Frame Semantics

Tatiana Anikina

Seminar: Hot and Odd Topics in Semantics

Saarland University
04.06.2018
Outline

- Frame as a Linguistic Concept
- Historical Background
- Use Cases for NLP
- Frame Identification
- Semantic Role Labeling
- Open Issues

Let’s answer following questions:
What are the frames? When do we need them?
How to find frames in a sentence?
Let’s answer following questions:

What are the frames? When do we need them? How to find frames in a sentence?
Frame Mismatch

IF I HAVE 12 TOMATOES AND TAKE AWAY TWO...

WHAT IS THE DIFFERENCE?

EXACTLY. I DON'T LIKE TOMATOES, EITHER.

WHAT ARE YOU IN FOR THIS TIME?

SEMANTICS.

DETENTION.
Frame as a Linguistic Concept

*Charles Fillmore (1977a):*

*Meanings are relativized to scenes.*

**Frames** define how we *interpret our environment*, formulate & understand messages, create internal model of the world.

Language has not only grammar rules and lexicon items, it also has frames!
Frame as a Linguistic Concept

Particular words or speech formulas, grammatical choices are associated in memory with particular frames.

**Interactional Frames:** appropriate contexts of interaction (e.g. greeting frame).

**Cognitive/conceptual frames:** setting + protagonists + objects (e.g. buying event).
Frame as a Linguistic Concept

Our experiences are memorable because we have some **cognitive frames/schemes** for their interpretation.

E.g.: **grapefruit vs orange**

For 7-years-old children orange is an object which can be peeled and eaten piece by piece.

Grapefruit is an object which can be cut in half and eaten using a spoon.
Frame as a Linguistic Concept

**Breakfast** frame:
The first meal in the day after one wakes up. Can differ a lot depending on country/age/personal preferences.
E.g.: Breakfast in Russia vs Breakfast in Japan
Frame as a Linguistic Concept

**Breakfast** frame:
The first meal in the day after one wakes up. Can differ a lot depending on country/age/personal preferences.
Lexical frames are not easy to define:

**Good** + chair/steak/teacher ok!

**Good** + leaf/triangle/widow ???

**Money frame**: a medium used for exchanging goods and services.

**Money** = prize, change, ransom, tip, bribe...

The goal of frame semantics is to create a uniform representation for word/sentence meanings and text interpretations.
Snow Frame Example ;-)

11
Frames introduce situations (events) together with corresponding properties, participants and their roles. Frames provide an outline of the event and define a set of plausible questions (e.g. what/where/how).

Not all possible slots must be filled, some of them are optional.

E.g.: Sam bought new shoes.
Where did he buy the shoes? Why did he buy them? How expensive were the new shoes? ...
Frame as a Linguistic Concept

Frames **introduce situations (events)** together with corresponding properties, participants and their roles. Frames provide an **outline of the event** and define a set of **plausible questions** (e.g. what/where/how). Not all possible slots must be filled, some of them are optional.

*E.g.: Sam bought new shoes.*

Where did he buy the shoes? Why did he buy them? How expensive were the new shoes? ...
Frame as a Linguistic Concept

Commerce_buy
(buy, buyer, purchase)

Core

- Buyer
- Goods
- Explanation
- Manner

Non-Core

- Means
- Money
- Place
- Purpose
- Recipient
- Seller
- Time

Lexical units (LUs):
Words that evoke the frame (usually verbs)

Frame elements (FEs):
semantic roles

\[
\text{[Agent } Sam\text{] bought [Goods new shoes] [Place in Karstadt] [Time yesterday].}
\]
**Historical Background: FrameNet**

**FrameNet** is a large corpus of sentences annotated with frames. It was created in the late 90s.

**FrameNet** uses real language data from British National Corpus (BNC) and North American Newswire (NAN) corpora.

**FrameNet** contains:
- 8900 lexical units
- 625 frames
- 135000 sentences
**Historical Background: FrameNet**

---

**Frame Index**

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

Abandonment  
Abounding with  
Absorb heat  
Abundance  
Abusing  
Access scenario  
Accompaniment  
Accomplishment  
Accoutrements  
Accuracy  
Achieving first  
Active substance  
Activity  
Activity abandoned state  
Activity done state  
Activity finish  
Activity ongoing  
Activity pause  
Activity paused state  
Activity prepare  
Activity ready state  
Activity resume  
Activity start  
Activity stop

---

**Commerce_buy**

**Definition:**

These are words describing a basic commercial transaction involving a **Buyer** and a **Seller** exchanging **Money** and **Goods** taking the perspective of the **Buyer**. The words vary individually in the patterns of frame element realization they allow. For example, the typical pattern for the verb **BUY**: **Buyer buys Goods from Seller for Money**.

Abby bought a car from Robin for $5,000.

**FEs:**

**Core:**

**Buyer [Byr]**

The **Buyer** wants the **Goods** and offers **Money** to a **Seller** in exchange for them.  
Joes BOUGHT a coat.

Let BOUGHT a textbook from Abby.

**Goods [Gds]**

The FE **Goods** is anything (including labor or time, for example) which is exchanged for **Money** in a transaction.

Only one winner PURCHASED the paintings.
Historical Background: PropBank

**PropBank** is based on Penn TreeBank and is a data-driven resource.

**PropBank** is focused on verb annotation and has no abstraction beyond verb senses.

Argument roles are not frame-specific.

**PropBank** contains:

- 3324 frame files
- 113000 propositions
Historical Background: PropBank

Core arguments:
A0 – Agent, A1 – Patient or Theme

Non-core arguments:
AM-LOC, TMP, CAU, DIR, NEG etc.

<table>
<thead>
<tr>
<th>ID</th>
<th>Form</th>
<th>Lemma</th>
<th>P_Lemma</th>
<th>POS</th>
<th>PPOS</th>
<th>Feats</th>
<th>PFeats</th>
<th>Head</th>
<th>PHead</th>
<th>Deprel</th>
<th>PDeprel</th>
<th>IsPred</th>
<th>Pred</th>
<th>Args: buy.01</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Sam</td>
<td>sam</td>
<td>NNP</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2</td>
<td>SBJ</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>buy.01</td>
</tr>
<tr>
<td>2</td>
<td>bought</td>
<td>buy</td>
<td>VBD</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
<td>ROOT</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td>A0</td>
</tr>
<tr>
<td>3</td>
<td>new</td>
<td>new</td>
<td>JJ</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>4</td>
<td>NMOD</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>shoes</td>
<td>shoes</td>
<td>NNS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2</td>
<td>OBJ</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>A1</td>
</tr>
<tr>
<td>5</td>
<td>in</td>
<td>in</td>
<td>IN</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2</td>
<td>LOC</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Karstadt</td>
<td>karstadt</td>
<td>NNP</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>5</td>
<td>PMOD</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>AM-LOC</td>
</tr>
<tr>
<td>7</td>
<td>yesterday</td>
<td>yesterday</td>
<td>NN</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2</td>
<td>TMP</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>AM-TMP</td>
</tr>
<tr>
<td>8</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2</td>
<td>P</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Frames helps to capture “intermediate semantics”. Some words *evoke frames* (e.g. *buy* → *commerce*) which have arguments with specific roles (*buyer, goods*).

Frames are useful if we want to find the descriptions of the *same event* expressed with a different syntactic structure and using other words.

**Use Cases:**
Machine Translation, Q&A, Dialogue Systems
Use Cases for NLP

Frames helps to capture “intermediate semantics”. Some words **evoke frames** (e.g. *buy* $\rightarrow$ *commerce*) which have arguments with specific roles (*buyer*, *goods*).

Frames are useful if we want to find the descriptions of the **same event** expressed with a different syntactic structure and using other words.

**Use Cases:**

Machine Translation, Q&A, Dialogue Systems
Use Cases for NLP: Machine Translation

Create **lexical entries** for nouns, verbs, adjectives in language X that correspond to **FrameNet entries**.

Link parallel lexicon fragments by **common semantic frames**.

Parallel lexicon fragments provide valuable information about **different syntactic realizations** of frame semantics across different languages.

\[
\text{[Theme } \text{Dieses Buch} \text{] hat [Donor } \text{Anna} \text{] [Recipient } \text{mir} \text{] geschenkt.}
\]

\[
\text{[Donor } \text{Anna} \text{] presented [Recipient } \text{me} \text{] with [Theme } \text{this book} \text{].}
\]
Use Cases for NLP: Q&A

Answer extraction benefits from semantic role annotations. Predicates with similar semantics evoke the same frame and share the same roles.

FrameNet provides lists of surface realizations for semantic roles and annotated sample sentences.

Q: What \(\text{[Time year]}\) did \(\text{[Buyer the U.S.]}\) buy \(\text{[Goods Alaska]}\)?

A: \(\text{[Seller Russia]}\) sold \(\text{[Goods Alaska]}\) to \(\text{[Buyer the U.S.]}\) in \(\text{[Time 1867]}\).

QA system can use the fact that the U.S. has the same role buyer in both sentences.
Dialogue act recognition can incorporate syntactic and semantic relations between words using frames and semantic role labeling.

Can \([\text{Agent you} \text{ put } \text{Theme a picture} \text{ Goal on the wall}]\)?

<predicate: put-3.1>
<agent: you>
<theme: posters or pictures>
<destination: on the wall>
**Frame Identification**

**Task:** relate predicates to the frames which they evoke.

**Distributed Frame Identification** [Hermann, 2014]: Model takes *word embeddings* as an input and learns to identify semantic frames.

Word embeddings represent the *syntactic context of a predicate*.

The goal is to learn a matrix which maps high dimensional sparse representation into a lower dimensional space. The model learns *embeddings for all possible frames* and assigns the nearest frame label.
Frame Identification

**Task:** relate predicates to the frames which they evoke.

**Distributed Frame Identification** [Hermann, 2014]:
Model takes *word embeddings* as an input and learns to identify semantic frames.

Word embeddings represent the *syntactic context of a predicate*.

The goal is to learn a matrix which maps high dimensional sparse representation into a lower dimensional space. The model learns *embeddings for all possible frames* and assigns the nearest frame label.
**Frame Identification**

**Task:** relate predicates to the frames which they evoke.

**Distributed Frame Identification** [Hermann, 2014]: Model takes *word embeddings* as an input and learns to identify semantic frames.

Word embeddings represent the **syntactic context of a predicate**.

The goal is to learn a matrix which maps high dimensional sparse representation into a lower dimensional space. The model learns *embeddings for all possible frames* and assigns the nearest frame label.
Distributed Frame Identification:

Given a dependency parse (1) the model extracts all words matching paths from the predicate to its direct dependents (2).

Then model computes a distributed vector representations for words (3) and embedding vectors for each word are concatenated.

After that the linear transformation function is learned and parametrized by the context blocks (4).
Semantic Role Labeling

**Task:** find semantic roles for each argument of every predicate in a sentence.

**Basic Algorithm (in a supervised fashion):**

1. Prune unlikely constituents.
2. Use binary classification for remaining candidates (arg vs non-arg).
3. Assign possible semantic roles.
4. Re-rank (multiple labels for constituents).
5. Find best global label (semantic role for arg).
Task: find semantic roles for each argument of every predicate in a sentence.

Basic Algorithm (in a supervised fashion):
1. Prune unlikely constituents.
2. Use binary classification for remaining candidates (arg vs non-arg).
3. Assign possible semantic roles.
4. Re-rank (multiple labels for constituents).
5. Find best global label (semantic role for arg).
Semantic Role Labeling

**Task:** find semantic roles for each argument of every predicate in a sentence.

**Basic Algorithm (in a supervised fashion):**
1. Prune unlikely constituents.
2. Use binary classification for remaining candidates (arg vs non-arg).
3. Assign possible semantic roles.
4. Re-rank (multiple labels for constituents).
5. Find best global label (semantic role for arg).
**Semantic Role Labeling**

**Task:** find semantic roles for each argument of every predicate in a sentence.

**Basic Algorithm (in a supervised fashion):**
1. Prune unlikely constituents.
2. Use binary classification for remaining candidates (arg vs non-arg).
3. Assign possible semantic roles.
4. Re-rank (multiple labels for constituents).
5. Find best global label (semantic role for arg).
**Semantic Role Labeling**

**Task:** find semantic roles for each argument of every predicate in a sentence.

**Basic Algorithm (in a supervised fashion):**

1. Prune unlikely constituents.
2. Use binary classification for remaining candidates (arg vs non-arg).
3. Assign possible semantic roles.
4. Re-rank (multiple labels for constituents).
5. Find best global label (semantic role for arg).
**Semantic Role Labeling**

**Task:** find semantic roles for each argument of every predicate in a sentence.

**Basic Algorithm (in a supervised fashion):**

1. Prune unlikely constituents.
2. Use binary classification for remaining candidates (arg vs non-arg).
3. Assign possible semantic roles.
4. Re-rank (multiple labels for constituents).
5. Find best global label (semantic role for arg).
Semantic Role Labeling

**Baseline features**  
[Gildea & Jurafsky, 2002]

Constituent Independent
- Target predicate (lemma)
- Voice
- Subcategorization

Constituent Specific
- Path
- Position (*left, right*)
- Phrase Type
- Governing Category (*S* or *VP*)
- Head Word

<table>
<thead>
<tr>
<th>Target</th>
<th>broke</th>
</tr>
</thead>
<tbody>
<tr>
<td>Voice</td>
<td>active</td>
</tr>
<tr>
<td>Subcategorization</td>
<td>VP→VBD NP</td>
</tr>
<tr>
<td>Path</td>
<td>VBD↑VP↑S↓NP</td>
</tr>
<tr>
<td>Position</td>
<td>left</td>
</tr>
<tr>
<td>Phrase Type</td>
<td>NP</td>
</tr>
<tr>
<td>Gov Cat</td>
<td>S</td>
</tr>
<tr>
<td>Head Word</td>
<td>She</td>
</tr>
</tbody>
</table>
Semantic Role Labeling: advanced

Neural Semantic Role Labeling with Dependency Path Embeddings
[Roth et al., 2016]

Motivation: sparse lexico-syntactic features in the training data (control & raising verbs, nested conjunctions etc.)

Approach: LSTM to model semantic relationships between the predicate and its arguments.
Path embeddings = vector representations of dependency paths
Neural Semantic Role Labeling with Dependency Path Embeddings

[Roth et al., 2016]

**Motivation:** sparse lexico-syntactic features in the training data (control & raising verbs, nested conjunctions etc.)

**Approach:** LSTM to model semantic relationships between the predicate and its arguments.

Path embeddings = vector representations of dependency paths
Semantic Role Labeling: advanced

Neural Semantic Role Labeling with Dependency Path Embeddings

[Roth et al., 2016]

**Motivation:** sparse lexico-syntactic features in the training data (control & raising verbs, nested conjunctions etc.)

**Approach:** LSTM to model semantic relationships between the predicate and its arguments.

Path embeddings = vector representations of dependency paths
Encoding dependencies as embeddings

Dependency path between predicate *raising* and argument *he*.

Embedding computation for the path from *raising* to *he*.
Semantic Role Labeling: advanced

Joint learning of path embeddings + binary features:

1. The path sequence is fed into a LSTM layer.
2. Hidden layer $h$ combines embedding $e$ and binary input features $B$.
3. Output layer $s$ assigns the highest probable class label $c$. 
## Semantic Role Labeling: advanced

### Results:

<table>
<thead>
<tr>
<th>System (local, single)</th>
<th>P</th>
<th>R</th>
<th>F₁</th>
</tr>
</thead>
<tbody>
<tr>
<td>Björkelund et al. (2010)</td>
<td>87.1</td>
<td>84.5</td>
<td>85.8</td>
</tr>
<tr>
<td>Lei et al. (2015)</td>
<td>—</td>
<td>—</td>
<td>86.6</td>
</tr>
<tr>
<td>FitzGerald et al. (2015)</td>
<td>—</td>
<td>—</td>
<td>86.7</td>
</tr>
<tr>
<td>PathLSTM w/o reranker</td>
<td>88.1</td>
<td>85.3</td>
<td>86.7</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>System (global, single)</th>
<th>P</th>
<th>R</th>
<th>F₁</th>
</tr>
</thead>
<tbody>
<tr>
<td>Björkelund et al. (2010)</td>
<td>88.6</td>
<td>85.2</td>
<td>86.9</td>
</tr>
<tr>
<td>Roth and Woodsend (2014)³</td>
<td>—</td>
<td>—</td>
<td>86.3</td>
</tr>
<tr>
<td>FitzGerald et al. (2015)</td>
<td>—</td>
<td>—</td>
<td>87.3</td>
</tr>
<tr>
<td>PathLSTM</td>
<td>90.0</td>
<td>85.5</td>
<td>87.7</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>System (global, ensemble)</th>
<th>P</th>
<th>R</th>
<th>F₁</th>
</tr>
</thead>
<tbody>
<tr>
<td>FitzGerald et al. 10 models</td>
<td>—</td>
<td>—</td>
<td>87.7</td>
</tr>
<tr>
<td>PathLSTM 3 models</td>
<td>90.3</td>
<td>85.7</td>
<td>87.9</td>
</tr>
</tbody>
</table>

---

CoNLL-2009 in-domain test set.  
CoNLL-2009 out-of-domain test set.
Unsupervised Induction of Frame-Semantic Representations

[Modi et al., 2012]

**Motivation:** jointly induce semantic frames and their roles without using pre-annotated data.

**Approach:** associate arguments with frame-specific argument keys (verb voice, arg position, syntactic rel to governor).

Semantic roles = clusters of arg keys; semantic frames are induced based on distribution.
Semantic Role Labeling: advanced

Unsupervised Induction of Frame-Semantic Representations

[Modi et al., 2012]

Motivation: jointly induce semantic frames and their roles without using pre-annotated data.

Approach: associate arguments with frame-specific argument keys (verb voice, arg position, syntactic rel to governor).

Semantic roles = clusters of arg keys; semantic frames are induced based on distribution.
Semantic Role Labeling: advanced

Unsupervised Induction of Frame-Semantic Representations

[Modi et al., 2012]

**Motivation:** jointly induce semantic frames and their roles without using pre-annotated data.

**Approach:** associate arguments with frame-specific argument keys (verb voice, arg position, syntactic rel to governor).

Semantic roles = clusters of arg keys; semantic frames are induced based on distribution.
Semantic Role Labeling: advanced

Frame-semantic parsing:

Parameters:
for each frame $f = 1, 2, \ldots$:
\[ \phi_f \sim DP(\gamma, H^{(P)}) \] [distrib of lexical units]
\[ B_f \sim CRP(\alpha) \] [partition of arg keys]
for each role $r \in B_f$:
\[ \theta_{f,r} \sim DP(\beta, H^{(A)}) \] [distrib of arg fillers]
\[ \psi_{f,r} \sim Beta(\eta_0, \eta_1) \] [geom distr for dup roles]

Data Generation:
for each frame $f = 1, 2, \ldots$:
for each occurrence of frame $f$:
\[ p \sim \phi_f \] [draw a lexical unit]
for every role $r \in B_f$:
if $[n \sim Unif(0, 1)] = 1$:
\[ \text{GenArgument}(f, r) \] [role appears at least once]
while $[n \sim \psi_{f,r}] = 1$:
\[ \text{GenArgument}(f, r) \] [continue generation]
\[ \text{GenArgument}(f, r) \] [draw more args]

GenArgument$(f, r)$:
\[ k_{f,r} \sim Unif(1, \ldots, |r|) \] [draw arg key]
\[ x_{f,r} \sim \theta_{f,r} \] [draw arg filler]
### Results (induced vs FrameNet annotation):

<table>
<thead>
<tr>
<th>Induced frames</th>
<th>FrameNet frames corresponding to the verbs</th>
</tr>
</thead>
<tbody>
<tr>
<td>(rush::dash::tiptoe)</td>
<td>rush : <a href="150">Self_motion</a> <a href="19">Fluidic_motion</a></td>
</tr>
<tr>
<td></td>
<td>dash : <a href="100">Self_motion</a></td>
</tr>
<tr>
<td></td>
<td>tiptoe : <a href="114">Self_motion</a></td>
</tr>
<tr>
<td>(ratify::sign::accede)</td>
<td>ratify : <a href="41">Ratification</a></td>
</tr>
<tr>
<td></td>
<td>sign : <a href="81">Sign_agreement</a> <a href="18">Hiring</a> <a href="1">Text_Creation</a></td>
</tr>
<tr>
<td></td>
<td>accede : <a href="31">Sign_Agreement</a></td>
</tr>
<tr>
<td>(crane::lean::bustle)</td>
<td>crane : <a href="26">Body_movement</a></td>
</tr>
<tr>
<td></td>
<td>lean : <a href="70">Change_posture</a> <a href="22">Placing</a> <a href="12">Posture</a></td>
</tr>
<tr>
<td></td>
<td>bustle : <a href="55">Self_motion</a></td>
</tr>
<tr>
<td>(cool::heat::warm)</td>
<td>cool : <a href="27">Cause_temperature_change</a></td>
</tr>
<tr>
<td></td>
<td>heat : <a href="52">Cause_temperature_change</a></td>
</tr>
<tr>
<td></td>
<td>warm : <a href="41">Cause_temperature_change</a> <a href="16">Inchoative_change_of_temperature</a></td>
</tr>
<tr>
<td>(want::fib::dare)</td>
<td>want : <a href="105">Desiring</a> <a href="44">Possession</a></td>
</tr>
<tr>
<td></td>
<td>fib : <a href="9">Prevarication</a></td>
</tr>
<tr>
<td></td>
<td>dare : <a href="21">Daring</a></td>
</tr>
<tr>
<td>(encourage::intimidate::confuse)</td>
<td>encourage : <a href="49">Stimulus_focus</a></td>
</tr>
<tr>
<td></td>
<td>intimidate : <a href="26">Stimulus_focus</a></td>
</tr>
<tr>
<td></td>
<td>confuse : <a href="45">Stimulus_focus</a></td>
</tr>
</tbody>
</table>
Semantic Role Labeling: advanced

Results (role-labeling vs frame-labeling):

<table>
<thead>
<tr>
<th></th>
<th>PU</th>
<th>CO</th>
<th>F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Our approach</td>
<td>78.9</td>
<td>71.0</td>
<td>74.8</td>
</tr>
<tr>
<td>NoFrameInduction</td>
<td>79.2</td>
<td>70.7</td>
<td>74.7</td>
</tr>
<tr>
<td>SyntF</td>
<td>69.9</td>
<td>73.3</td>
<td>71.6</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>PU</th>
<th>CO</th>
<th>F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Our approach</td>
<td>77.9</td>
<td>31.4</td>
<td>44.7</td>
</tr>
<tr>
<td>NoClustering</td>
<td>80.8</td>
<td>29.0</td>
<td>42.7</td>
</tr>
</tbody>
</table>

Role labeling performance. Frame labeling performance.

Issues: only 25% of verbs belong to induced multi-verb clusters. This could be due to the relatively small size of FrameNet, not enough evidence for clustering.
Semantic Role Labeling: advanced

Joint Semantic Parsers from Disjoint Data
[Peng et al., 2018]

Motivation: SRL with multiple datasets, combine frame semantic parsing with dependency parsing

Approach: use latent variables to extend cross-task part scoring if datasets do not overlap; learn cross-task using different formalisms (span vs dependency-based parsing); prune the space of cross task parts.
Motivation: SRL with multiple datasets, combine frame semantic parsing with dependency parsing.

Approach: use latent variables to extend cross-task part scoring if datasets do not overlap; learn cross-task using different formalisms (span vs dependency-based parsing); prune the space of cross task parts.
Semantic Role Labeling: advanced

Joint Semantic Parsers from Disjoint Data

[Peng et al., 2018]

**Motivation:** SRL with multiple datasets, combine frame semantic parsing with dependency parsing

**Approach:** use latent variables to extend cross-task part scoring; learn cross-task with different formalisms (span vs dependency-based parsing); prune the space of cross task parts.
Semantic Role Labeling: advanced

Frame-Semantic Parsing (FrameNet style)
For a sentence X frame-semantic parse consists of:
1. Set of **targets** (short spans) that evokes a frame.
2. For each target t, the **frame** f that it evokes.
3. For each frame f, a set of **non-overlapping arg** spans in the sentence: arg = (i, j, r)
   start token index i; end token index j; role label r

[Diagram of a sentence with semantic roles marked]
Semantic Role Labeling: advanced

Semantic Dependency Parsing (PropBank style)
Each semantic dependency represents a labeled, directed edge between two words.
A single token with a top label indicates the main predicate.
head of an arc = target in frame semantics
destination = argument
label = role
Semantic Role Labeling: advanced

Idea:

(1) Given a sentence $x$ and target $t$ ($l$ for latent variable), find a set of valid frame-semantic parses as $Y(x, t, l)$ and dependency parses as $Z(x)$.

$$(\hat{y}, \hat{z}) = \arg \max_{(y, z) \in Y(x, t, l) \times Z(x)} S(y, z, x, t, l).$$
(2)

Learn a parameterized function $S$ that scores candidate parses. **Jointly predict** a frame-semantic parse & dependency graph by selecting the best candidates.

**Overall score $S$** is a sum of frame SRL score $S_f$, dependency score $S_d$, and a cross-task score $S_c$

$$S(y, z, x, t, \ell) = S_f(y, x, t, \ell) + S_d(z, x) + S_c(y, z, x, t, \ell).$$
Semantic Role Labeling: advanced

1. Learn contextualized token & span vectors with a bidirectional LSTM & multilayer perceptrons.

2. Learn embeddings for lexical units, frames, roles, and arc labels.

3. Combine all representations into a single scalar score using a (learned) low-rank multilinear mapping.

4. Apply scoring function to candidate arguments.
   Each candidate consists of:
   span + frame-dependent role + target + lexical unit
Semantic Role Labeling: advanced

1. Learn contextualized token & span vectors with a bidirectional LSTM & multilayer perceptrons.

2. **Learn embeddings** for lexical units, frames, roles, and arc labels.

3. **Combine all representations** into a single scalar score using a (learned) low-rank multilinear mapping.

4. **Apply scoring function** to candidate arguments.

   Each candidate consists of:

   span + frame-dependent role + target + lexical unit
1. Learn contextualized token & span vectors with a bidirectional LSTM & multilayer perceptrons.

2. Learn embeddings for lexical units, frames, roles, and arc labels.

3. Combine all representations into a single scalar score using a (learned) low-rank multilinear mapping.

4. Apply scoring function to candidate arguments.

   Each candidate consists of:

   span + frame-dependent role + target + lexical unit
1. Learn contextualized token & span vectors with a bidirectional LSTM & multilayer perceptrons.
2. Learn embeddings for lexical units, frames, roles, and arc labels.
3. Combine all representations into a single scalar score using a (learned) low-rank multilinear mapping.
4. Apply scoring function to candidate arguments.
   Each candidate consists of:
   span + frame-dependent role + target + lexical unit
## Results:

<table>
<thead>
<tr>
<th>Model</th>
<th>Prec.</th>
<th>Rec.</th>
<th>$F_1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Roth</td>
<td>72.2</td>
<td>68.0</td>
<td>70.0</td>
</tr>
<tr>
<td>Täckström</td>
<td>75.4</td>
<td>65.8</td>
<td>70.3</td>
</tr>
<tr>
<td>FitzGerald</td>
<td>74.8</td>
<td>65.5</td>
<td>69.9</td>
</tr>
<tr>
<td>FitzGerald (10×)</td>
<td>75.0</td>
<td>67.3</td>
<td>70.9</td>
</tr>
<tr>
<td>open-SESAME</td>
<td>71.0</td>
<td>67.8</td>
<td>69.4</td>
</tr>
<tr>
<td>open-SESAME (5×)</td>
<td>71.2</td>
<td>70.5</td>
<td>70.9</td>
</tr>
<tr>
<td>Yang and Mitchell (REL)</td>
<td>77.1</td>
<td>68.7</td>
<td>72.7</td>
</tr>
<tr>
<td>†+Yang and Mitchell (ALL)</td>
<td>78.8</td>
<td>74.5</td>
<td>76.6</td>
</tr>
<tr>
<td>†This work (FULL)</td>
<td>80.4</td>
<td>73.5</td>
<td>76.8</td>
</tr>
<tr>
<td>†This work (FULL, 2×)</td>
<td><strong>80.4</strong></td>
<td><strong>74.7</strong></td>
<td><strong>77.4</strong></td>
</tr>
<tr>
<td>†This work (BASIC)</td>
<td>79.2</td>
<td>71.7</td>
<td>75.3</td>
</tr>
<tr>
<td>†This work (NOCTP)</td>
<td>76.9</td>
<td>74.8</td>
<td>75.8</td>
</tr>
<tr>
<td>Hartmann</td>
<td>87.6</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Yang and Mitchell</td>
<td>88.2</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Hermann</td>
<td>88.4</td>
<td>73.1</td>
<td></td>
</tr>
<tr>
<td>†This work (BASIC)</td>
<td>89.2</td>
<td>76.3</td>
<td></td>
</tr>
<tr>
<td>†This work (NOCTP)</td>
<td>89.2</td>
<td>76.4</td>
<td></td>
</tr>
<tr>
<td>†This work (FULL)</td>
<td>89.9</td>
<td>77.7</td>
<td></td>
</tr>
<tr>
<td>†This work (FULL, 2×)</td>
<td><strong>90.0</strong></td>
<td><strong>78.0</strong></td>
<td></td>
</tr>
</tbody>
</table>

Frame + SRL performance:

† is for models jointly predicting frames & arguments, other systems are two-stage pipelines.

Frame labeling performance.
Open Issues for SRL & Frames

1. **SRL degrades across different domains.**
   
   WSJ → Brown Corpus (12% decrease in F-measure)

2. **SRL depends on pre-processing** (POS-tagging, parsing).
   
   Errors can accumulate!

3. **Frame coverage is challenging.**
   
   FrameNet doesn’t provide universal generalizations.
Resources & Tools to try/play with

FrameNet: http://framenet.icsi.berkeley.edu/
PropBank: https://propbank.github.io/

Useful Tools:

MateTools (SRL): https://code.google.com/archive/p/mate-tools/
MatePlus (SRL): https://github.com/microth/mateplus
NeurboParser: https://github.com/Noahs-ARK/NeurboParser
Let’s move from rather static **frames** or **pictures** of events to more dynamic representations.

We can combine frames together & have more complex relations between them.

If we talk about **frames as pictures** than **scripts** are more about **movies**.
Let’s learn **scripts**!

**Recipe**
**CHOCOLATE CAKE**

- 4 oz. chocolate
- 3 eggs
- 1 cup butter
- 1 tsp. vanilla
- 2 cups sugar
- 1 cup flour

Melt chocolate and butter. Stir sugar into melted chocolate. Stir in eggs and vanilla. Mix in flour. Spread mix in greased pan. Bake at 350° for 40 minutes or until inserted fork comes out almost clean. Cool in pan before eating.

**Program Code**

Declare variables:
- chocolate
- eggs
- mix
- butter
- vanilla
- sugar
- flour

Mix = melted (4*chocolate + butter)
Mix = stir (Mix + (2*sugar))
Mix = stir (Mix + (3*eggs) + vanilla)
Mix = mix + flour
Spread (mix)
While not clean (fork)
Bake (mix, 350)

Test with fork

Eat

Remove from Oven

Let Cool

Not Ready

Ready

Spread in Pan

Bake at 350°
References


References

All screenshots with evaluation results & formulas were taken from the corresponding research papers.

Other pictures:

front page frame: https://myframe.co/frame-nlp/

p. 7 grapefruit: http://www.thepinsta.com/parts-of-a-grapefruit_5AmS7nbDfZzy3hI3dmJDj10PeCwJMia%7CleRHnPz6f4/

p. 4 semantics: https://translationmusings.com/2013/04/

p. 9 Russian breakfast: https://www.yummly.com/recipes/russian-breakfast

p. 9 Japanese breakfast: http://www.japansubculture.com/zeppin-%E7%B5%B6%E5%93%81-3-wachoshoku-japanese-breakfast-rules/
