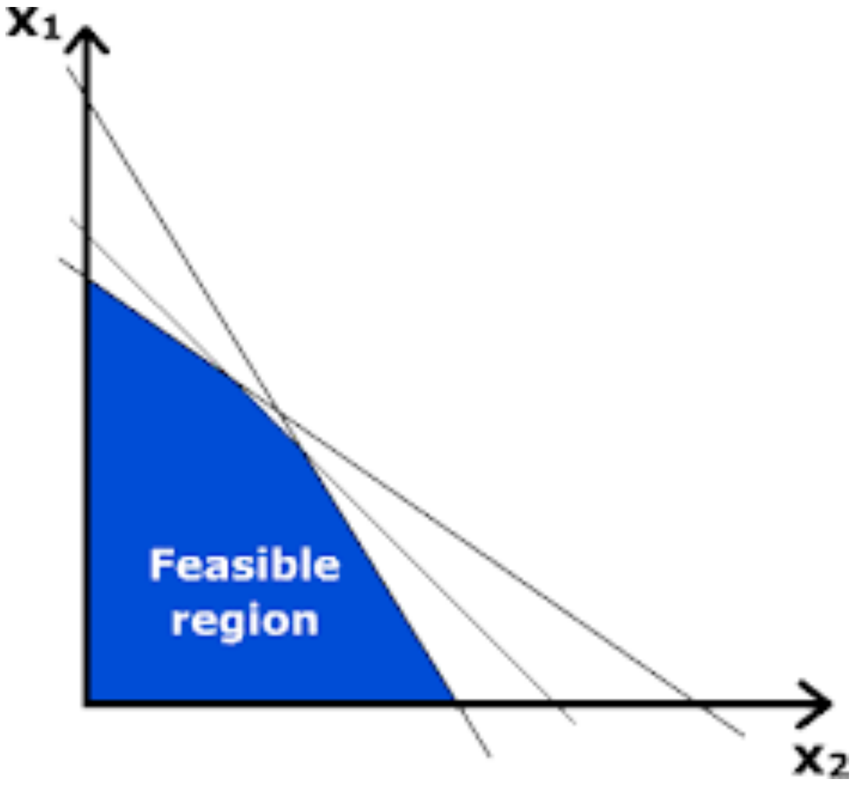
A Linear Programming Formulation for Global Inference in Natural Language Tasks



Dan Roth **Wen-tau Yih**
Department of Computer Science
University of Illinois at Urbana-Champaign
{danr, yih}@uiuc.edu

Proceedings of the Eighth conference on Computational Natural Language Learning (CoNLL-2004) at HLT-NAACL 2004

A presentation by Aditya M. Kashyap

*Some of the slides were adapted from a tutorial given by Prof. Dan Roth at EACL 2017

Introduction

- **Task:** Identify Entities and Relations
- **Sentence:** “Tom married Mary in England”
- **Entity Extraction:**

Introduction

- **Task:** Identify Entities and Relations

- **Sentence:** "$\overset{\mathbb{E}_1}{\text{Tom}}$ married $\overset{\mathbb{E}_2}{\text{Mary}}$ in $\overset{\mathbb{E}_3}{\text{England}}$"

- **Entity Extraction:**
  - (Tom, *person*)
  - (Mary, *person*)
  - (England, *location*)

Introduction

- **Task:** Identify Entities and Relations

$\mathbb{E}_1$ $\mathbb{E}_2$ $\mathbb{E}_3$

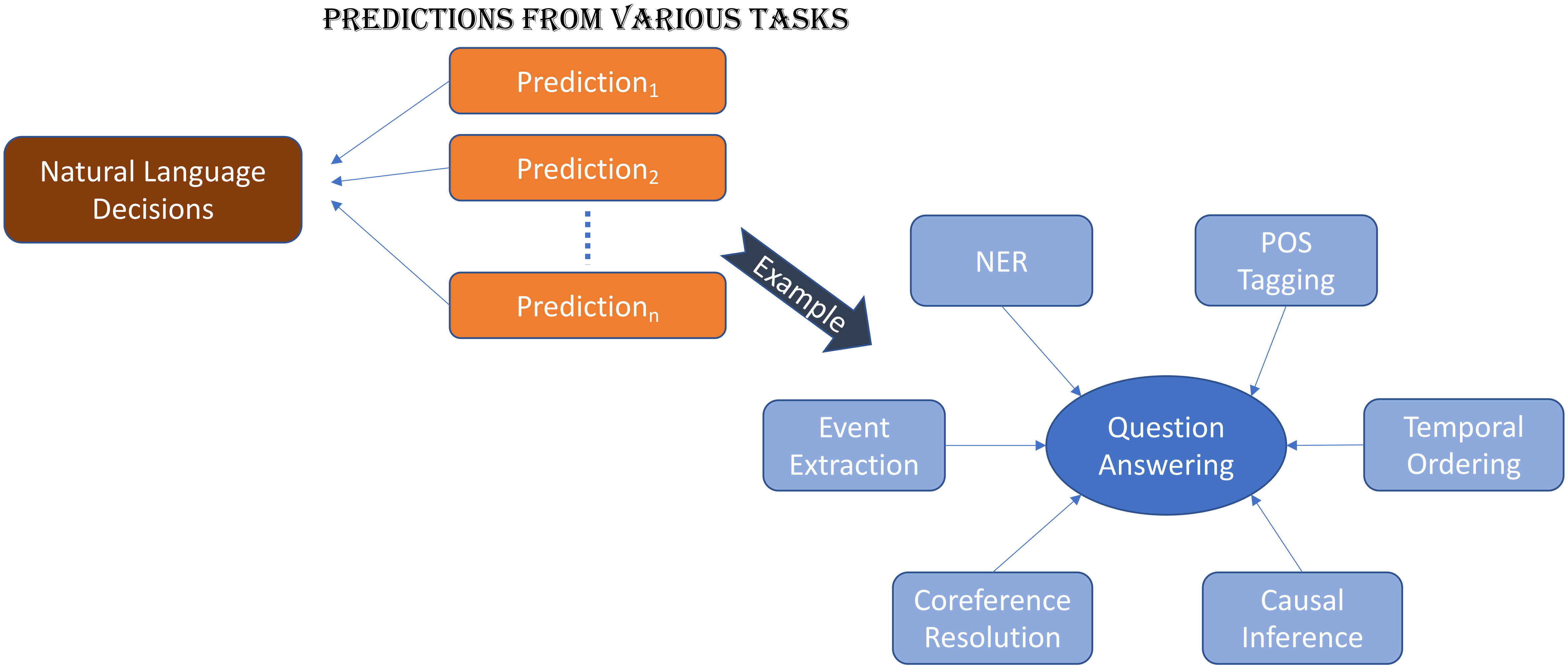
- **Sentence:** "Tom married Mary in England"

$\mathbb{R}_{12}$

- **Entity Extraction:**
  - (Tom, *person*)
  - (Mary, *person*)
  - (England, *location*)
- **Relation Extraction:**
  - *Marry*(Tom, Mary)

The relation tag "marry" is **constrained** by the two entity labels for "Tom" and "Mary"

Introduction



Introduction

- These predictions must typically respect some constraints
  - Part of Speech (POS) Tagging:
    - Sentence must have at least 1 verb
    - Cannot have 3 consecutive verbs
  - Name Entity Resolution
    - No two entities can overlap

POS Tagging

NER

- Efficient solutions for these type of problems have been given when the constraints are sequential.
- These solutions can be categorized into two different frameworks
  - **Learning Global Models**: Ex. Variations of HMMs, conditional models, etc.
  - **Inference with Classifiers**

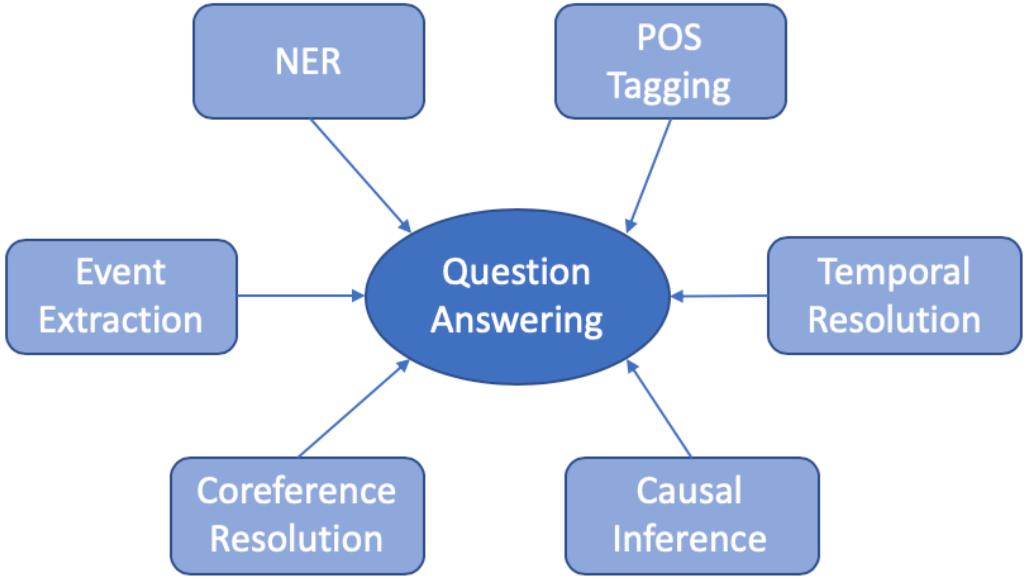
We are going to focus on this

Introduction

- Typically, both these frameworks rely on dynamic programming → works well with sequential data
- Many problems → structure is more general → computationally intractable inference
- This paper develops a novel *inference with classifiers* approach
  - Studies a general setting → does not restrict to sequential data
- The problem is formulated as:
  - Collection of discrete random variables
  - Binary relations
  - Constraints on the binary relations

Introduction

- Can contrast model in this approach to other sequential inference methods.
- However, a key difference:
  - **Other approaches**: Model is learned globally, under constraints imposed by the domain
  - **The paper's approach**:
    - Predictors don't need to be learned in the context of decision tasks
      - It is related to the notion of the ability to decouple the learning (or some of it) from the final global decision
    - Push the global decision to minimally violate constraints



Entity and Relation Recognition

- The model first learns a collection of “local” predictors:
  - Entity Identifier
  - Relation Identifier
- A global decision is produced that optimizes over:
  - ✓ the suggestions of the classifiers
  - ✓ Known constraints among them
  - ✓ Domain or task specific constraints
- Brute force:
  - *n* entities in a sentence → $O(n^2)$ possible relations
  - If each variable (entity or relation) can take l values, l^a assignments, where $a = n^2$

Entity and Relation Recognition

- While evaluated on simultaneous learning of named entities and relations, this papers approach:
  - Provides a significant improvement in the predictor’s accuracy
  - Provides *coherent* solutions
- Coherent solutions: No inconsistencies among predictions (“stupid mistakes”)

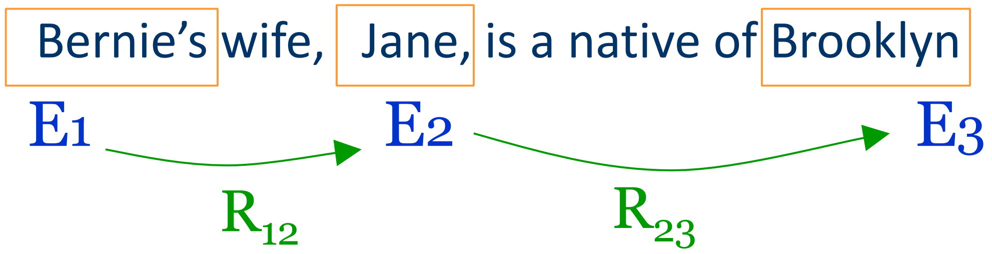
Joint Inference with General Constraint Structure [Roth&Yih'04,07,….]

Recognizing Entities and Relations

other	0.05
per	0.85
loc	0.10

other	0.10
per	0.60
loc	0.30

other	0.05
per	0.50
loc	0.45



irrelevant	0.05
spouse_of	0.45
born_in	0.50

irrelevant	0.10
spouse_of	0.05
born_in	0.85

Joint Inference with General Constraint Structure [Roth&Yih'04,07,….]

Recognizing Entities and Relations

other	0.05
per	**0.85**
loc	0.10

other	0.10
per	**0.60**
loc	0.30

other	0.05
per	**0.50**
loc	0.45

Bernie's wife, Jane, is a native of Brooklyn

$E_1 \xrightarrow{R_{12}} E_2 \xrightarrow{R_{23}} E_3$

irrelevant	0.05
spouse_of	0.45
born_in	**0.50**

irrelevant	0.10
spouse_of	0.05
born_in	**0.85**

Joint Inference with General Constraint Structure [Roth&Yih'04,07,….]

Recognizing Entities and Relations

other	0.05
per	**0.85**
loc	0.10

other	0.10
per	**0.60**
loc	0.30

other	0.05
per	0.50
loc	**0.45**

Bernie's wife, Jane, is a native of Brooklyn

$E_1 \xrightarrow{R_{12}} E_2 \xrightarrow{R_{23}} E_3$

irrelevant	0.05
spouse_of	0.45
born_in	**0.50**

irrelevant	0.10
spouse_of	0.05
born_in	**0.85**

Joint Inference with General Constraint Structure [Roth&Yih'04,07,….]

Recognizing Entities and Relations

other	0.05
per	**0.85**
loc	0.10

other	0.10
per	**0.60**
loc	0.30

other	0.05
per	0.50
loc	**0.45**

$\boxed{\text{Bernie's}}$ wife, $\boxed{\text{Jane,}}$ is a native of $\boxed{\text{Brooklyn}}$

$E_1 \xrightarrow{R_{12}} E_2 \xrightarrow{R_{23}} E_3$

irrelevant	0.05
spouse_of	0.45
born_in	**0.50**

irrelevant	0.10
spouse_of	0.05
born_in	**0.85**

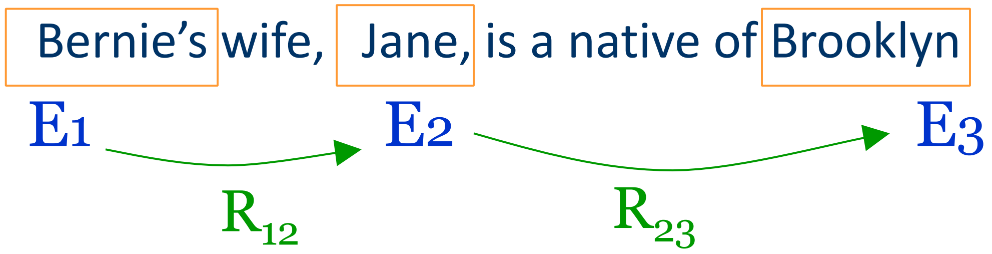
Joint Inference with General Constraint Structure [Roth&Yih'04,07,….]

Recognizing Entities and Relations

other	0.05
per	**0.85**
loc	0.10

other	0.10
per	**0.60**
loc	0.30

other	0.05
per	0.50
loc	**0.45**



irrelevant	0.05
spouse_of	**0.45**
born_in	0.50

irrelevant	0.10
spouse_of	0.05
born_in	**0.85**

Joint Inference with General Constraint Structure [Roth&Yih'04,07,….]

Recognizing Entities and Relations

Joint inference gives good improvement

other	0.05
per	**0.85**
loc	0.10

other	0.10
per	**0.60**
loc	0.30

other	0.05
per	0.50
loc	

Key Questions:

How to learn the model(s)?

What is the source of the knowledge?

Bernie's wife, Jane, is a native of Brooklyn

$E_1 \xrightarrow{R_{12}} E_2 \xrightarrow{R_{23}} E_3$

irrelevant	0.05
spouse_of	**0.45**
born_in	0.50

irrelevant	0.10
spouse_of	0.05
born_in	**0.85**

Models could be learned separately/jointly; constraints may come up only at decision time.

Joint Inference with General Constraint Structure [Roth&Yih'04,07,....]

Recognizing Entities and Relations

Joint inference gives good improvement

An Objective function that incorporates learned models with knowledge **(output constraints)**

A Constrained Conditional Model

	0.05

other	0.10

other	0.05
per	

yn

E_1 E_2 E_3

R_{12} R_{23}

Key Questions:
How to learn the model(s)?
What is the source of the knowledge?

irrelevant	0.05
spouse_of	**0.45**
born_in	0.50

irrelevant	0.10
spouse_of	0.05
born_in	**0.85**

Models could be learned separately/jointly; constraints may come up only at decision time.

The Relational Inference Problem

- Under weak assumptions, we can view the inference problem as an optimization problem, which aims to minimize the sum of the following:
  - **Assignment cost:** The cost of deviating from the assignment $\mathcal{V}$ given by the classifiers.
    - Let *l* is the label assigned to variable *u* with a probabilit $p = P(f_u = l)$
    - The assignment cost is given by $c_u(l)$ = $-\log p$
  - **Constraint cost:** The cost imposed by breaking constraints between neighboring nodes.
    - $d^1(f_{E_i}, f_{R_{ij}})$ = 0 if $(f_{R_{ij}}, f_{E_i}) \in \mathcal{C}^1$, otherwise $d^1(f_{E_i}, f_{R_{ij}})$ = ∞
    - Similarly, d^2 is used to force consistency of the second argument of a relation

The Relational Inference Problem

- The overall cost function optimized, for a global labeling *f* of all variables is:

$$\begin{aligned} C(f) &= \overbrace{\sum_{u\in\mathcal{V}} c_u(f_u)}^{\textbf{Assignment Cost}} \\ &+ \underbrace{\sum_{R_{ij}\in\mathcal{R}} \left[d^1(f_{R_{ij}}, f_{E_i}) + d^2(f_{R_{ij}}, f_{E_j})\right]}_{\textbf{Constraint Cost}} \end{aligned}$$

This combinatorial problem is computationally intractable

A Computational Approach to Relational Inference

$$\min \sum_{E\in\mathcal{E}}\sum_{e\in\mathcal{L}_\mathcal{E}} c_E(e)\cdot x_{\{E,e\}} + \sum_{R\in\mathcal{R}}\sum_{r\in\mathcal{L}_\mathcal{R}} c_R(r)\cdot x_{\{R,r\}}$$
$$+ \sum_{\substack{E_i,E_j\in\mathcal{E}\\ E_i\neq E_j}}\left[\sum_{r\in\mathcal{L}_\mathcal{R}}\sum_{e_1\in\mathcal{L}_\mathcal{E}} d^1(r,e_1)\cdot x_{\{R_{ij},r,E_i,e_1\}} + \sum_{r\in\mathcal{L}_\mathcal{R}}\sum_{e_2\in\mathcal{L}_\mathcal{E}} d^2(r,e_2)\cdot x_{\{R_{ij},r,E_j,e_2\}}\right]$$

subject to:

$$\sum_{e\in\mathcal{L}_\mathcal{E}} x_{\{E,e\}} = 1 \qquad \forall E\in\mathcal{E}$$
$$\sum_{r\in\mathcal{L}_\mathcal{R}} x_{\{R,r\}} = 1 \qquad \forall R\in\mathcal{R}$$
$$x_{\{E,e\}} = \sum_{r\in\mathcal{L}_\mathcal{R}} x_{\{R,r,E,e\}} \qquad \forall E\in\mathcal{E} \text{ and } \forall R\in\{R: E=\mathcal{N}^1(R) \text{ or } R: E=\mathcal{N}^2(R)\}$$
$$x_{\{R,r\}} = \sum_{e\in\mathcal{L}_\mathcal{E}} x_{\{R,r,E,e\}} \qquad \forall R\in\mathcal{R} \text{ and } \forall E=\mathcal{N}^1(R) \text{ or } E=\mathcal{N}^2(R)$$
$$x_{\{E,e\}} \in \{0,1\} \qquad \forall E\in\mathcal{E}, e\in\mathcal{L}_\mathcal{E}$$
$$x_{\{R,r\}} \in \{0,1\} \qquad \forall R\in\mathcal{R}, r\in\mathcal{L}_\mathcal{R}$$
$$x_{\{R,r,E,e\}} \in \{0,1\} \qquad \forall R\in\mathcal{R}, r\in\mathcal{L}_\mathcal{R},\; E\in\mathcal{E}, e\in\mathcal{L}_\mathcal{E}$$

A Computational Approach to Relational Inference

Main Takeaway:

Increases coherent solutions and decreases inconsistent predictions

$$\min \sum_{E \in \mathcal{E}} \sum_{e \in \mathcal{L}_{\mathcal{E}}} c_E(e) \cdot x_{\{E,e\}} + \sum_{R \in \mathcal{R}} \sum_{r \in \mathcal{L}_{\mathcal{R}}} c_R(r) \cdot x_{\{R,r\}} + \sum_{\substack{E_i, E_j \in \mathcal{E} \\ E_i \neq E_j}} \left[\sum_{r \in \mathcal{L}_{\mathcal{R}}} \sum_{e_1 \in \mathcal{L}_{\mathcal{E}}} d^1(r, e_1) \cdot x_{\{R_{ij}, r, E_i, e_1\}} + \sum_{r \in \mathcal{L}_{\mathcal{R}}} \sum_{e_2 \in \mathcal{L}_{\mathcal{E}}} d^2(r, e_2) \cdot x_{\{R_{ij}, r, E_j, e_2\}} \right]$$

subject to:

$$\sum_{e \in \mathcal{L}_{\mathcal{E}}} x_{\{E,e\}} = 1 \qquad \forall E \in \mathcal{E}$$

$$\sum_{r \in \mathcal{L}_{\mathcal{R}}} x_{\{R,r\}} = 1 \qquad \forall R \in \mathcal{R}$$

$$x_{\{E,e\}} = \sum_{r \in \mathcal{L}_{\mathcal{R}}} x_{\{R,r,E,e\}} \qquad \forall E \in \mathcal{E} \text{ and } \forall R \in \{R : E = \mathcal{N}^1(R) \text{ or } R : E = \mathcal{N}^2(R)\}$$

$$x_{\{R,r\}} = \sum_{e \in \mathcal{L}_{\mathcal{E}}} x_{\{R,r,E,e\}} \qquad \forall R \in \mathcal{R} \text{ and } \forall E = \mathcal{N}^1(R) \text{ or } E = \mathcal{N}^2(R)$$

$$x_{\{E,e\}} \in \{0,1\} \qquad \forall E \in \mathcal{E}, e \in \mathcal{L}_{\mathcal{E}}$$

Assignment Cost

Objective Function:

$$\min \sum_{E \in \mathcal{E}} \sum_{e \in \mathcal{L}_{\mathcal{E}}} c_E(e) \cdot x_{\{E,e\}} + \sum_{R \in \mathcal{R}} \sum_{r \in \mathcal{L}_{\mathcal{R}}} c_R(r) \cdot x_{\{R,r\}} + \sum_{\substack{E_i, E_j \in \mathcal{E} \\ E_i \neq E_j}} \left[\sum_{r \in \mathcal{L}_{\mathcal{R}}} \sum_{e_1 \in \mathcal{L}_{\mathcal{E}}} d^1(r, e_1) \cdot x_{\{R_{ij}, r, E_i, e_1\}} + \sum_{r \in \mathcal{L}_{\mathcal{R}}} \sum_{e_2 \in \mathcal{L}_{\mathcal{E}}} d^2(r, e_2) \cdot x_{\{R_{ij}, r, E_j, e_2\}} \right]$$

Constraint Cost

A Computational Approach to Relational Inference

$$\min \sum_{E\in\mathcal{E}}\sum_{e\in\mathcal{L}_\mathcal{E}} c_E(e)\cdot x_{\{E,e\}} + \sum_{R\in\mathcal{R}}\sum_{r\in\mathcal{L}_\mathcal{R}} c_R(r)\cdot x_{\{R,r\}}$$
$$+ \sum_{\substack{E_i,E_j\in\mathcal{E}\\ E_i\neq E_j}} \left[\sum_{r\in\mathcal{L}_\mathcal{R}}\sum_{e_1\in\mathcal{L}_\mathcal{E}} d^1(r,e_1)\cdot x_{\{R_{ij},r,E_i,e_1\}} + \sum_{r\in\mathcal{L}_\mathcal{R}}\sum_{e_2\in\mathcal{L}_\mathcal{E}} d^2(r,e_2)\cdot x_{\{R_{ij},r,E_j,e_2\}}\right]$$

subject to:

$$\sum_{e\in\mathcal{L}_\mathcal{E}} x_{\{E,e\}} = 1 \qquad \forall E\in\mathcal{E}$$
$$\sum_{r\in\mathcal{L}_\mathcal{R}} x_{\{R,r\}} = 1 \qquad \forall R\in\mathcal{R}$$
$$x_{\{E,e\}} = \sum_{r\in\mathcal{L}_\mathcal{R}} x_{\{R,r,E,e\}} \qquad \forall E\in\mathcal{E} \text{ and } \forall R\in\{R: E=\mathcal{N}^1(R) \text{ or } R: E=\mathcal{N}^2(R)\}$$
$$x_{\{R,r\}} = \sum_{e\in\mathcal{L}_\mathcal{E}} x_{\{R,r,E,e\}} \qquad \forall R\in\mathcal{R} \text{ and } \forall E=\mathcal{N}^1(R) \text{ or } E=\mathcal{N}^2(R)$$
$$x_{\{E,e\}}\in\{0,1\} \qquad \forall E\in\mathcal{E}, e\in\mathcal{L}_\mathcal{E}$$
$$x_{\{R,r\}}\in\{0,1\} \qquad \forall R\in\mathcal{R}, r\in\mathcal{L}_\mathcal{R}$$
$$x_{\{R,r,E,e\}}\in\{0,1\} \qquad \forall R\in\mathcal{R}, r\in\mathcal{L}_\mathcal{R},\ E\in\mathcal{E}, e\in\mathcal{L}_\mathcal{E}$$

$$\sum_{e\in\mathcal{L}_\mathcal{E}} x_{\{E,e\}} = 1 \qquad \forall E\in\mathcal{E}$$
$$\sum_{r\in\mathcal{L}_\mathcal{R}} x_{\{R,r\}} = 1 \qquad \forall R\in\mathcal{R}$$

Main Takeaway:
Each entity or relation variable can be assigned only 1 value

A Computational Approach to Relational Inference

$$\min \sum_{E\in\mathcal{E}}\sum_{e\in\mathcal{L}_\mathcal{E}} c_E(e)\cdot x_{\{E,e\}} + \sum_{R\in\mathcal{R}}\sum_{r\in\mathcal{L}_\mathcal{R}} c_R(r)\cdot x_{\{R,r\}}$$
$$+ \sum_{\substack{E_i,E_j\in\mathcal{E}\\ E_i\neq E_j}}\left[\sum_{r\in\mathcal{L}_\mathcal{R}}\sum_{e_1\in\mathcal{L}_\mathcal{E}} d^1(r,e_1)\cdot x_{\{R_{ij},r,E_i,e_1\}} + \sum_{r\in\mathcal{L}_\mathcal{R}}\sum_{e_2\in\mathcal{L}_\mathcal{E}} d^2(r,e_2)\cdot x_{\{R_{ij},r,E_j,e_2\}}\right]$$

subject to:

$$\sum_{e\in\mathcal{L}_\mathcal{E}} x_{\{E,e\}} = 1 \qquad \forall E\in\mathcal{E}$$
$$\sum_{r\in\mathcal{L}_\mathcal{R}} x_{\{R,r\}} = 1 \qquad \forall R\in\mathcal{R}$$
$$\boxed{\begin{aligned} x_{\{E,e\}} &= \sum_{r\in\mathcal{L}_\mathcal{R}} x_{\{R,r,E,e\}} \qquad \forall E\in\mathcal{E} \text{ and } \forall R\in\{R: E=\mathcal{N}^1(R) \text{ or } R: E=\mathcal{N}^2(R)\} \\ x_{\{R,r\}} &= \sum_{e\in\mathcal{L}_\mathcal{E}} x_{\{R,r,E,e\}} \qquad \forall R\in\mathcal{R} \text{ and } \forall E=\mathcal{N}^1(R) \text{ or } E=\mathcal{N}^2(R)\end{aligned}}$$
$$x_{\{E,e\}}\in\{0,1\} \qquad \forall E\in\mathcal{E}, e\in\mathcal{L}_\mathcal{E}$$
$$x_{\{R,r\}}\in\{0,1\} \qquad \forall R\in\mathcal{R}, r\in\mathcal{L}_\mathcal{R}$$
$$x_{\{R,r,E,e\}}\in\{0,1\} \qquad \forall R\in\mathcal{R}, r\in\mathcal{L}_\mathcal{R},\ E\in\mathcal{E}, e\in\mathcal{L}_\mathcal{E}$$

Main Takeaway:

Assures that the assignment to each entity or relation variable is consistent with respect to the assignment of neighboring variables

$$\boxed{\begin{aligned} x_{\{E,e\}} &= \sum_{r\in\mathcal{L}_\mathcal{R}} x_{\{R,r,E,e\}} \qquad \forall E\in\mathcal{E} \text{ and } \forall R\in\{R: E=\mathcal{N}^1(R) \text{ or } R: E=\mathcal{N}^2(R)\} \\ x_{\{R,r\}} &= \sum_{e\in\mathcal{L}_\mathcal{E}} x_{\{R,r,E,e\}} \qquad \forall R\in\mathcal{R} \text{ and } \forall E=\mathcal{N}^1(R) \text{ or } E=\mathcal{N}^2(R)\end{aligned}}$$

A Computational Approach to Relational Inference

Main Takeaway:

Integral constraints on Binary Variables

$$
\begin{aligned}
\min \quad & \sum_{E \in \mathcal{E}} \sum_{e \in \mathcal{L}_\mathcal{E}} c_E(e) \cdot x_{\{E,e\}} + \sum_{R \in \mathcal{R}} \sum_{r \in \mathcal{L}_\mathcal{R}} c_R(r) \cdot x_{\{R,r\}} \\
& + \sum_{\substack{E_i, E_j \in \mathcal{E} \\ E_i \neq E_j}} \left[\sum_{r \in \mathcal{L}_\mathcal{R}} \sum_{e_1 \in \mathcal{L}_\mathcal{E}} d^1(r, e_1) \cdot x_{\{R_{ij}, r, E_i, e_1\}} + \sum_{r \in \mathcal{L}_\mathcal{R}} \sum_{e_2 \in \mathcal{L}_\mathcal{E}} d^2(r, e_2) \cdot x_{\{R_{ij}, r, E_j, e_2\}} \right]
\end{aligned}
$$

subject to:

$$
\begin{aligned}
\sum_{e \in \mathcal{L}_\mathcal{E}} x_{\{E,e\}} &= 1 && \forall E \in \mathcal{E} \\
\sum_{r \in \mathcal{L}_\mathcal{R}} x_{\{R,r\}} &= 1 && \forall R \in \mathcal{R} \\
x_{\{E,e\}} &= \sum_{r \in \mathcal{L}_\mathcal{R}} x_{\{R,r,E,e\}} && \forall E \in \mathcal{E} \text{ and } \forall R \in \{R : E = \mathcal{N}^1(R) \text{ or } R : E = \mathcal{N}^2(R)\} \\
x_{\{R,r\}} &= \sum_{e \in \mathcal{L}_\mathcal{E}} x_{\{R,r,E,e\}} && \forall R \in \mathcal{R} \text{ and } \forall E = \mathcal{N}^1(R) \text{ or } E = \mathcal{N}^2(R) \\
x_{\{E,e\}} &\in \{0,1\} && \forall E \in \mathcal{E}, e \in \mathcal{L}_\mathcal{E} \\
x_{\{R,r\}} &\in \{0,1\} && \forall R \in \mathcal{R}, r \in \mathcal{L}_\mathcal{R} \\
x_{\{R,r,E,e\}} &\in \{0,1\} && \forall R \in \mathcal{R}, r \in \mathcal{L}_\mathcal{R},\ E \in \mathcal{E}, e \in \mathcal{L}_\mathcal{E}
\end{aligned}
$$

$$
\begin{aligned}
x_{\{E,e\}} &\in \{0,1\} && \forall E \in \mathcal{E}, e \in \mathcal{L}_\mathcal{E} \\
x_{\{R,r\}} &\in \{0,1\} && \forall R \in \mathcal{R}, r \in \mathcal{L}_\mathcal{R} \\
x_{\{R,r,E,e\}} &\in \{0,1\} && \forall R \in \mathcal{R}, r \in \mathcal{L}_\mathcal{R},\ E \in \mathcal{E}, e \in \mathcal{L}_\mathcal{E}
\end{aligned}
$$

Linear Programming Relaxation (LPR)

- To solve the ILP, a natural idea is to relax the integral constraints:

$$
\begin{aligned}
x_{\{E,e\}} &\geq 0 && \forall E \in \mathcal{E}, e \in \mathcal{L}_{\mathcal{E}} \\
x_{\{R,r\}} &\geq 0 && \forall R \in \mathcal{R}, r \in \mathcal{L}_{\mathcal{R}} \\
x_{\{R,r,E,e\}} &\geq 0 && \forall R \in \mathcal{R}, r \in \mathcal{L}_{\mathcal{R}}, \\
& && E \in \mathcal{E}, e \in \mathcal{L}_{\mathcal{E}}
\end{aligned}
$$

  - If the solution returned is an integer solution, then it's the solution to the ILP problem.
  - If the solution returned is non-integer, then a lower bound to the cost is achieved

Linear Programming Relaxation (LPR)

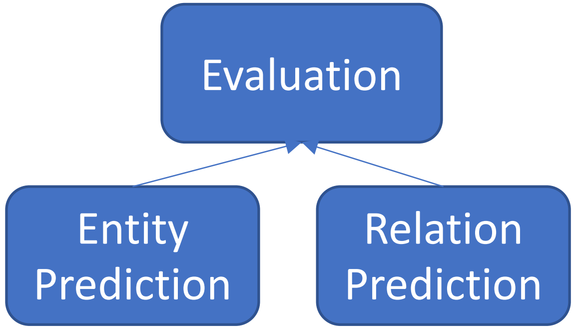
- Ways to deal with a non-integer solution:
  - **Rounding:** Finds an integer point that is close to the non-integer solution
    - **Merit:** Can be a good approximation to the optimal solution
    - **Demerit:** Outcome may not even be a legal solution to the problem
  - **Branch & Bound:**
    - Divides the ILP problem into several LP problems
    - Uses LPR to generate dual (upper and lower) bounds to reduce search space
      - ➢ Suppose $\min\{cx : x \in S, x \in \{0,1\}^n\}$ is fractional in a non-integer solution to the ILP it can be $x_i$lit into two sub problems:
        1. $\min\{cx : x \in S \cap \{x_i = 0\}\}$
        2. $\min\{cx : x \in S \cap \{x_i = 1\}\}$
  - **Cutting Plane:**
    - When a non-integer solution is given, it makes it infeasible

Experiments

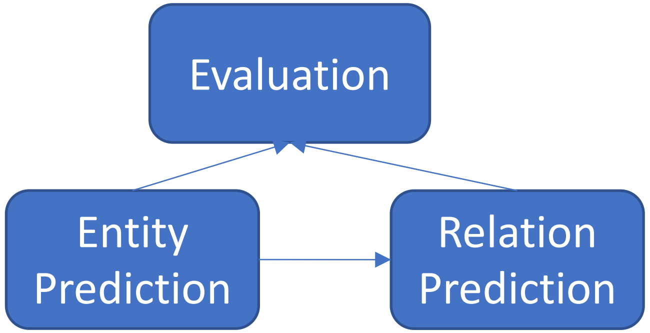
- The authors ran experiments on the problem of simultaneously recognizing entities and relations
- Dataset: Text Retrieval Collection (TREC) dataset : WSJ, AP,..
- Examples of constraints between relation variable

Relation	Entity1	Entity2	Example
located_in	loc	loc	(New York, US)
work_for	per	org	(Bill Gates, Microsoft)
orgBased_in	org	loc	(HP, Palo Alto)
live_in	per	loc	(Bush, US)
kill	per	per	(Oswald, JFK)

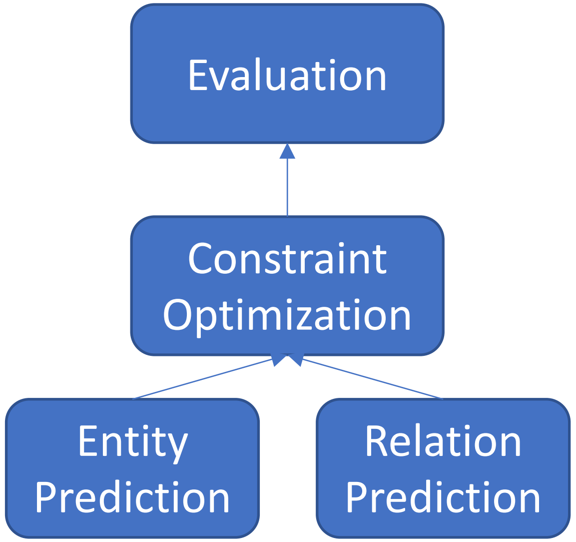
Experiments



- 4 methods of evaluating the model:
  - **Basic:**
    - Only tests the entity and relation classifiers, which are trained independently
    - The algorithm used to learn this : SNOW
      - Learns sparse network of linear functions
  - **Pipeline:**
    - Typical strategy in solving complex NLP problems
    - First trains an entity classifier on a different corpus in advance
    - Uses the prediction of the entities along with local features in training the relation classifier
    - Better performance for relation classifier when using predicted entities vs. using true entities



Experiments



- **Linear Programming Approach:**
  - Global inference procedure
  - Takes as input:
    - Constraints between entities and relations
    - Output of the entity classifier
    - Output of Relation classifier
  - It could potentially change the prediction for both entity and relation classifier
- **Omniscience** (Unrealistic in practical settings):
  - Tests the conceptual upper-bound of this entity/relation classification problem
  - Assumes that:
    - The entity classifier knows the correct relation labels
    - The relation classifier knows the correct entity labels

