Markov Logic Networks for Text Mining: A Qualitative and Empirical Comparison with Integer Linear Programming

Luis Gerardo Mojica and Vincent Ng
Human Language Technology Research Institute
University of Texas at Dallas
Motivation

• Many NLP systems adopt a pipeline architecture
  – A given task is broken into a sequence of sub-tasks, where the output of one sub-task is the input of the next one

• Strengths
  – Modularity, modeling convenience, manageable computational complexity, ...

• Weakness
  – Error propagation
Joint Inference Frameworks

• Integer Linear Programming (ILP)
• Markov Logic Networks (MLNs)

• Enable manual specification of output constraints
  – Allow incorporation of background knowledge
  – Address error propagation by allowing downstream components to influence upstream components
Joint Inference Frameworks (Cont’)

• ILP is used a lot more than MLNs in NLP

• Is ILP better than MLNs?
• Should we care about MLNs at all?
• …
Plan for the Talk

• Preliminaries
  – ILP
  – MLN
  – Task: fine-grained opinion extraction

• ILP and MLN formulations of the task

• Qualitative and empirical comparison
  – Strengths and weaknesses of MLNs
  – Evaluation
Plan for the Talk

• Preliminaries
  – ILP
  – MLN
  – Task: fine-grained opinion extraction

• ILP and MLN formulation of the task

• Qualitative and empirical comparison
  – Strengths and weaknesses of MLNs
  – Evaluation
ILP

• A constrained optimization framework
  – **Goal**: optimize an objective function subject to a set of linear (equality and inequality) constraints

\[
\text{Maximize:} \quad f(x_1, x_2, \ldots, x_n) \\
\text{Subject to:} \quad g_j(x_1, x_2, \ldots, x_n) \geq b_j \quad (j = 1, 2, \ldots, m)
\]

– A variety of methods can be used to solve ILP problems
– Software for solving ILP problems available
MLNs

• A statistical relational learning approach
• Combines graphical models with first-order logic
• A MLN is a set of weighted first-order logic formulas \((f_i, w_i)\), where \(w_i\) is the weight associated with formula \(f_i\)
  
  
  \[- 0.8 \quad \forall x \text{ Smoke}(x) \rightarrow \text{Cancer}(x)\]
MLNs

• A statistical relational learning approach
• Combines graphical models with first-order logic
• A MLN is a set of weighted first-order logic formulas \((f_i, w_i)\), where \(w_i\) is the weight associated with formula \(f_i\)
  – 0.8 \(\forall x \, \text{Smoke}(x) \rightarrow \text{Cancer}(x)\)
• Given a set of constants that model objects in the domain of interest, a MLN defines a Markov network
  – One node per grounded predicate
    • Cancer(John), Cancer(Mary), Cancer(Ed),…
    • Smoke(John), Smoke(Mary), Smoke(Ed),…
  – One feature per each grounding of each first-order formula
    • Smoke(John) \(\rightarrow\) Cancer(John), Smoke(Ed) \(\rightarrow\) Cancer(Ed),…
    • Feature weight is the weight of the first-order formula
MLN: Key Learning Task

- **Weight learning**: learn the weights of the soft formulas so that the conditional likelihood of the training data is optimized
  - In ILP, there is no learning
  - In ILP, the function to be optimized is user-defined
MLN: Key Inference Task

- **MAP inference**: Finding the most probable world
  - A world: assignment of values to the grounded predicates
  - Probability of a world $\omega$ is given by
    \[
    \Pr(\omega) = \frac{1}{Z} \exp \left( \sum_i w_i N(f_i, \omega) \right)
    \]
    $N(f_i, \omega)$ is the number of groundings of $f_i$ that evaluate to True in $\omega$
    $Z$ is the normalization constant
  - Software for MAP inference available
    - can be reduced to **propositional** MAP inference and the MAP can be found using an ILP solver
Plan for the Talk

• Preliminaries
  – ILP
  – MLN
  – Task: fine-grained opinion extraction

• ILP and MLN formulation of the task

• Qualitative and empirical comparison
  – Strengths and weaknesses of MLNs
  – Evaluation
Task: Fine-Grained Opinion Extraction

• involves two subtasks
  – Entity extraction
  – Relation extraction
Fine-Grained Opinion Extraction

- Subtask 1: **Entity extraction**
  - Extracts three types of entities
    - opinions
    - their **sources** *(who expressed the opinions?)*
    - their **targets** *(what the opinions are about)*
Fine-Grained Opinion Extraction

• Subtask 1: **Entity extraction**
  – Extracts three types of entities
    • opinions
    • their **sources** (*who* expressed the opinions?)
    • their **targets** (*what* the opinions are **about**)
Fine-Grained Opinion Extraction

• Subtask 1: **Entity extraction**
  
  – Extracts three types of entities
    
    • **opinions**
    
    • their **sources** *(who expressed the opinions?)*
    
    • their **targets** *(what the opinions are about)*

The agency *considered* that the trade was favorable, but their partners are *still not satisfied.*
Fine-Grained Opinion Extraction

• Subtask 1: *Entity extraction*
  
  – Extracts three types of entities
    
    • *opinions*
    
    • their *sources* (who expressed the opinions?)
    
