Max-Margin Weight Learning for Markov Logic Networks

Tuyen N. Huynh and Raymond J. Mooney

Machine Learning Group
Department of Computer Science
The University of Texas at Austin

ECML-PKDD-2009, Bled, Slovenia

Motivation

- Markov Logic Network (MLN) combining probability and first-order logic is an expressive formalism which subsumes other SRL models
- All of the existing training methods for MLNs learn a model that produce good predictive probabilities

Motivation (cont.)

- In many applications, the actual goal is to optimize some application specific performance measures such as classification accuracy, F_1 score, etc…
- Max-margin training methods, especially Structural Support Vector Machines (SVMs), provide the framework to optimize these application specific measures

→ Training MLNs under the max-margin framework

Outline

- Background
  - MLNs
  - Structural SVMs
- Max-Margin Markov Logic Networks
  - Formulation
  - LP-relaxation MPE inference
- Experiments
- Future work
- Summary

Background

Markov Logic Networks (MLNs)

[Richardson & Domingos, 2006]

- An MLN is a weighted set of first-order formulas

0.25	HasWord(“assignment”,p) => PageClass(Course,p)
0.19	PageClass(Course,p1) ^ Linked(p1,p2) => PageClass(Faculty,p2)

- Larger weight indicates stronger belief that the clause should hold
- Probability of a possible world (a truth assignment to all ground atoms) x:

$$P(X = x) = \frac{1}{Z} \exp\left(\sum_i w_i n_i(x) \right)$$

w_i: Weight of formula i

$n_i(x)$: No. of true groundings of formula i in x

Inference in MLNs

- **MAP/MPE inference:** find the most likely state of a set of query atoms given the evidence

$$y_{MAP} = \arg\max_{y \in Y} P(y \mid x)$$

  - MaxWalkSAT algorithm [Kautz et al., 1997]
  - Cutting Plane Inference algorithm [Riedel, 2008]
- **Computing the marginal conditional probability of a set of query atoms:** P(y|x)
  - MC-SAT algorithm [Poon & Domingos, 2006]
  - Lifted first-order belief propagation [Singla & Domingos, 2008]

Existing weight learning methods in MLNs

- **Generative**: maximize the Pseudo-Log Likelihood [Richardson & Domingos, 2006]
- **Discriminative** : maximize the Conditional Log Likelihood (CLL) [Singla & Domingos, 2005], [Lowd & Domingos, 2007], [Huynh & Mooney, 2008]

Generic Strutural SVMs[Tsochantaridis et.al., 2004]

- Learn a discriminant function f: $\mathbf{X} \times \mathbf{Y} \rightarrow \mathbf{R}$

$$f(x, y; w) = w^T \Phi(x, y)$$

- Predict for a given input x:

$$h(x; w) = \arg\max_{y \in Y} w^T \Phi(x, y)$$

- Maximize the separation margin:

$$\gamma(x, y; w) = w^T \Phi(x, y) - \max_{y' \in Y \backslash y} w^T \Phi(x, y')$$

- Can be formulated as a quadratic optimization problem

Generic Strutural SVMs (cont.)

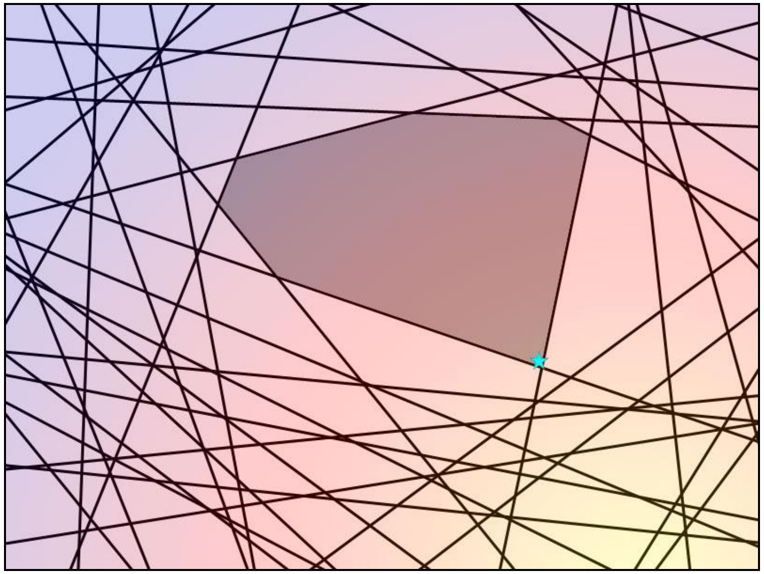
- [Joachims et.al., 2009] proposed the 1-slack formulation of the Structural SVM:

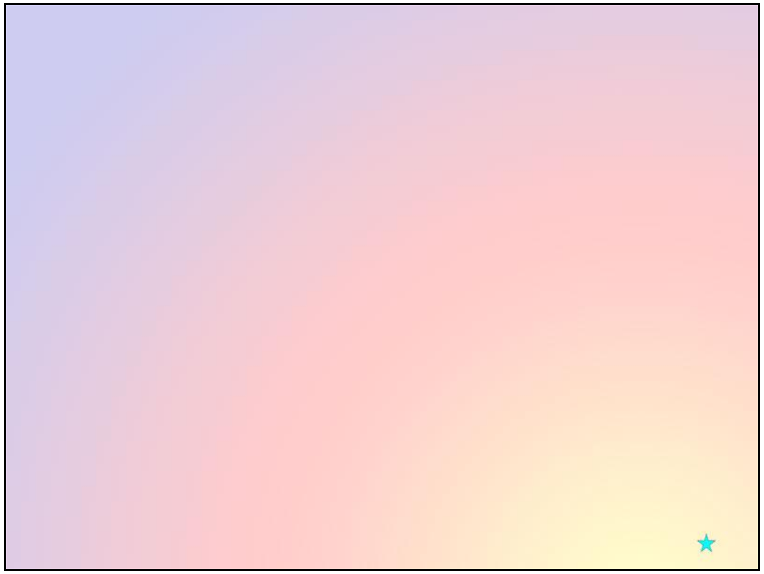
$$\min_{w,\xi>0} \frac{1}{2} w^T w + C\xi$$

$$st. \quad \forall (y'_1, \ldots, y'_n) \in Y^n : \frac{1}{n} w^T \sum_{i=1}^{n} [\Phi(x_i, y_i) - \Phi(x_i, \bar{y}_i)] \geq \frac{1}{n} \sum_{i=1}^{n} \Delta(y_i, y'_i) - \xi$$

→Make the original cutting-plane algorithm [Tsochantaridis et.al., 2004] run faster and more scalable

Cutting plane algorithm for solving the structural SVMs





Structural SVM Problem

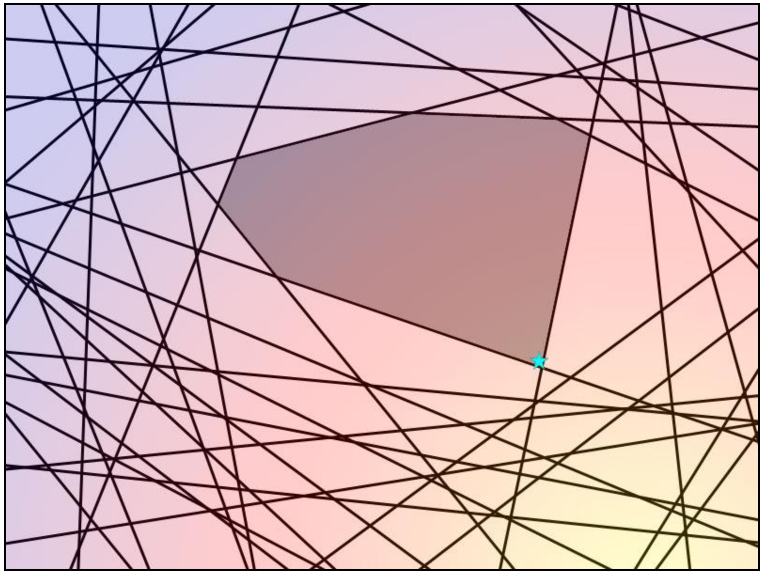
- Exponential constraints
- Most are dominated by a small set of "important" constraints

Cutting plane algorithm

- Repeatedly finds the next most violated constraint…
- … until cannot find any new constraint

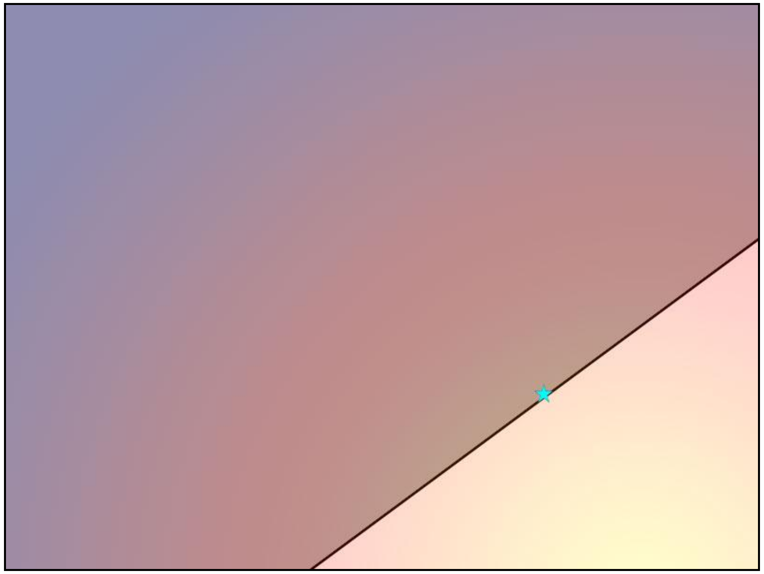
*Slide credit: Yisong Yue

Cutting plane algorithm for solving the 1-slack SVMs



Structural SVM Problem

- Exponential constraints
- Most are dominated by a small set of "important" constraints

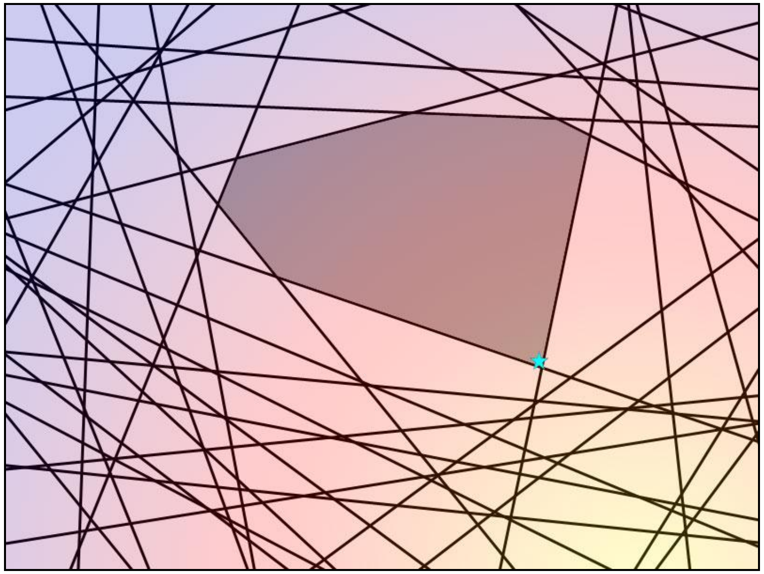


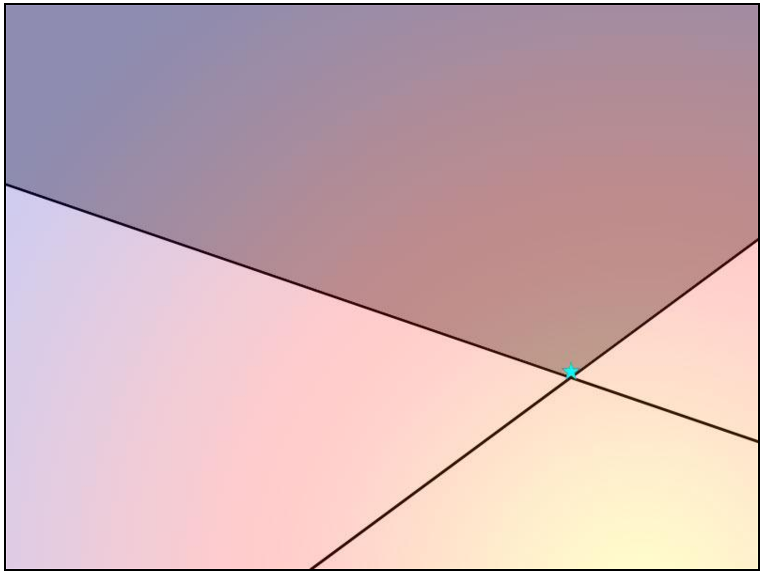
Cutting plane algorithm

- Repeatedly finds the next most violated constraint…
- … until cannot find any new constraint

*Slide credit: Yisong Yue

Cutting plane algorithm for solving the 1-slack SVMs





Structural SVM Problem

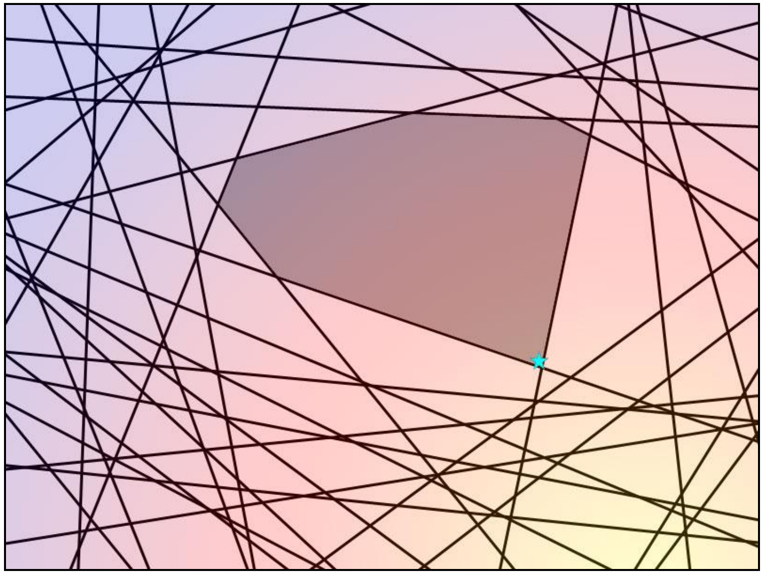
- Exponential constraints
- Most are dominated by a small set of "important" constraints

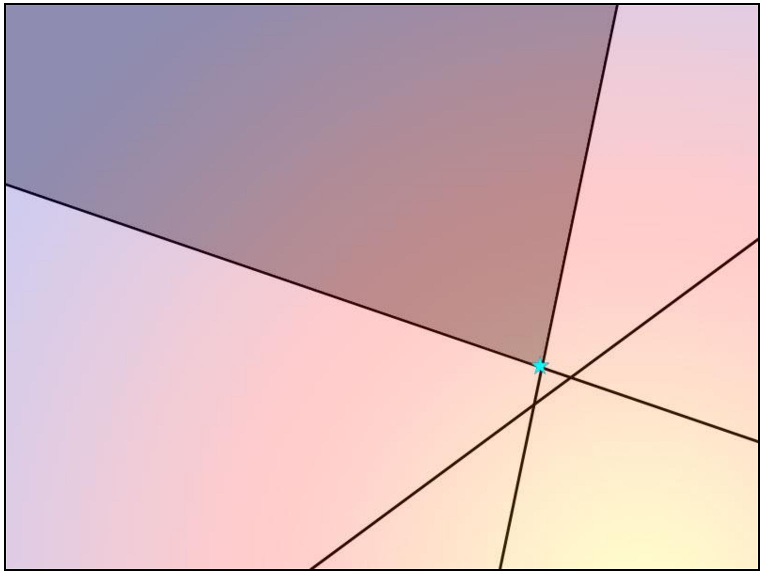
Cutting plane algorithm

- Repeatedly finds the next most violated constraint…
- … until cannot find any new constraint

*Slide credit: Yisong Yue

Cutting plane algorithm for solving the 1-slack SVMs





Structural SVM Problem

- Exponential constraints
- Most are dominated by a small set of "important" constraints

Cutting plane algorithm

- Repeatedly finds the next most violated constraint…
- … until cannot find any new constraint

*Slide credit: Yisong Yue

Applying the generic structural SVMs to a new problem

- Representation: $\Phi(\mathbf{x},\mathbf{y})$
- Loss function: $\Delta(\mathbf{y},\mathbf{y}')$
- Algorithms to compute
  - Prediction:

    $$\hat{y} = \arg\max_{y' \in Y} \{w^T \Phi(x, y')\}$$

  - Most violated constraint: separation oracle [Tsochantaridis et.al., 2004] or loss-augmented inference [Taskar et.al.,2005]

$$\hat{y} = \arg\max_{y' \in Y} \{w^T \Phi(x, y') + \Delta(y, y')\}$$

Max-Margin Markov Logic Networks

Formulation

- Maximize the ratio:

$$\frac{P(y \mid x)}{P(\hat{y} \mid x)} = \frac{\sum_i w_i n_i(x, y)}{\sum_i w_i n_i(x, \hat{y})}$$

$$\hat{y} = \arg\max_{\bar{y} \in Y \setminus y} P(\bar{y} \mid x)$$

- Equivalent to maximize the separation margin:

$$\gamma(x, y; w) = w^T n(x, y) - w^T n(x, \hat{y})$$
$$= w^T n(x, y) - \max_{y' \in Y \setminus y} w^T n(x, y')$$

Joint feature: $\Phi(x,y)$

- Can be formulated as a 1-slack Structural SVMs

Problems need to be solved

- MPE inference:

$$\hat{y} = \arg\max_{y' \in Y} w^T n(x, y')$$

- Loss-augmented MPE inference:

$$\hat{y} = \arg\max_{y' \in Y} \left\{ \Delta(y, y') + w^T n(x, y') \right\}$$

Problem: Exact MPE inference in MLNs are intractable

Solution: Approximation inference via relaxation methods [Finley et.al.,2008]

Relaxation MPE inference for MLNs

- Many work on approximating the Weighted MAX-SAT via Linear Programming (LP) relaxation [Goemans and Williamson, 1994], [Asano and Williamson, 2002], [Asano, 2006]
  - Convert the problem into an Integer Linear Programming (ILP) problem
  - Relax the integer constraints to linear constraints
  - Round the LP solution by some randomized procedures
  - Assume the weights are finite and positive

Relaxation MPE inference for MLNs (cont.)

- