Stacked Ensembles of Information Extractors by Combining Supervised and Unsupervised Approaches

Nazneen Rajani and Ray Mooney

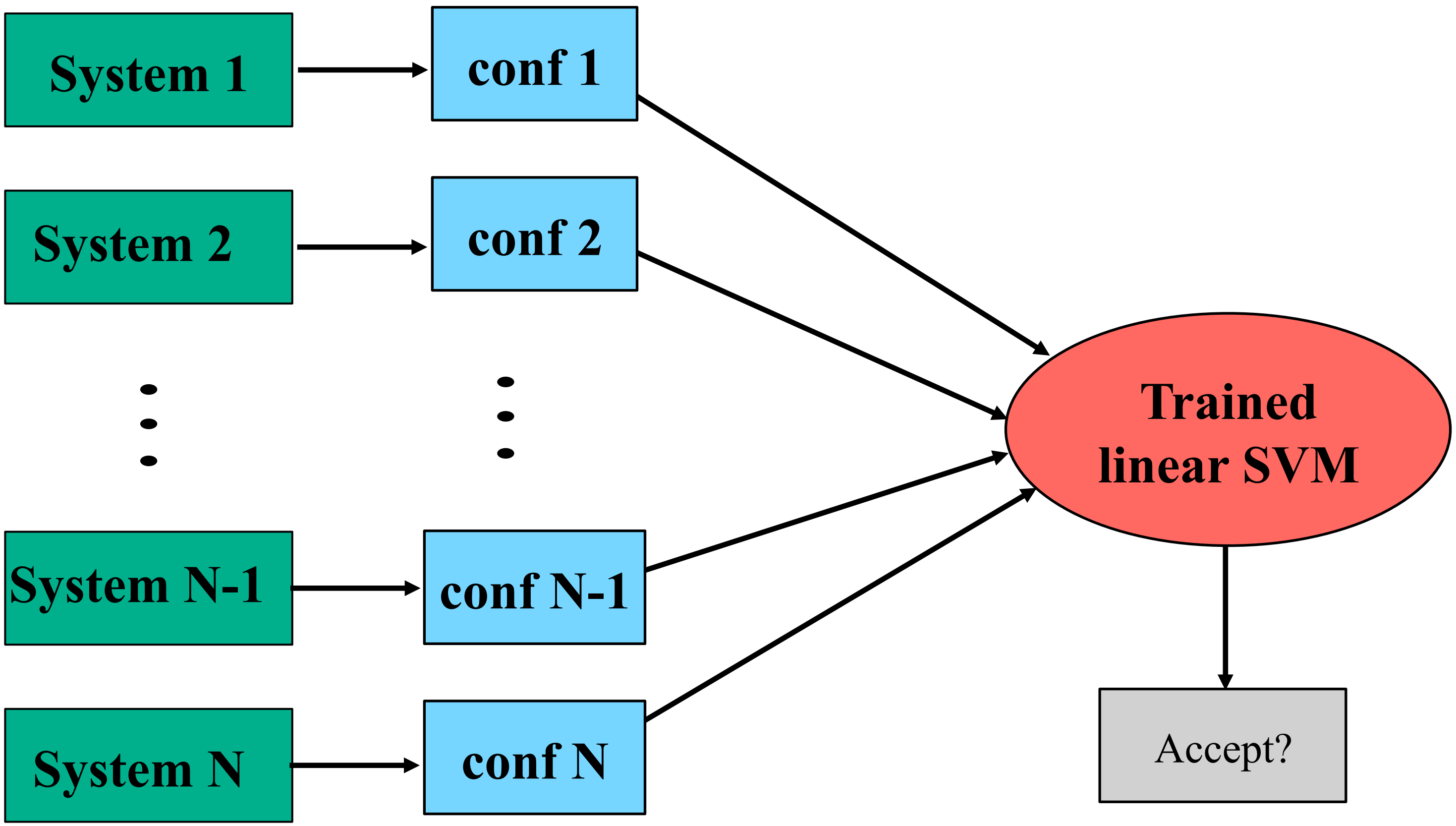
NIST KBP Evaluation