Results

Approach	person			organization			location		
	Rec.	Prec.	F_1	Rec.	Prec.	F_1	Rec.	Prec.	F_1
Basic	89.4	89.2	89.3	86.9	91.4	89.1	68.2	90.9	77.9
Pipeline	89.4	89.2	89.3	86.9	91.4	89.1	68.2	90.9	77.9
LP	90.4	90.0	90.2	88.5	91.7	90.1	71.5	91.0	80.1
Omniscient	94.9	93.5	94.2	92.3	96.5	94.4	88.3	93.4	90.8

Table 2: Results of Entity Classification

Approach	located_in			work_for			orgBased_in		
	Rec.	Prec.	F_1	Rec.	Prec.	F_1	Rec.	Prec.	F_1
Basic	54.7	43.0	48.2	42.1	51.6	46.4	36.1	84.9	50.6
Pipeline	51.2	51.6	51.4	41.4	55.6	47.5	36.9	76.6	49.9
LP	53.2	59.5	56.2	40.4	72.9	52.0	36.3	90.1	51.7
Omniscient	64.0	54.5	58.9	50.5	69.1	58.4	50.2	76.7	60.7

Approach	live_in			kill		
	Rec.	Prec.	F_1	Rec.	Prec.	F_1
Basic	39.7	61.6	48.3	82.1	73.6	77.6
Pipeline	42.6	62.2	50.6	83.2	76.4	79.6
LP	41.5	68.1	51.6	81.3	82.2	81.7
Omniscient	57.0	60.7	58.8	82.1	74.6	78.2

Table 3: Results of Relation Classification

Results

- **LP** performs consistently better than **basic** and **pipeline**
- **LP** uses the learned model and an additional ILP inference on top of them, and therefore outperforms pipeline, which uses entity predictions as new features in learning

Approach	person			organization			location		
	Rec.	Prec.	F_1	Rec.	Prec.	F_1	Rec.	Prec.	F_1
Basic	89.4	89.2	89.3	86.9	91.4	89.1	68.2	90.9	77.9
Pipeline	89.4	89.2	89.3	86.9	91.4	89.1	68.2	90.9	77.9
LP	90.4	90.0	90.2	88.5	91.7	90.1	71.5	91.0	80.1
Omniscient	94.9	93.5	94.2	92.3	96.5	94.4	88.3	93.4	90.8

Table 2: Results of Entity Classification

Approach	located_in			work_for			orgBased_in		
	Rec.	Prec.	F_1	Rec.	Prec.	F_1	Rec.	Prec.	F_1
Basic	54.7	43.0	48.2	42.1	51.6	46.4	36.1	84.9	50.6
Pipeline	51.2	51.6	51.4	41.4	55.6	47.5	36.9	76.6	49.9
LP	53.2	59.5	56.2	40.4	72.9	52.0	36.3	90.1	51.7
Omniscient	64.0	54.5	58.9	50.5	69.1	58.4	50.2	76.7	60.7

Approach	live_in			kill		
	Rec.	Prec.	F_1	Rec.	Prec.	F_1
Basic	39.7	61.6	48.3	82.1	73.6	77.6
Pipeline	42.6	62.2	50.6	83.2	76.4	79.6
LP	41.5	68.1	51.6	81.3	82.2	81.7
Omniscient	57.0	60.7	58.8	82.1	74.6	78.2

Table 3: Results of Relation Classification

Results

- The results of the omniscient classifiers reveal that there is still room for improvement

Approach	person			organization			location		
	Rec.	Prec.	F_1	Rec.	Prec.	F_1	Rec.	Prec.	F_1
Basic	89.4	89.2	89.3	86.9	91.4	89.1	68.2	90.9	77.9
Pipeline	89.4	89.2	89.3	86.9	91.4	89.1	68.2	90.9	77.9
LP	90.4	90.0	90.2	88.5	91.7	90.1	71.5	91.0	80.1
Omniscient	94.9	93.5	94.2	92.3	96.5	94.4	88.3	93.4	90.8

Table 2: Results of Entity Classification

Approach	located_in			work_for			orgBased_in		
	Rec.	Prec.	F_1	Rec.	Prec.	F_1	Rec.	Prec.	F_1
Basic	54.7	43.0	48.2	42.1	51.6	46.4	36.1	84.9	50.6
Pipeline	51.2	51.6	51.4	41.4	55.6	47.5	36.9	76.6	49.9
LP	53.2	59.5	56.2	40.4	72.9	52.0	36.3	90.1	51.7
Omniscient	64.0	54.5	58.9	50.5	69.1	58.4	50.2	76.7	60.7

Approach	live_in			kill		
	Rec.	Prec.	F_1	Rec.	Prec.	F_1
Basic	39.7	61.6	48.3	82.1	73.6	77.6
Pipeline	42.6	62.2	50.6	83.2	76.4	79.6
LP	41.5	68.1	51.6	81.3	82.2	81.7
Omniscient	57.0	60.7	58.8	82.1	74.6	78.2

Table 3: Results of Relation Classification

Results

- One of the more significant results → improvement in quality of predictions
- if the label of an active relation is predicted correctly, and if both its entities are also predicted correctly → *coherent solution*
- *Quality of a decision* → |coherent|/(|coherent| + |incoherent|)
- *Pipeline* and *Basic* → 5% to 25% incoherent, *LP* → 0%
- Another significant result → adding constraint not present during learning
  - One of the key motivations for this framework.
  - The ability to incorporate knowledge not present in training but only becomes available during testing

Conclusion

- Presented an LP approach for global inference:
  - Works for non-sequential data
  - Provides an efficient way of finding optimal solution
  - Predictions are coherent
- This framework became known as the ILP formulation of NLP Problems
- What it really is doing is abductive reasoning
  - **Observations:** Sentence containing entities and relations
  - **Simple most likely explanation:** Predictions from individual models followed by Global inference