Stacked Ensembles of Information Extractors for Knowledge-Base Population by Combining Supervised and Unsupervised Approaches

Nazneen Rajani

Research Preparation Exam

University of Texas at Austin

Information Extraction (IE)

- Automatically extract information from large corpus of unlabeled text.
- Information Extraction (IE) systems:
  - Extract clean and factual information
  - Find and understand relevant parts of text
  - Gather information from many pieces of text
  - Produce a structured representation of that information:
    - relations (in the database sense),a.k.a, Knowledge Base(KB)
    - For example:

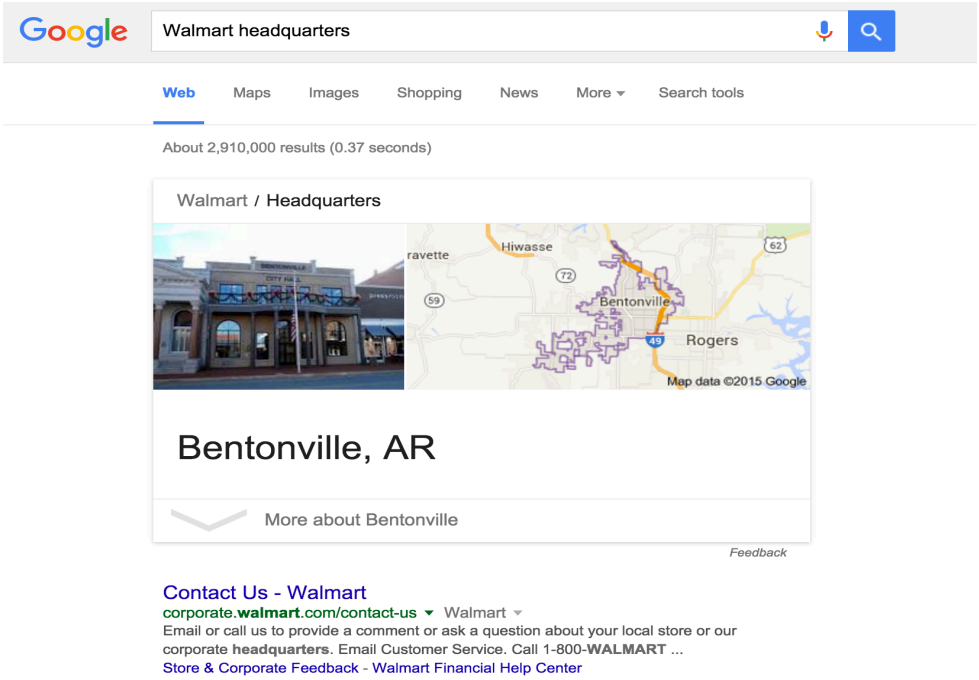
Michelle Obama is an American lawyer and writer. She is married to the 44th and current president of United States, Barack Obama.

IE

spouse("Barack Obama", "Michelle Obama")

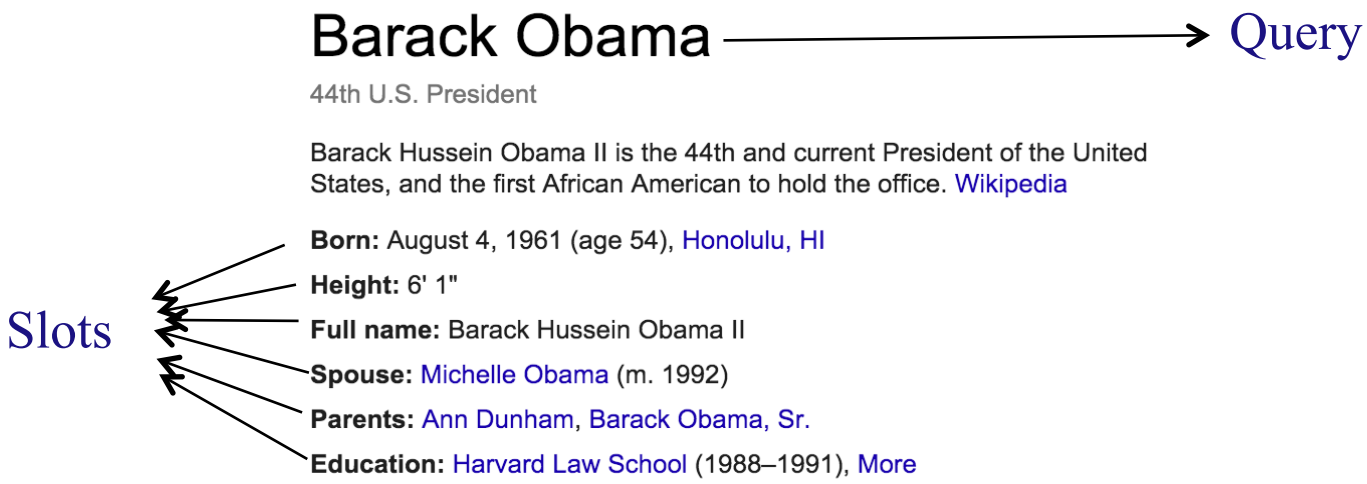
Information Extraction (IE)

- Goals:
  - Organize information so that it is useful to people
  - Organize information in a semantically precise form to allow further inference by downstream applications
- For example: Google's knowledge graph



Knowledge Base Population (KBP)

- Knowledge Base(KB) is a collection of information that follows an ontology (schema).
  - For example: DBPedia and FreeBase
- KBP is the task of taking an incomplete KB, and a large corpus of text, and completing the incomplete elements of the KB.
  - For example: Wikipedia Infoboxes



Information Extraction Performance Metrics

- Precision - ratio of correct slot fills to total number of slot fills provided by the system.
  - Precision = $\frac{\text{\# correct slot fills}}{\text{\# system's slot fills}}$
- Recall - ratio of correct slot fills to total number of correct slot fills.
  - Recall = $\frac{\text{\# correct slot fills}}{\text{\# total correct slot fills}}$
- Inverse relation between precision and recall.
- F1 – harmonic mean of precision and recall
  - F1 = $\frac{2 \text{ x precision x recall}}{\text{precision + recall}}$

Knowledge-Base Population (KBP)

- Annual evaluation of relation extraction from natural language documents organized by NIST.
- English Slot Filling (ESF) task:

per: Barack Obama
country_of_birth United States
spouse Michelle Obama
children Malia Obama Sasha Obama

org: Microsoft
city_of_headquarters Redmond
website microsoft.com
subsidiaries Skype Nokia

KBP Provenance

- Systems must provide information on where the evidence for each slot fill is in the document corpus.
- Given by:
  - Doc ID
  - Start Offset
  - End Offset

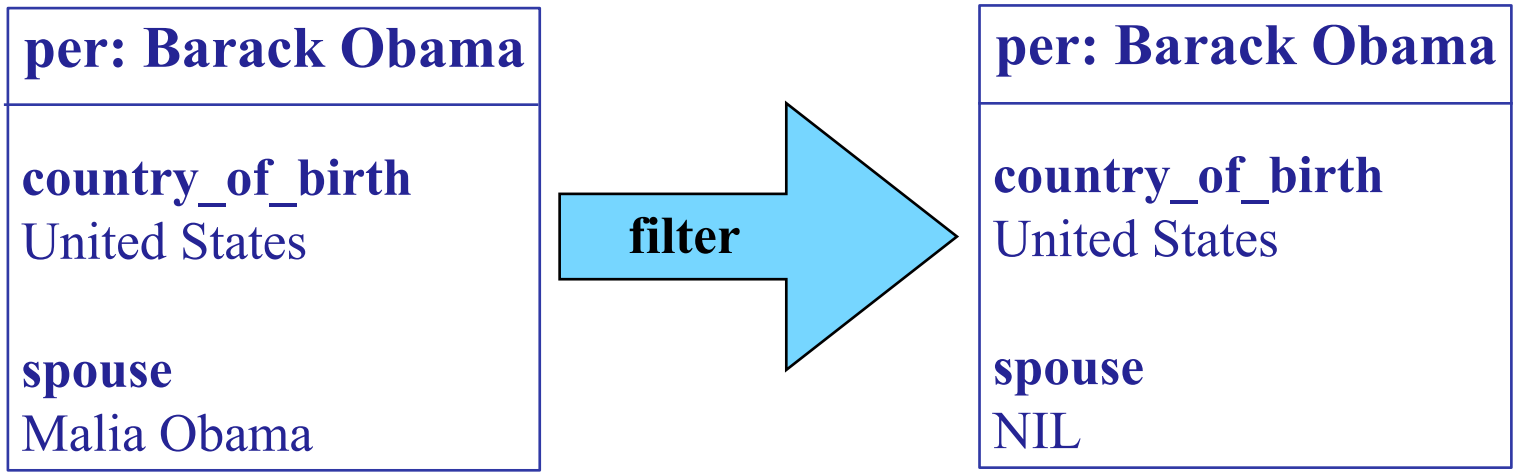
org: Microsoft

<eng-NG-31-1007> : Microsoft is a technology company headquartered in Redmond, Washington, that develops …

city_of_headquarters
Redmond
Doc ID
eng-NG-31-1007
Start Offset
48
End Offset
54

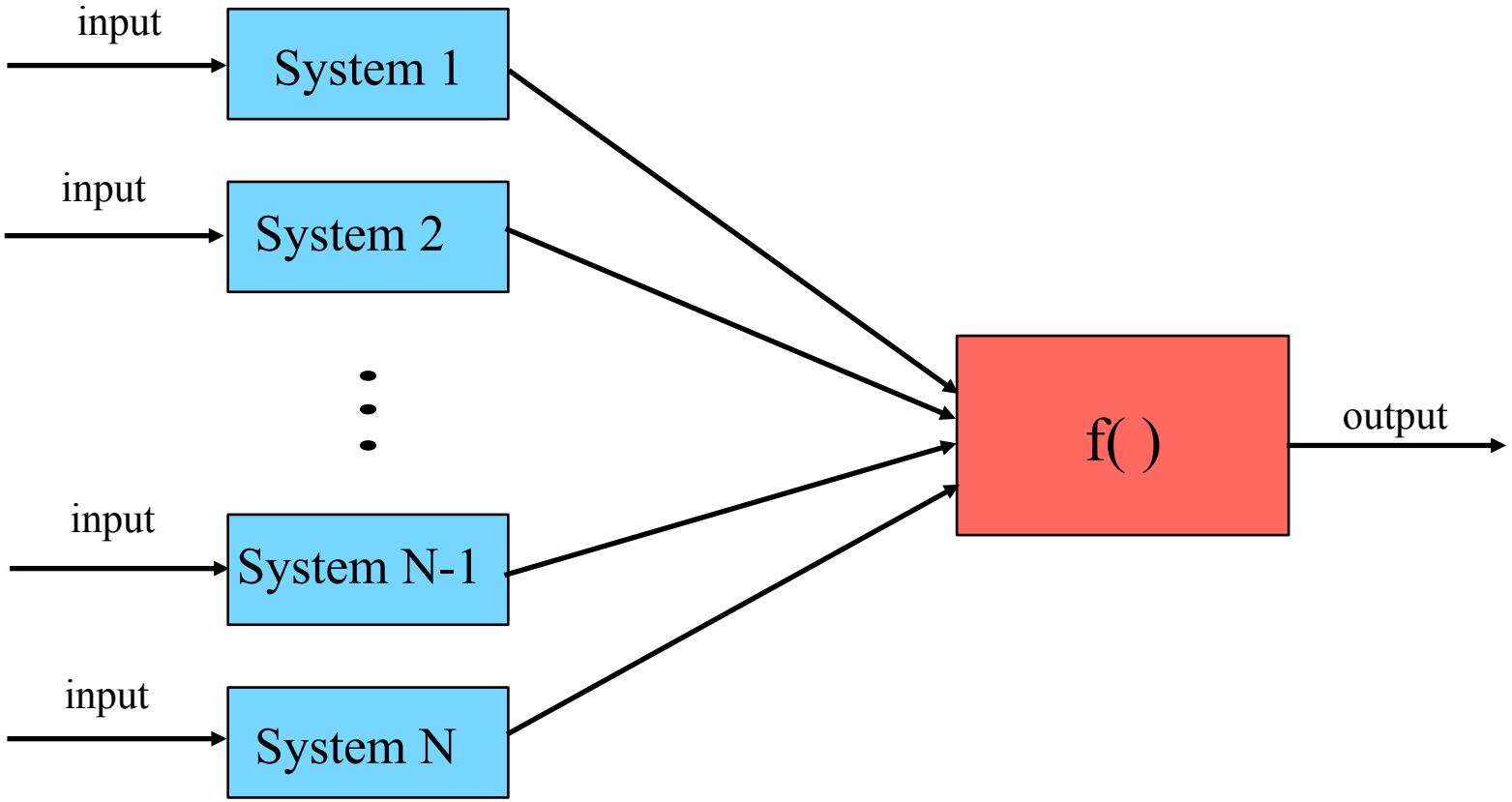
KBP Slot Filler Validation

- Aim: Improve precision of individual systems.
- Input is system outputs from the ESF task.
- Output is filtered slot fills.



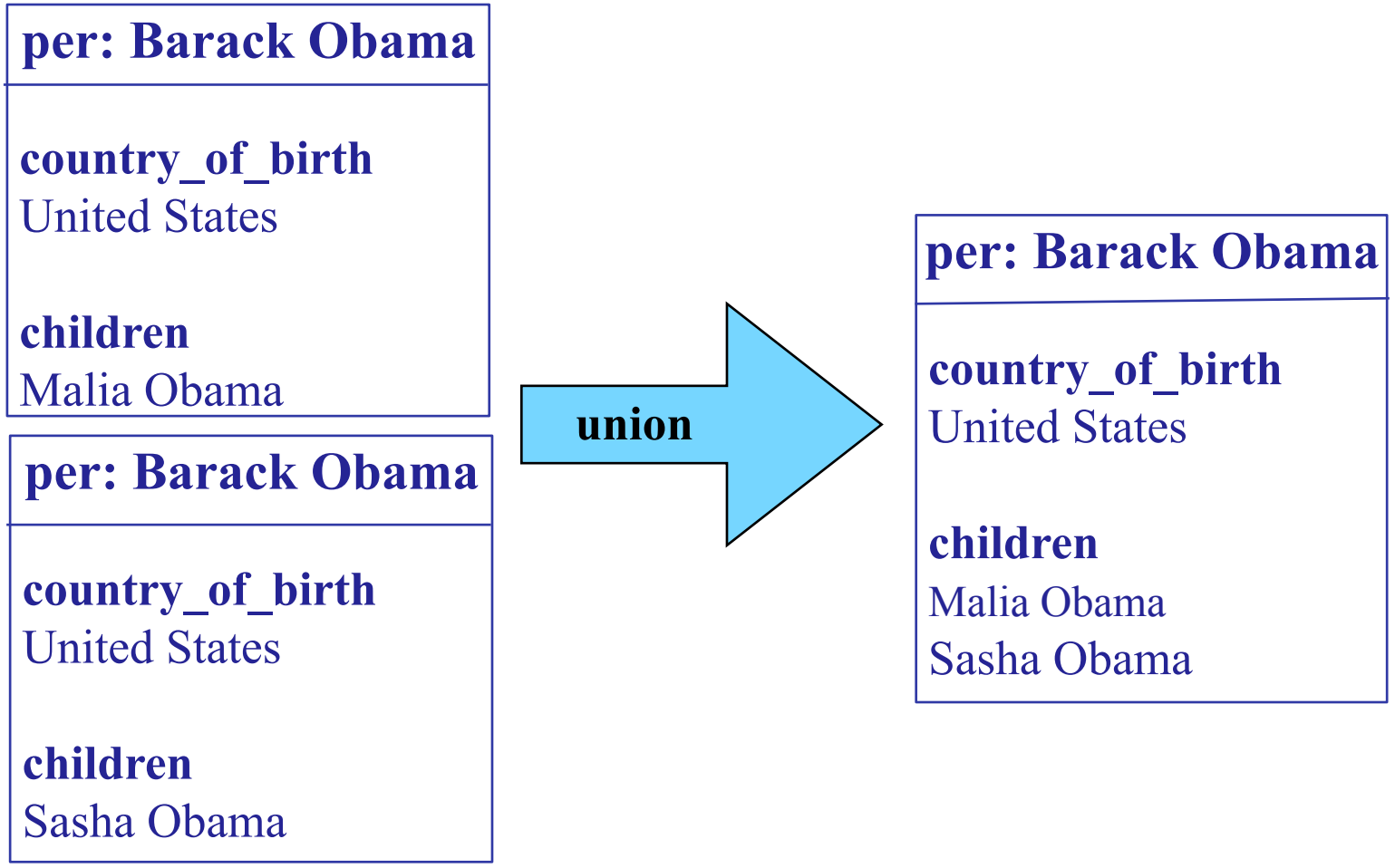
Ensembling

- Netflix $1M prize winning team's algorithm used ensembling.



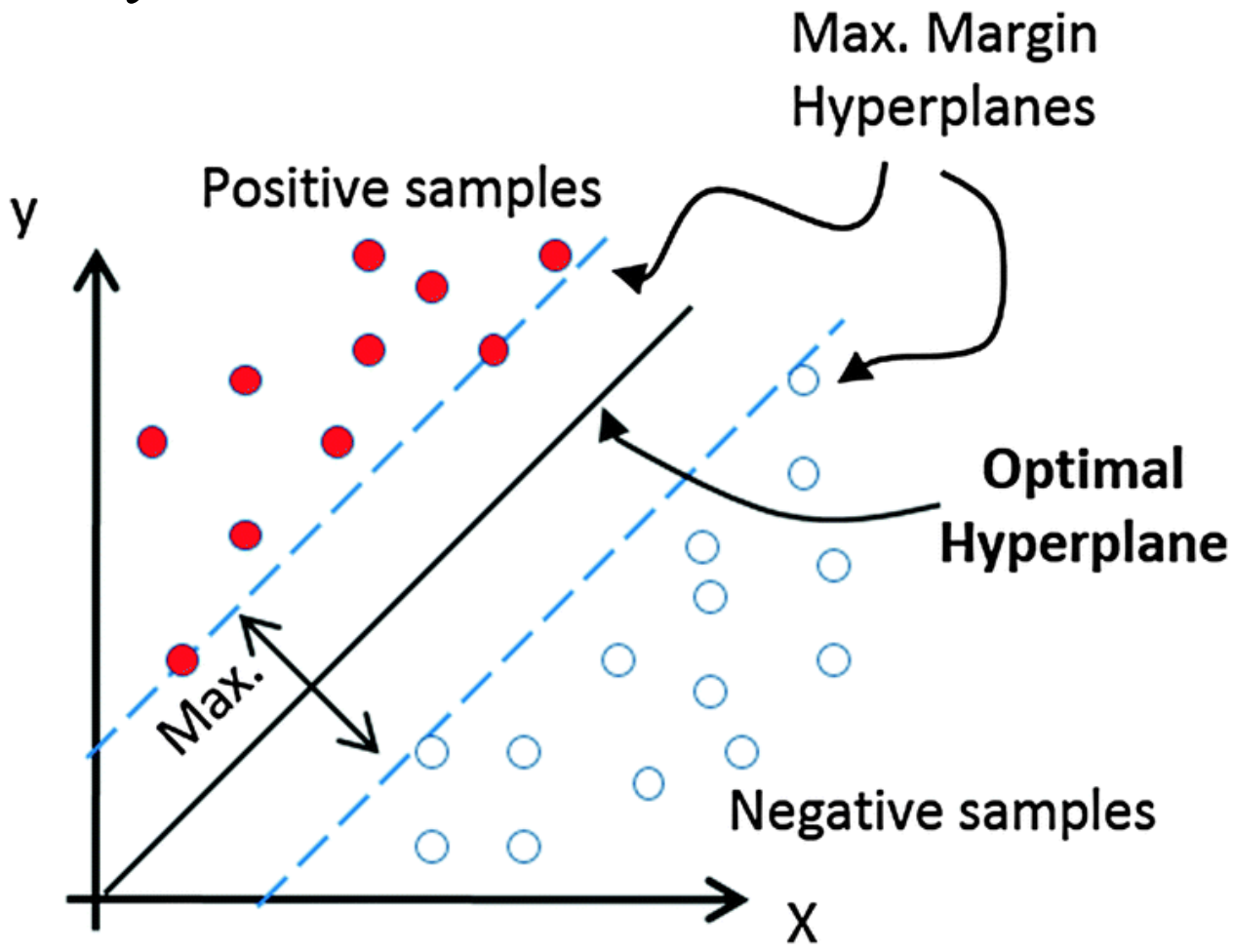
KBP Slot Filler Validation

- Ensembling used to improve recall as well



Support Vector Machine (SVM)

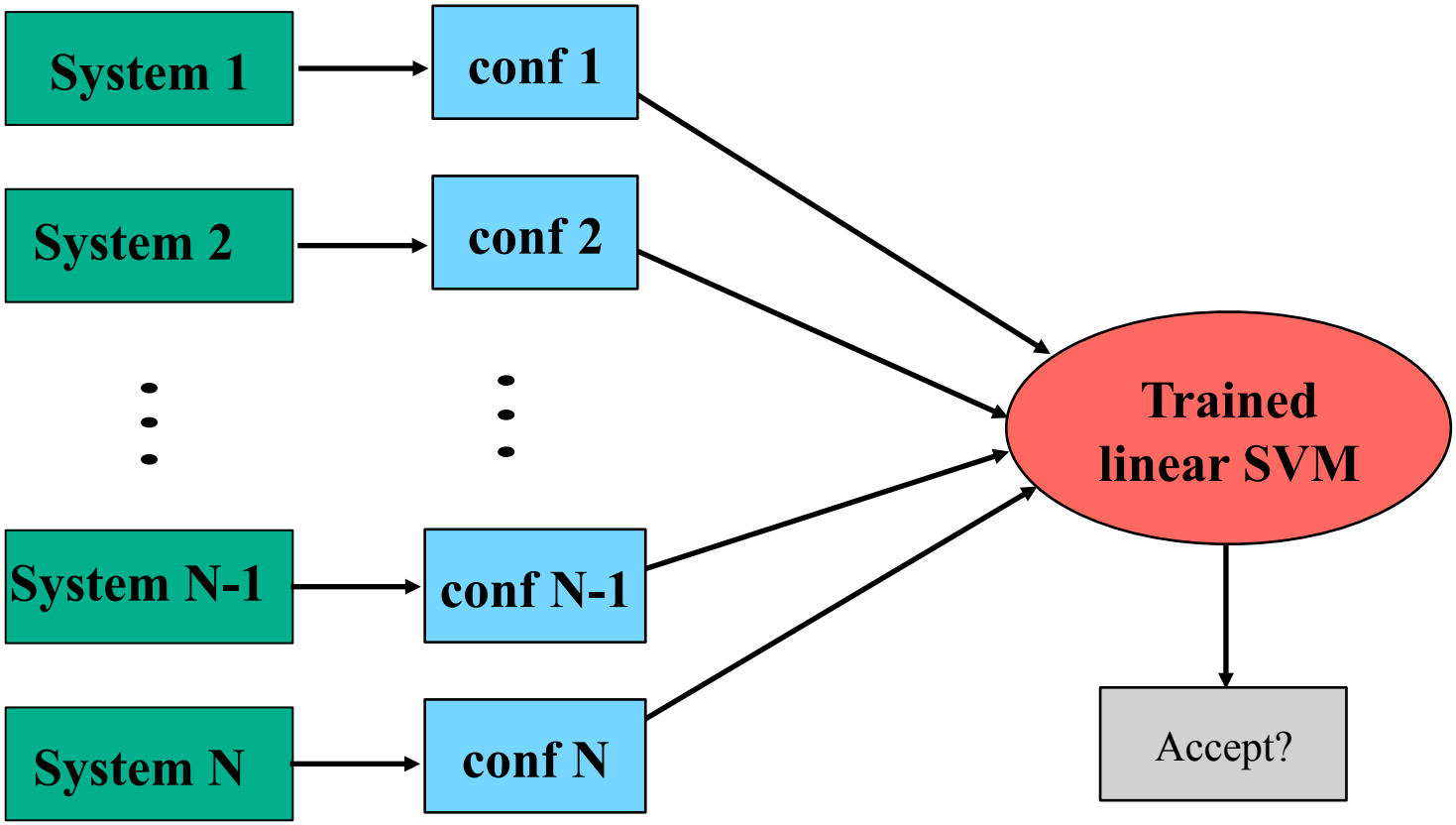
- Binary class classification model



Stacking

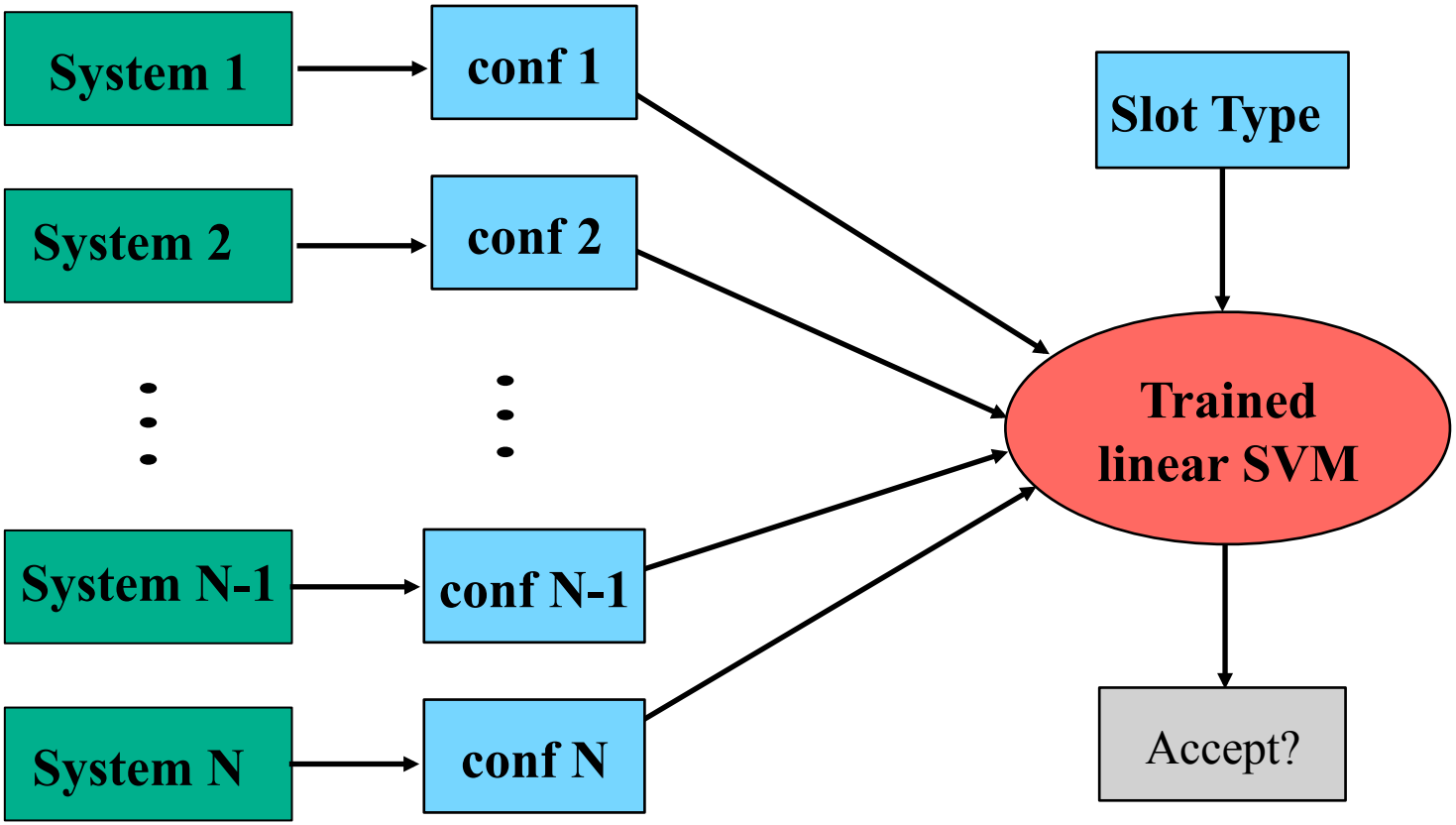
(Wolpert, 1992)

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



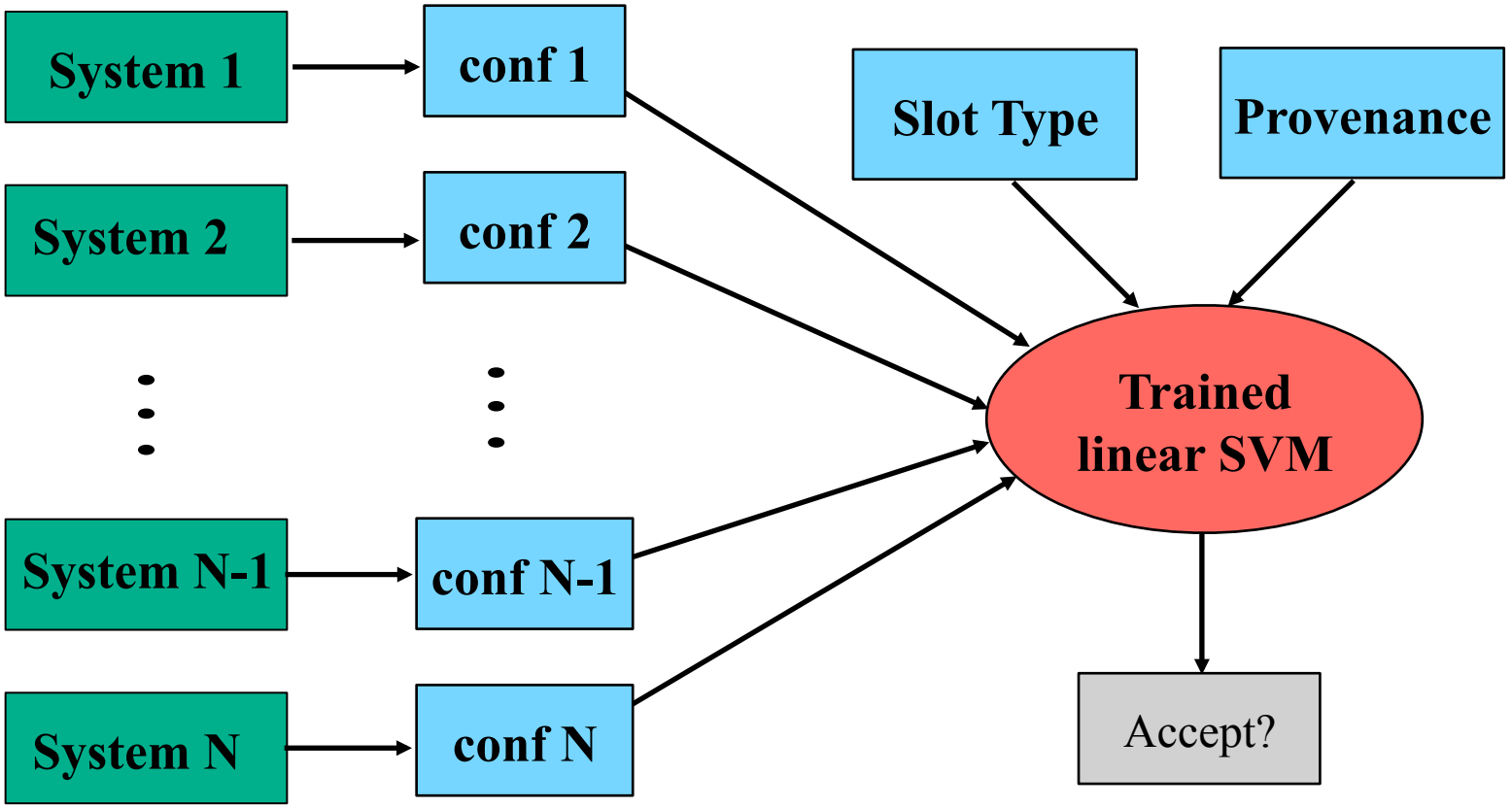
Stacking with Features

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

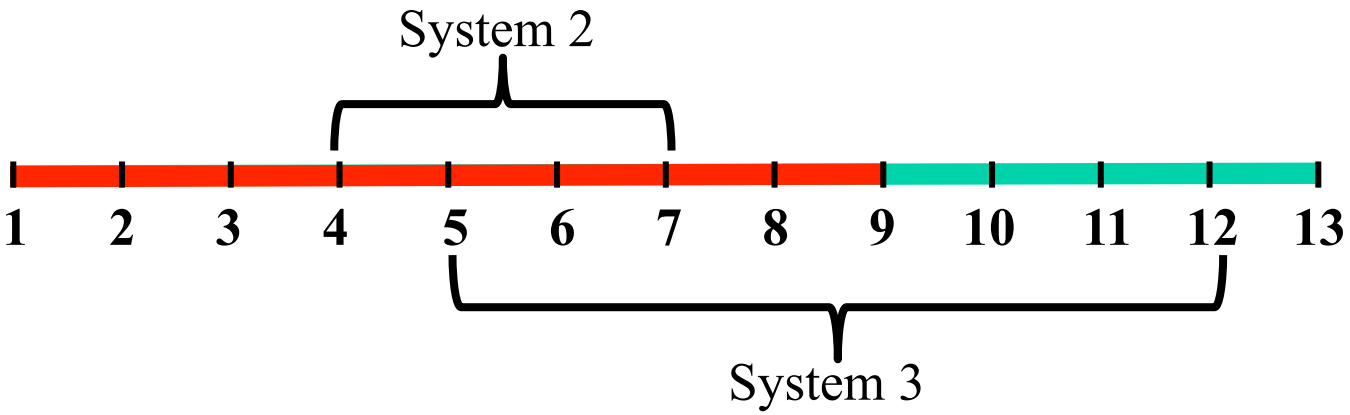
- Degree of overlap between systems' provenance strings (prov).
- Uses Jaccard similarity coefficient.
- For a given query and slot, for each system, *i,* there is a feature OP_i :
  - *N* systems provide a fill with same *docid*
  - Offset provenance for a system *i* is calculated as:

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

  - Systems with different *docid* have zero OP

Offset Provenance Score

Offsets	System 1	System 2	System 3
Start Offset	1	4	5
End Offset	9	7	12



$$OP_1 = \frac{1}{2} \times \left(\frac{4}{9} + \frac{5}{12} \right)$$

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 → confidence scores → #systems
- Document provenance feature → #systems
- Offset provenance feature → #systems
- Document similarity feature → #systems
- Slot type → 40 (person + organization)
- #systems = 38 in 2015 and 10 in 2014

Datasets for 2014

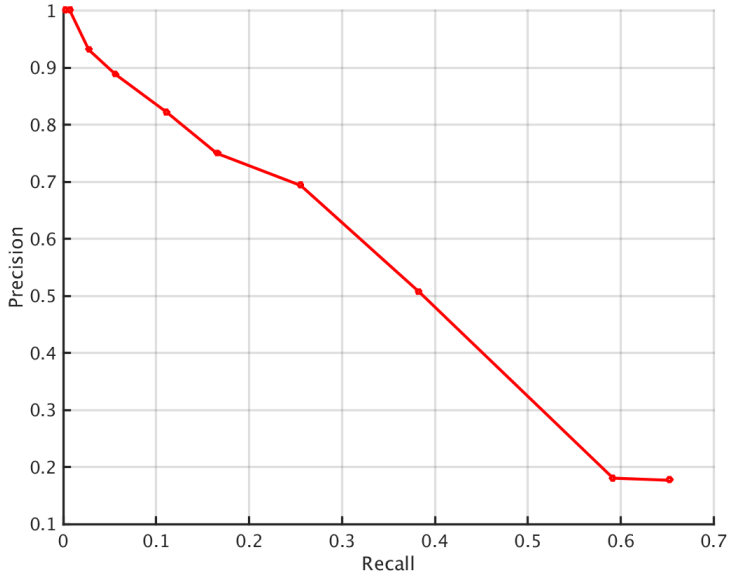
- 2014 Slot Filler Validation (SFV) data
  - 17 teams
  - 65 systems
- Ten Common Systems that participated both in 2013 and 2014 English Slot Filling (ESF) task:
  - LSV
  - IIRG
  - UMASS_IESL
  - Stanford
  - BUPT_PRIS
  - RPI_BLENDER
  - CMUML
  - NYU
  - Compreno
  - UWashington

Baselines

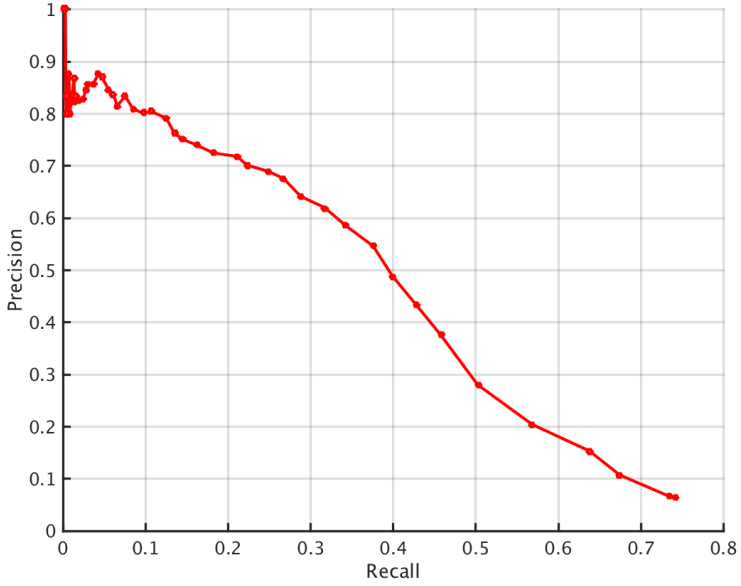
- Union
  - Combine systems for maximizing recall
  - List valued slot fills => always included
  - Single valued slot fills => highest confidence

Baselines

- Voting
  - Combine systems for maximizing F1
  - Vary threshold on #systems that must agree
  - Learn threshold on 2013 data



Common Systems Dataset (3)



SFV Dataset (10)

KBP English Slot Filling (ESF) Results

2014 Slot Filler Validation (SFV) Data

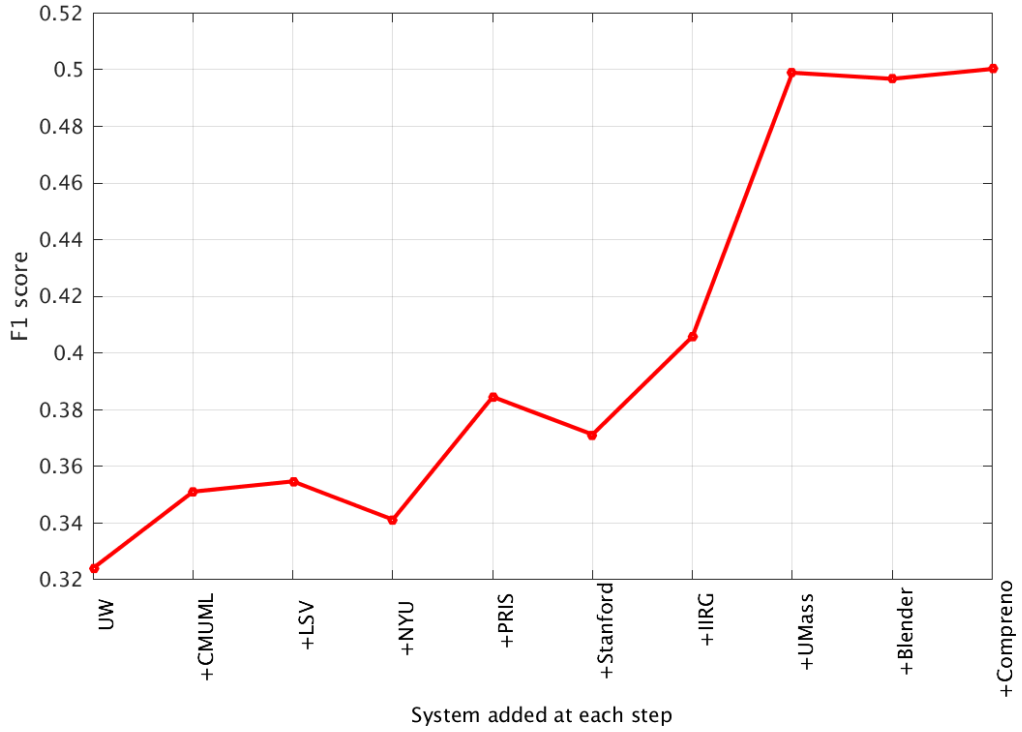
Baseline	Precision	Recall	F1
Union	0.067	**0.762**	0.122
Voting	**0.641**	0.288	**0.397**

Common systems for 2013 and 2014 ESF task

Approach	Precision	Recall	F1
Union	0.176	**0.647**	0.277
Voting	**0.694**	0.256	0.374
Best ESF system in 2014 (Stanford)	0.585	0.298	0.395
Stacking	0.606	0.402	0.483
Stacking + Slot Type	0.607	0.406	0.486
Stacking + Provenance + Slot Type	0.541	0.466	**0.501**

Incremental Training on Systems

- Sort the common systems based on their performance.
- Train the classifier adding one system at each step.
- Test on 2014 data.



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.

Constraint Optimization for Single-Valued Slots

- Consider a single-value slot for a given query with possible values $E_1, E_2, \ldots, E_k$:

$$P(E_1) + P(E_2) + \cdots + P(E_k) \leq 1.$$

  - Because each E_i is mutually exclusive.

$$\min_{0 \leq x_i \leq 1} \sum_{i=1}^{M} \sum_{j=1}^{N_i} w_{ij} \left(x_i - c_i(j)\right)^2$$

  - $w_{i,j}$ are weights equal to the inverse of the rank of the system based on precision, learnt from the previous year.
  - $w_{i,j}$ is chosen to be uniform for new systems.

Constraint Optimization for Single-Valued Slots

- 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

  - $w_{1,1} = 1/7,\ w_{2,1} = 1/7,\ w_{2,2} = 1/7$
  - $\Rightarrow x_1 = 0.36823,\ x_2 = 0.63177$

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

(per:country_of_birth, Harvey Milk, United States)

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

Constraint Optimization for List-Valued Slots

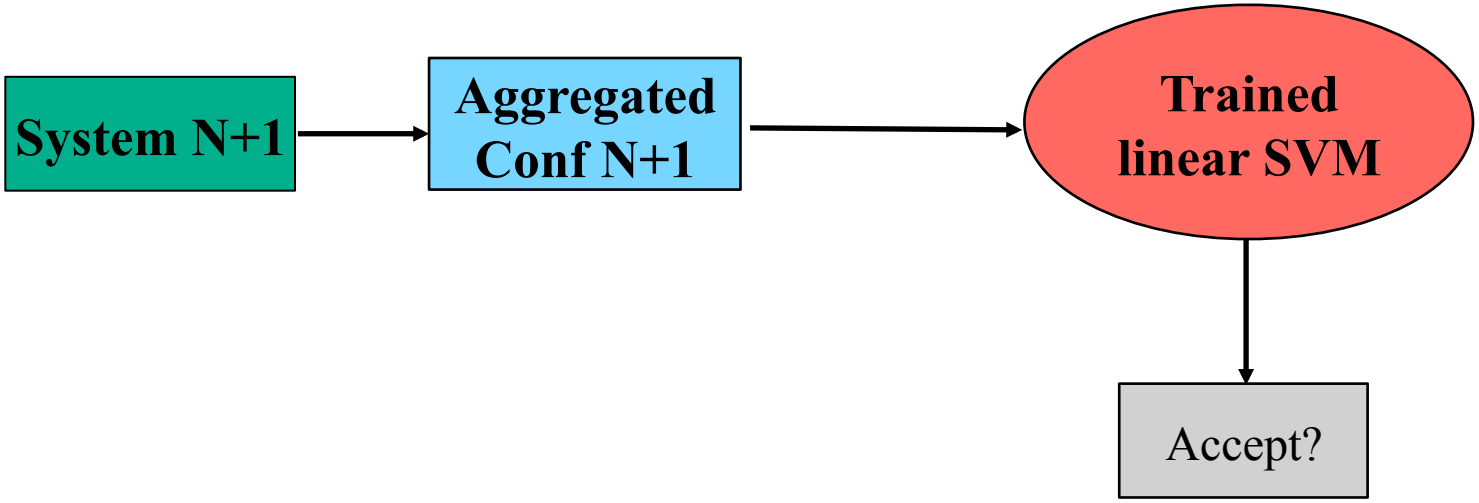
- Consider a list-value slot for a given query with possible values E_1,E_2,…,E_k:

$$P(E_1) + P(E_2) + \cdots + P(E_k) \leq Avg(\frac{n_c}{n}) \times k$$

  - For a given list-value slot type:
    - n_c is the number of correct slot fills in 2014
    - n is the total number of slot fills in 2014
    - Average it over all query entities in 2014
- The "collective precision" is only a rough estimate.
- Under-estimate or over-estimate may lead to poor recall or precision respectively.

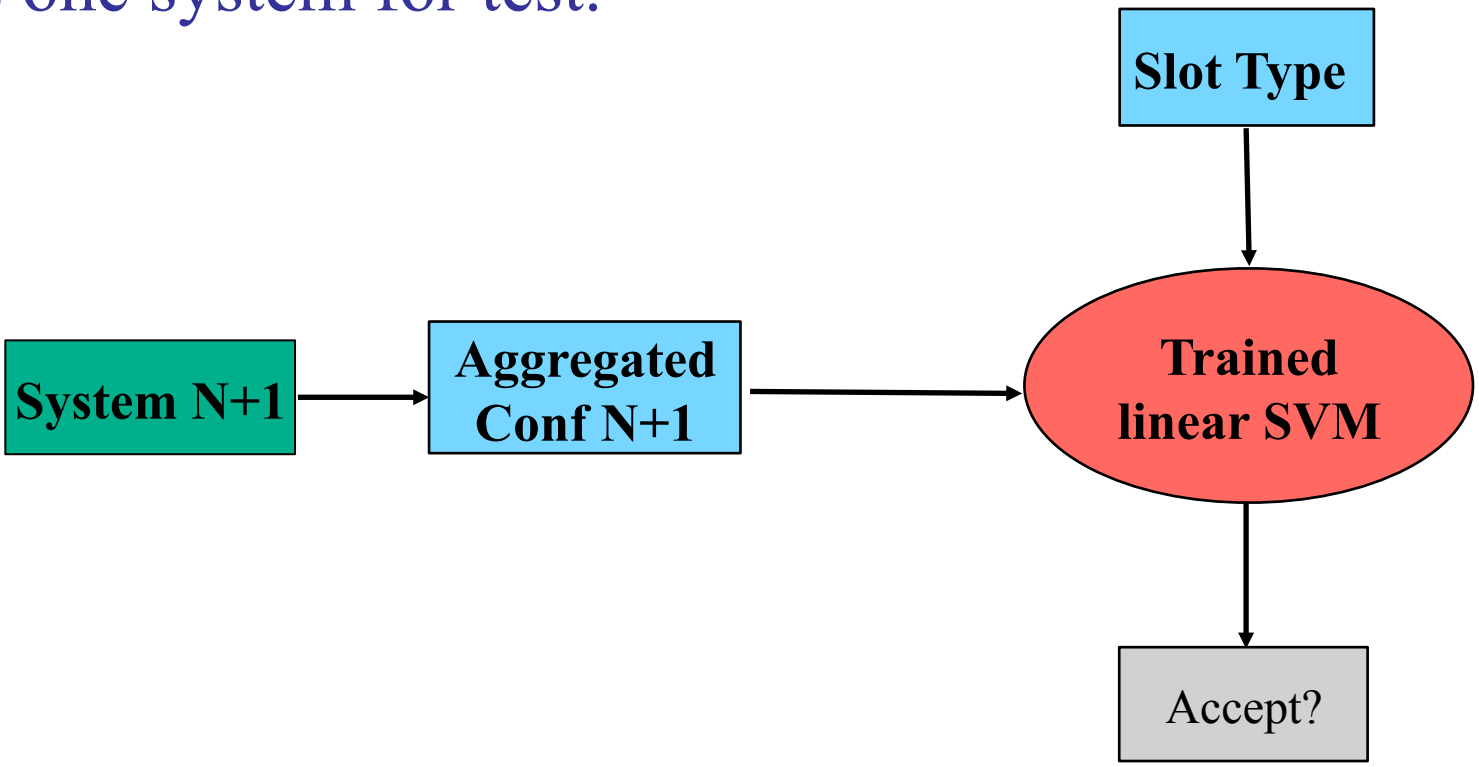
Stacking over the Unsupervised Approach

- 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.



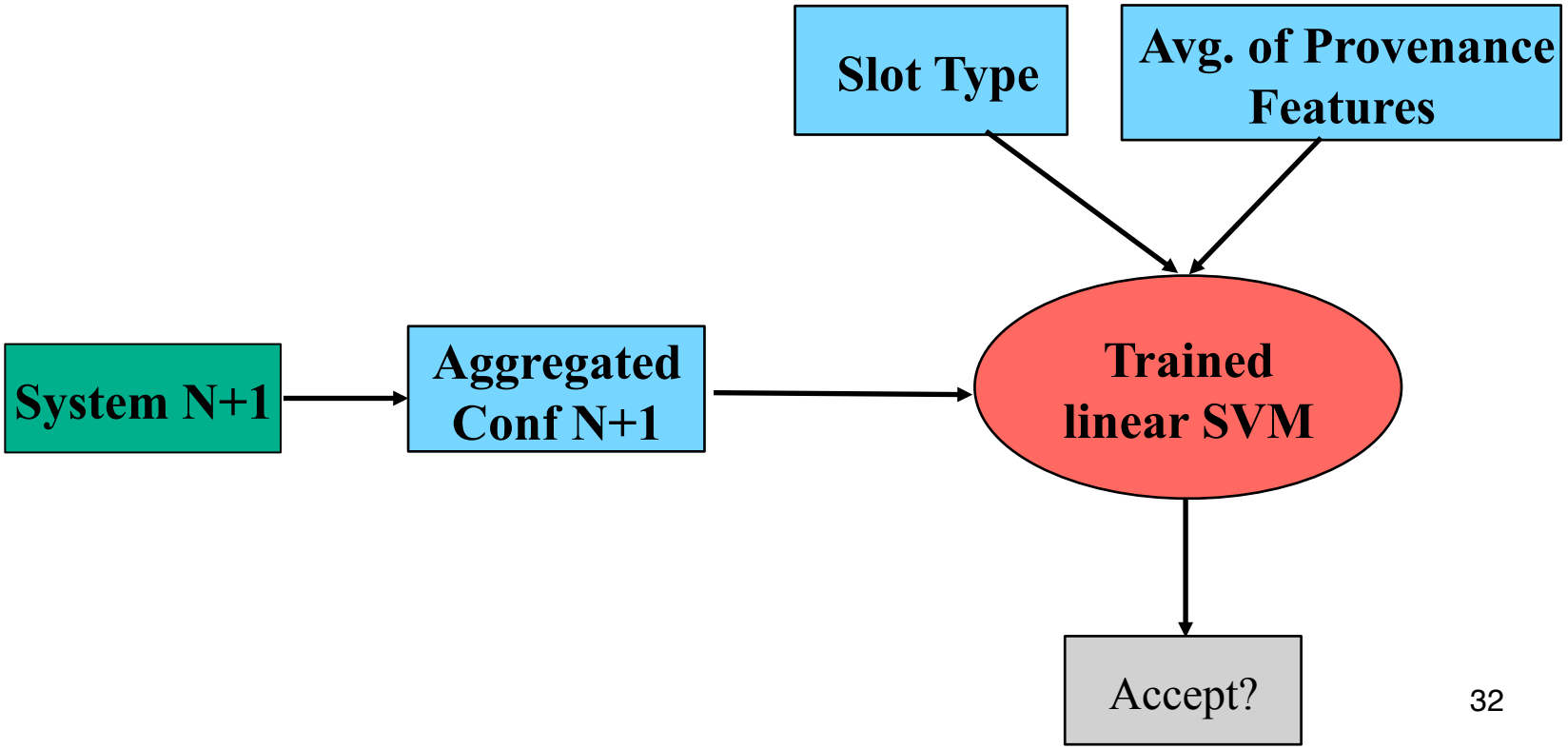
Stacking over the Unsupervised Approach

- 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)

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**

Conclusion

- Stacked meta-classifier beats the best performing 2014 KBP ESF system by an F1 gain of **11** points.
- Features that utilize provenance information improve stacking performance.
- Ensembling has clear advantages but naive approaches such as voting do not perform as well.

Conclusion

- 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.

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.
- Image on slide 11 taken from pubs.rsc.org

Thank You