    • their *targets* (what the opinions are *about*)

  The agency *considered* that the trade was favorable, but *their partners* are *still not satisfied*. 
Fine-Grained Opinion Extraction

• Subtask 1: **Entity extraction**
  – Extracts three types of entities
    • **opinions**
    • their **sources** (who expressed the opinions?)
    • their **targets** (what the opinions are **about**)

The agency considered that the trade was favorable, but their partners are still not satisfied.
Fine-Grained Opinion Extraction

• Subtask 1: **Entity extraction**
  – Extracts three types of entities
    • opinions
    • their sources (**who** expressed the opinions?)
    • their targets (**what** the opinions are **about**)  

  – Some opinions don’t have a source and/or target
    • Source-implicit opinions
    • Target-implicit opinions
Fine-Grained Opinion Extraction

- Subtask 2: **Relation extraction**
  - Extracts two types of relations
    - *is_from* (between an opinion and its source)
    - *is_about* (between an opinion and its target)

The agency considered that the trade was favorable, but their partners are still not satisfied.
Fine-Grained Opinion Extraction

• Subtask 2: **Relation extraction**
  – Extracts two types of relations
    • is\_from (between an opinion and its source)
    • is\_about (between an opinion and its target)

The agency **considered** that the trade was favorable, but **their partners** are **still not satisfied**.
Fine-Grained Opinion Extraction

• Subtask 2: Relation extraction
  – Extracts two types of relations
    • is_from (between an opinion and its source)
    • is_about (between an opinion and its target)

The agency considered that the trade was favorable, but their partners are still not satisfied.
Fine-Grained Opinion Extraction

• Subtask 2: **Relation extraction**
  – Extracts two types of relations
    • **is_from** (between an opinion and its source)
    • **is_about** (between an opinion and its target)

The agency **considered** that the trade **was favorable,** but their partners are still not satisfied.
Fine-Grained Opinion Extraction

• Subtask 2: **Relation extraction**
  – Extracts two types of relations
    • **is_from** (between an opinion and its source)
    • **is_about** (between an opinion and its target)

The agency considered that the trade was favorable, but their partners are still not satisfied.
Why Joint Inference for Fine-Grained Opinion Extraction?

- Errors propagate in a pipeline architecture

Diagram:
- Document → Entity extraction → Relation extraction
Why Joint Inference for Fine-Grained Opinion Extraction?

• Errors propagate in a pipeline architecture

Train a CRF to extract the 3 types of entities

Train two SVMs to determine if an opinion is source-implicit or target-implicit (or both)
Why Joint Inference for Fine-Grained Opinion Extraction?

- Errors propagate in a pipeline architecture

1. Train a CRF to extract the 3 types of entities.
2. For each pair of entities extracted, train an SVM to determine what type of relation exists between them, if any.
3. Train two SVMs to determine if an opinion is source-implicit or target-implicit (or both).
Plan for the Talk

• Preliminaries
  – ILP
  – MLN
  – Task: fine-grained opinion extraction

• ILP and MLN formulations of the task

• Qualitative and empirical comparison
  – Strengths and weaknesses of MLNs
  – Evaluation
Key Issue

• Encode output constraints
Constraint 1
(Consistency on entity extraction)

• Every text span has exactly one label (S, T, O, N)

\[ \exists c \text{ Span}(i, c!) \]
Constraint 1
(Consistency on entity extraction)

• Every text span has exactly one label (S, T, O, N)

\[ \exists c \text{ Span}(i,c!) \]
Constraint 2
(Consistency on entity extraction)

- Entities cannot overlap

\[ \text{Overlap}(i,j) \rightarrow \text{Span}(i,N) \lor \text{Span}(j,N) \]
Constraint 3
(Consistency on Entity & Rel. Extraction)