UT Austin

Stacking

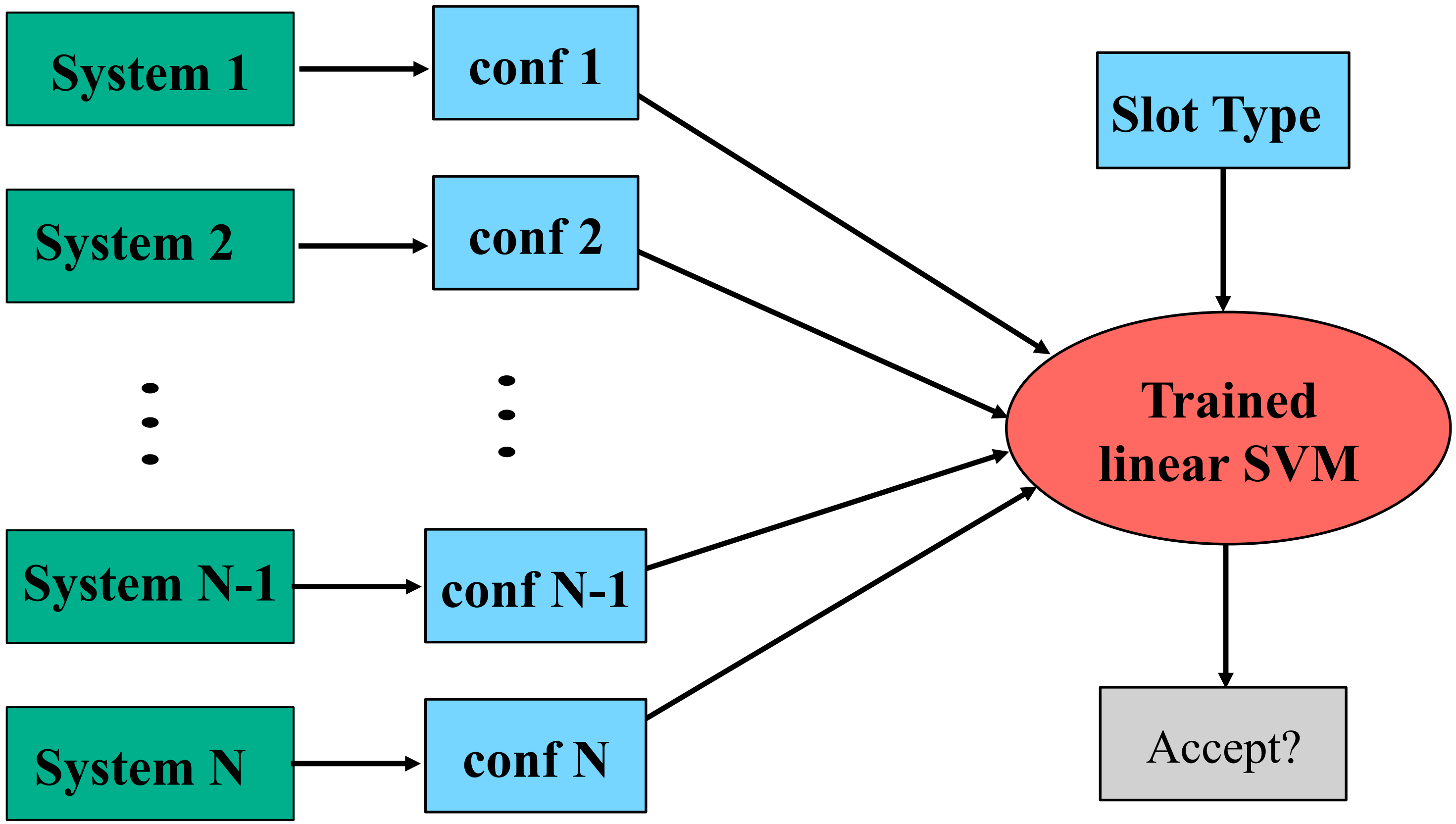
(Wolpert, 1992)

For a given proposed slot-fill, e.g. `spouse(Barack, Michelle)`, combine confidences from multiple systems:



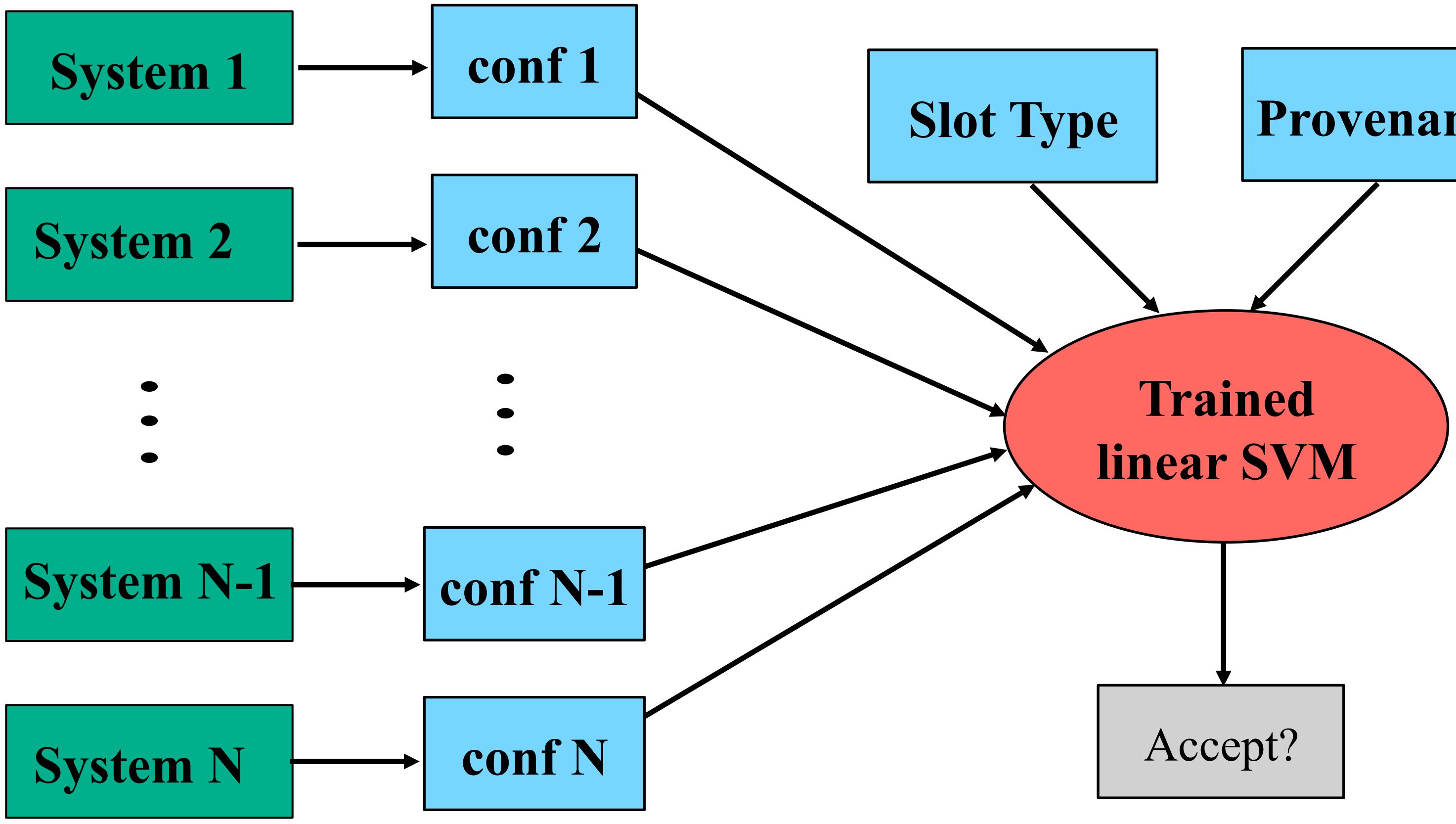
Stacking with Features

For a given proposed slot-fill, e.g. `spouse(Barack, Michelle)`, combine confidences from multiple systems:



Stacking with Features

For a given proposed slot-fill, e.g. `spouse(Barack, Michelle)`, combine confidences from multiple systems:



Document Provenance Feature

- For a given query and slot, for each system, *i*, there is a feature DP_i:
  - N systems provide a fill for the slot.
  - Of these, n give same provenance *docid* as *i*.
  - $DP_i = n/N$ is the document provenance score.
- Measures extent to which systems agree on document provenance of the slot fill.

