meanings. In other words, *(labels are meaningful with respect to a single sense of a verb)*. This causes a few setbacks, which you can guess.

PropBank Limitations (contd.)

In a nutshell, PropBank's annotation style makes it difficult to make generalizations (of any type) beyond a single sense of a verb.

* It cannot tell us how arguments of a verb can be related to arguments of other verbs
* We cannot find two verbs' arguments that have the same role.
* We cannot generalize findings, e.g., over verb classes, etc.
* The scope of `Arg`-based inferences is limited to a verb sense.
* . . .

PropBank Limitations (contd.)

To build an accurate statistical data-driven model (i.e., to successfully train a machine learning algorithm), we need enough training data.

Since `Args` are valid only within the cope of a single sense of a verb, in machine learning applications, we often run into the problem of *lacking sufficient data for training statistical models*.

Put simply, too many classes to learn from a few training examples.

Several solutions have been proposed (e.g. as in Carreras and Màrquez (2004)) to merge `Args` from different verbs into a set of smaller role/arg categories.

PropBank Limitations (contd.)

Since `Arg0` and `Arg1` are often the proto-agent and proto-patient of predicates (disregarding the verb), a common solution is to consider them semantically (at least, approx.) equivalent.

Additionally, adjuncts (`ArgMs`) are annotated with the same set of labels across verbs: `ArgM-MNR`, `Argm-CAU`, `Argm-TMP`, `Argm-LOC`, . . . : These can (perhaps) be merged, too.

Training records for `Arg0` and `Arg1`, as well as `ArgMs`, can thus be derived from all verbs.

PropBank Limitations (contd.)

NB:

Arg0s and Arg1s account for a very large portion of the annotated Args in PropBank (61% in CoNLL sharedtask train data – Heavy tailed distribution again!).

70% of the arguments in the test data are Arg0, Arg1, ArgM-TMP.

No surprise a system trained based on this data can approximate PropBank annotations with the F-score of more than 80%.

Typical complaints about training data in machine learning applications are applicable here, too (out-of-domain text, . . . , the GIGO rule).

PropBank Limitations (contd.)

Next generation of PropBank-style semantic analysis: Abstract Meaning Representations (AMR)

Language Resource: The AMR Bank (https://catalog.ldc.upenn.edu/LDC2017T10).

In short: AMR represents predicate structure of sentences using a directed graph (in a notation named PENMAN): See slides from https://github.com/nschneid/amr-tutorial/tree/master/slides.

Useful links

PropBank Framesets: http://verbs.colorado.edu/propbank/framesets-english-aliases/

Human readable annotated sentences (PropBank I):
https://www.cs.rochester.edu/~gildea/PropBank/

List of `ArgM`s (adjunct-like modifiers): LOC: location
CAU: cause
EXT: extent
TMP: time
DIS: discourse connectives
PNC: purpose
ADV: general purpose
MNR: manner
NEG: negation marker
DIR: direction
MOD: modal verb

Bibliography

Carreras, X. and Màrquez, L. (2004). Introduction to the conll-2004 shared task: Semantic role labeling. In Ng, H. T. and Riloff, E., editors, *HLT-NAACL 2004 Workshop: Eighth Conference on Computational Natural Language Learning (CoNLL-2004)*, pages 89–97, Boston, Massachusetts, USA. Association for Computational Linguistics.