• An opinion is source-implicit if and only if it doesn’t have a source
• An opinion is target-implicit if and only if it doesn’t have a target

\[
\text{Implicit}_\text{src}(i) \iff \neg \text{Is}_\text{from}(i, j) \\
\text{Implicit}_\text{trg}(i) \iff \neg \text{Is}_\text{about}(i, j)
\]
Constraint 4
(Consistency on Entity & Rel. Extraction)

• If the entity extractor predicts a span to be a source or target, it must also be predicted by the relation extractor as being linked to an opinion

• $\text{Span}(j, S) \rightarrow \exists i \text{ Is}_\text{from}(i, j)$

• $\text{Is}_\text{from}(i, j) \rightarrow \text{Span}(i, O)$

• $\text{Span}(j, T) \rightarrow \exists i \text{ Is}_\text{about}(i, j)$

• $\text{Is}_\text{about}(i, j) \rightarrow \text{Span}(i, O)$
Constraint 4
(Consistency on Entity & Rel. Extraction)

• If the entity extractor predicts a span to be a source or target, it must also be predicted by the relation extractor as being linked to an opinion

- $\text{Span}(j, S) \rightarrow \exists i \text{ Is\_from}(i, j)$
- $\text{Is\_from}(i, j) \rightarrow \text{Span}(i, O)$

- $\text{Span}(j, T) \rightarrow \exists i \text{ Is\_about}(i, j)$
- $\text{Is\_about}(i, j) \rightarrow \text{Span}(i, O)$
Prior Knowledge as Soft Evidence

• When doing joint inference over the test instances, we can’t just have constraints
  – We need knowledge

• The probabilistic classifications made by the 3 independently-trained models (entity extractor, relation classifier, implicit classifier) can be exploited as prior knowledge when encoded as soft evidences
ILP Formulation: Constraint 1

• Every text span has exactly one label (S, T, O, N)
  – $x_{iz}$: binary variable whose value is 1 if span i is assigned entity label z
ILP Formulation: Constraint 2

• Entities cannot overlap
  – $x$: binary variable
  – $i, j$: span
  – $z$: entity label

\[
\sum_{z \neq N} x_{iz} + \sum_{z \neq N} x_{jz} \leq 1
\]
ILP Formulation: Constraint 3

- An opinion is source-implicit if and only if it doesn’t have a source
- An opinion is target-implicit if and only if it doesn’t have a target
- \( u_{ij} \): 1 iff opinion i is related to j in relation type k
- \( v_{ik} \): 1 iff opinion i is implicit w.r.t. relation type k

\[
\sum_{j \in A_k} u_{ij} = 1 - v_{ik} + a_{ik} + b_{ik}
\]

\[
a_{ik} \leq 1 - v_{ik}; \quad b_{ik} \leq 1 - v_{ik}
\]
ILP Formulation: Constraint 4

• If the entity extractor predicts a span to be a source or target, it must also be predicted by the relation extractor as being linked to an opinion

• $x_{jz}$: 1 iff span $j$ is predicted to have entity label $z$

• $u_{ij}$: 1 iff opinion $i$ is related to span $j$

$$
\sum_{i \in O} u_{ij} = x_{jz} + c_{jk} + d_{jk}
$$

$$
c_{jk} \leq x_{jz}; \quad d_{jk} \leq x_{jz}
$$
ILP Formulation: Objective Function

- Weighted combination of the prior knowledge provided by the 3 models

\[
\arg\max_{x,u,v} \lambda \sum_{i \in S} \sum_{z} f_{iz} x_{iz} + (1 - \lambda) \sum_{k} \sum_{i \in O} \left( \sum_{j \in A_k} r_{ij} u_{ij} + r_{i0} v_{ik} \right)
\]
ILP Formulation: Objective Function

- Weighted combination of the prior knowledge provided by the 3 models

\[
\arg \max_{x,u,v} \lambda \sum_{i \in S} \sum_{z} f_{iz} v_{iz} + (1 - \lambda) \sum_{k} \sum_{i \in O} \left( \sum_{j \in A_k} r_{ij} u_{ij} + r_{i\emptyset} v_{ik} \right)
\]

Prob. classification of entity extractor for span i
ILP Formulation: Objective Function

- Weighted combination of the prior knowledge provided by the 3 models

\[
\arg \max_{x,u,v} \lambda \sum_{i \in S} \sum_{z} f_{iz} v_{iz} + (1 - \lambda) \sum_{k} \sum_{i \in O} \left( \sum_{j \in A_{k}} r_{ij} u_{ij} + r_{i\emptyset} v_{ik} \right)
\]

- Prob. classification of entity extractor for span i
- Prob. classification of relation classifier for spans i & j
ILP Formulation: Objective Function

- Weighted combination of the prior knowledge provided by the 3 models

\[
\arg \max_{x,u,v} \lambda \sum_{i \in S} \sum_{z} f_{iz} v_{iz} + (1 - \lambda) \sum_{k} \sum_{i \in O} \left( \sum_{j \in A_k} r_{ij} u_{ij} + r_{i0} v_{ik} \right)
\]

- Prob. classification of entity extractor for span i
- Prob. classification of relation classifier for spans i & j
- Prob. classification of implicit classifier for span i
ILP Formulation: Objective Function

- Weighted combination of the prior knowledge provided by the 3 models

\[
\text{arg max}_{x,u,v} \lambda \sum_{i \in S} \sum_{z} f_{iz} v_{iz} + (1 - \lambda) \sum_{k} \sum_{i \in O} \left( \sum_{j \in A_k} r_{ij} u_{ij} + r_{i0} u_{ik} \right)
\]

ILP incorporates prior knowledge into the objective function, whereas MLN encodes it as soft evidences.
Plan for the Talk

• Preliminaries
  – ILP
  – MLN
  – Task: fine-grained opinion extraction

• ILP and MLN formulation of the task

• Qualitative and empirical comparison
  – Strengths and weaknesses of MLNs
  – Evaluation
MLNs: Strengths

• The ability to employ soft constraints and learn weights for them

What if we want to exploit semantic role labels?

• A span i with verb sense s is likely to have entity type c
  \[ \text{Sense}(i, s+) \Rightarrow \text{Span}(i, c+) \]

• A span i with semantic role r is likely to have entity type c
  \[ \text{Role}(i, r+) \Rightarrow \text{Span}(i, c+) \]

Soft formulas: manually or automatically attach weights to them
MLNs: Strengths

• The ability to employ soft constraints and learn weights for them

• Compact representation

• Ease of specification
MLNs: Strengths

• The ability to employ soft constraints and learn weights for them

• Compact representation

• Ease of specification

Especially important when we have tasks with a large domain and with complex output constraints
MLNs: Weaknesses

• Exponential time and space complexity
  – Need to ground an MLN
  – But... lifted inference algorithms have been developed

• Failure to exploit prior knowledge (i.e., the soft evidences) in weight learning
  – Can only be applied during test time
  – ILP doesn’t have to deal with this issue: no learning

• No support for functions
  – To express i != j, need to define predicate Neq(i,j)
  – Could incur preprocessing overhead
  – ILP natively supports functions
Plan for the Talk

• Preliminaries
  – ILP
  – MLN
  – Task: fine-grained opinion extraction

• ILP and MLN formulation of the task

• Qualitative and empirical comparison
  – Strengths and weaknesses of MLNs
  – Evaluation
Experimental Setup

• Corpus
  – 433 documents in the MPQA 2.0 corpus after discarding those that are ill-formed

• Software packages
  – Gurobi: ILP joint inference
  – Tuffy: MLN joint inference

• Evaluation metrics: R/P/F, inference time
Entity Extraction F-Scores

<table>
<thead>
<tr>
<th></th>
<th>Opinion</th>
<th>Target</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>ILP</td>
<td>59.4</td>
<td>40.1</td>
<td>48.1</td>
</tr>
<tr>
<td>MLN</td>
<td>56.8</td>
<td>42.6</td>
<td>60.4</td>
</tr>
</tbody>
</table>

- MLN underperforms ILP on Opinion extraction but outperforms it on Source and Target extraction
Relation Extraction F-Scores

<table>
<thead>
<tr>
<th></th>
<th>Is from</th>
<th>Is about</th>
</tr>
</thead>
<tbody>
<tr>
<td>ILP</td>
<td>19.8</td>
<td>22.7</td>
</tr>
<tr>
<td>MLN</td>
<td>21.0</td>
<td>28.5</td>
</tr>
</tbody>
</table>

- MLN outperforms ILP on both relation types due to better Source and Target extraction
Inference Time

- ILP: 550 seconds
- MLN: 7,200 seconds
Summary

• Empirical results are too preliminary
  – Corpus is too small and constraints are too simple to reveal the frameworks’ relative strengths and weaknesses
    • E.g., No soft constraints
  – Can’t draw any conclusions from the empirical results

• Qualitative comparisons are more important
  – MLN strengths: Compact representation, ease of specification, ability to encode soft constraints
  – MLN weakness: inability to scale large problems

• Ongoing work: fast and scalable inference for MLNs so that they can be applied to complex NLP tasks