Offset Provenance Feature

- Degree of overlap between systems' provenance strings (prov).
- Uses Jaccard similarity coefficient.
- For a given query and slot, for each system, *i*, there is a feature OP_i :
  - *N* systems provide a fill with same *docid*
  - Offset provenance for a system *i* is calculated as:

$$OP_i = \frac{1}{|N|} \times \sum_{j \in N, j \neq i} \frac{|\text{prov(i)} \cap \text{prov(j)}|}{|\text{prov(i)} \cup \text{prov(j)}|}$$

  - Systems with different *docid* have zero OP

Document Similarity Feature

- KBP queries have the following format:

```
<query id="CSSF15_ENG_0006e06ebf">
  <name>Walmart</name>
  <docid>ad4358e0c4c18e472c13bbc27a6b7ca5</docid>
  <beg>232</beg>
  <end>238</end>
  <enttype>org</enttype>
  <slot0>org:date_dissolved</slot0>
</query>
```

- For each system, measure the similarity between the document in the provenance and query document.
- For a given query and slot fill, each system contributes a score as a feature or zero.

Total Number of Features

- Vanilla stacking $\longrightarrow$ confidence scores $\longrightarrow$ #systems
- Document provenance feature $\longrightarrow$ #systems
- Offset provenance feature $\longrightarrow$ #systems
- Document similarity feature $\longrightarrow$ #systems
- Slot type $\longrightarrow$ 60 (per + org + gpe)
- #systems = 38 in 2015

Unsupervised Learning on Remaining Systems

- Stacking restricts us to common systems between years.
- Use unsupervised techniques to learn a confidence score for all the remaining systems combined.
- We use constrained optimization (Weng et al., 2013) for single valued and list slots separately.
- Aggregate “raw” confidence values produced by individual systems into a single aggregated confidence value for each slot.

Unsupervised Learning on Remaining Systems

- For example:

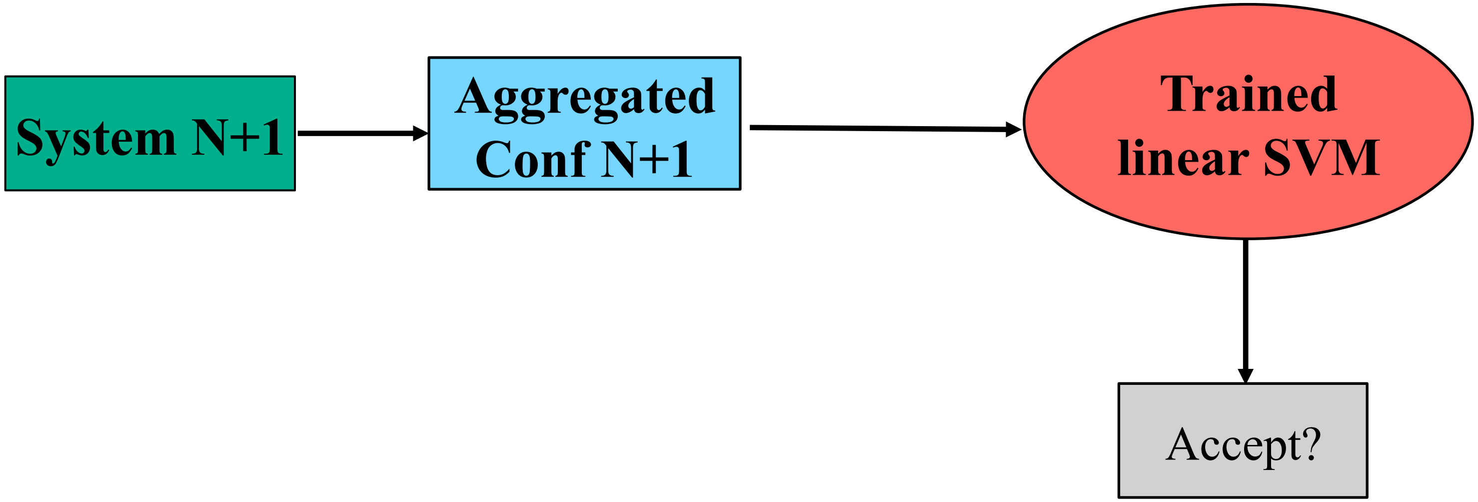
Harvey Milk	per:country_of_birth	new york city	SFV2015_SF_10_2	0.7892
Harvey Milk	per:country_of_birth	united states	SFV2015_SF_18_1	0.2291
Harvey Milk	per:country_of_birth	united states	SFV2015_SF_18_2	0.3437

- For a given query and slot, for each slot fill the aggregated confidence score is produced

Harvey Milk	per:country_of_birth	new york city	0.36823
Harvey Milk	per:country_of_birth	united states	0.63177

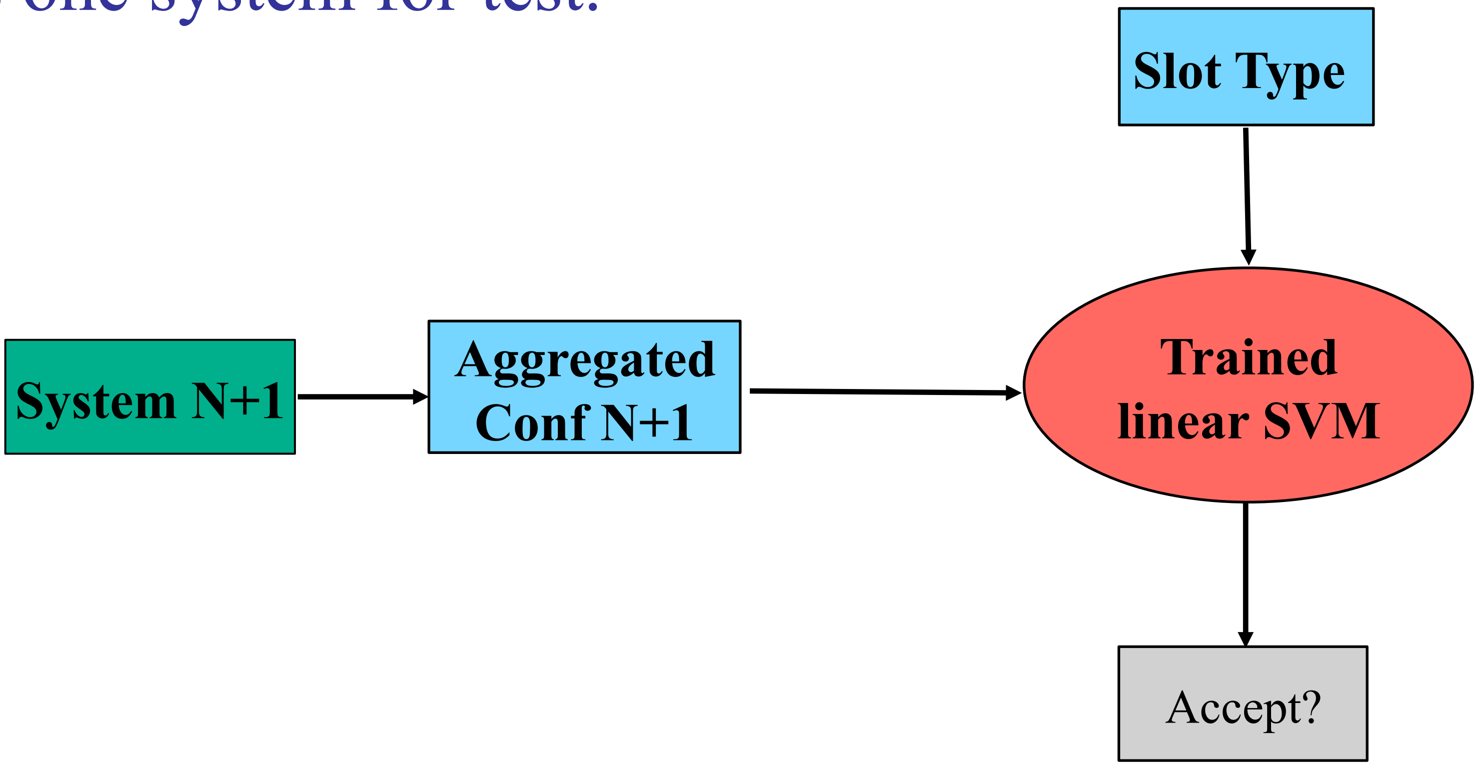
Stacking over the Unsupervised Approach

- Train the stacker on previous year's unsupervised aggregated confidence scores treating it as one system.
- Similarly all the unsupervised output can be considered as one system for test.



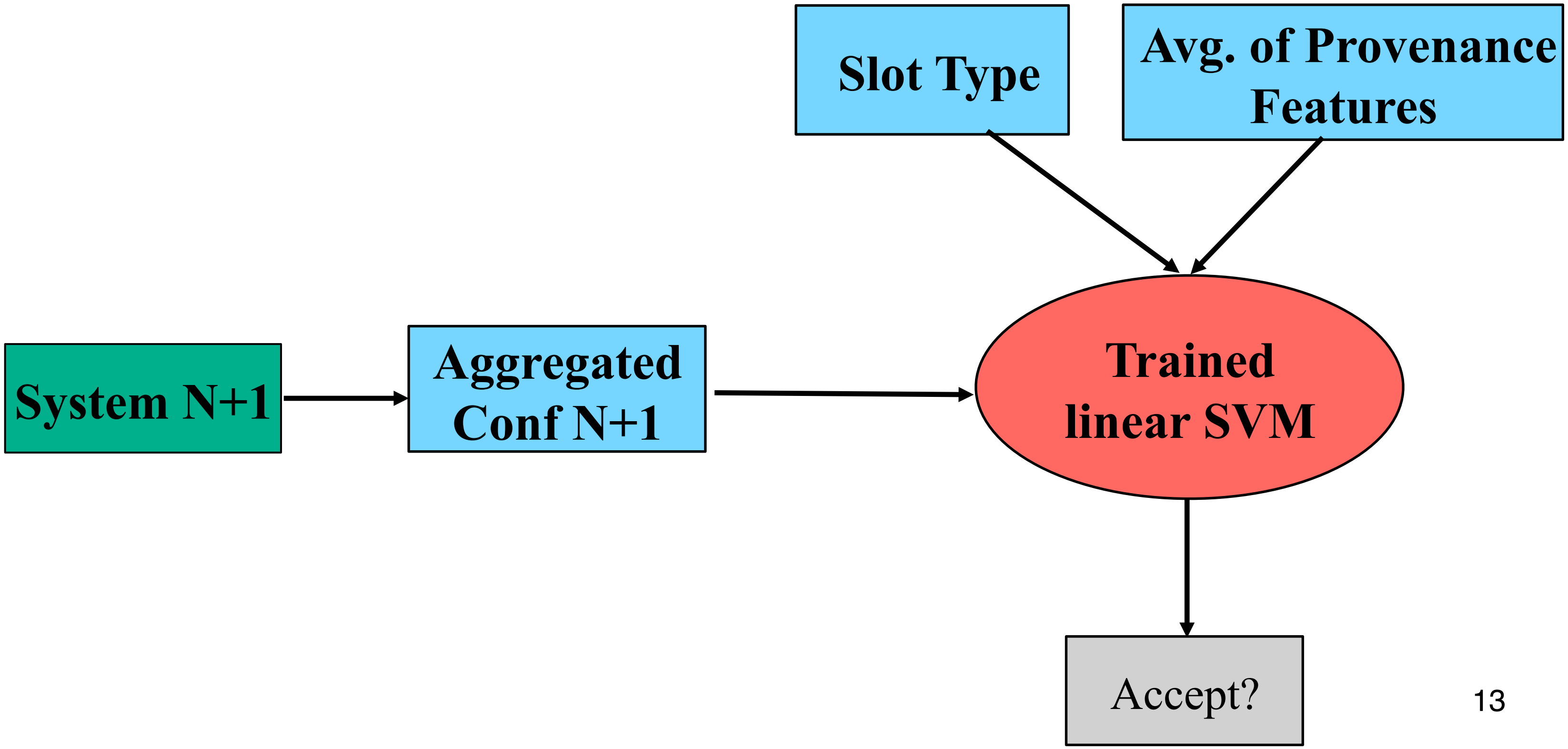
Stacking over the Unsupervised Approach

- Train the stacker on previous year's unsupervised aggregated confidence scores treating it as one system.
- Similarly all the unsupervised output can be considered as one system for test.



Stacking over the Unsupervised Approach

- Train the stacker on previous year's unsupervised aggregated confidence scores treating it as one system.
- Similarly all the unsupervised output can be considered as one system for test.



Combining the Stacking and Unsupervised Approaches

- For single-valued slot fill, add the slot fill with highest confidence if multiple fills are labeled correct.
- For a list-value slot fill, add all the slot fills labeled correct, only if the confidence score exceeds a threshold
  – This threshold is derived for each list-value slot type based on 2014 data.

Datasets for 2015

- 2015 Slot Filler Validation (SFV) data
  – 18 Teams
  – 70 Systems
- 38 common systems from 10 teams
  – Stanford (1)
  – UMass (4)
  – UW (3)
  – CMUML (3)
  – BUPT_PRIS (5)
  – CIS (5)
  – ICTCAS (4)
  – NYU (4)
  – STARAI (5)
  – Ugent (4)

Filtering Subtask

- Aim: Improve precision of individual systems.
- For a given query and slot:
  - If the stacker predicts that the hop-0 slot fill is incorrect,
  - But the hop-1 slot fill is correct,
  - Then reject both hop-0 and hop-1 slot fills.

Ensembling Subtask

- Aim: Ensemble individual systems to maximize F1.
- For a given query and slot:
  - If the stacker predicts that the hop-0 slot fill is incorrect,
  - But the hop-1slot fill is correct,
  - Then accept both hop-0 and hop-1 slot fills by including the corresponding hop-0 slot fill.

Results

- 2015 Slot Filler Validation (SFV) dataset
  - Partially evaluated set of queries made available to all teams

Approach	Precision	Recall	F1
Unsupervised on common systems data	0.402	0.103	0.164
Unsupervised on all data (JHU)	0.455	0.292	0.355
Unsupervised with additional features	**0.637**	0.252	0.361
Stacking on common systems data	0.453	**0.314**	0.371
Stacking and Unsupervised combined on all data	0.542	0.285	**0.374**

Official Results

- Cold Start

Approach	Precision	Recall	F1
Hop-0	**0.6570**	0.1435	0.2356
Hop-1	0.0	0.0	0.0
All	**0.6570**	0.0813	0.1447

- SFV

Approach	Precision	Recall	F1
Hop-0	0.3210	0.3831	**0.3494**
Hop-1	0.0341	0.0033	0.0060
All	0.3029	0.2105	0.2484

Conclusion

- Stacked meta-classifier produces high precision ensemble.
- Unsupervised approach works well on single value slots but fails on list value slots.
- Only considering common systems affects our performance even if the remaining systems do not perform well by themselves.
- Combination of stacking and unsupervised approaches performs better than both individual approaches.

Future Work

- Features related to the entity type which is given by the CSSF systems.
- Ensembling round-1 and round-2 slot fills separately and have different features for each.
- More sophisticated approach for combining the slot fills.
    - Multi-level stacking.

References

- Nazneen Fatema Rajani, Vidhoon Vishwananthan, Yinon Bentor, and Raymond Mooney. Stacked ensembles of information extractors for knowledge-base population. In proceedings on the Association for Computational Linguistics, 2015.
- I-Jeng Wang, Edwina Liu, Cash Costello, and Christine Piatko. 2013. JHUAPL TAC-KBP2013 slot filler validation system. In Proceedings of the Sixth Text Analysis Conference.
- David H. Wolpert. 1992. Stacked generalization. Neural Networks, 5:241–259.

Thank You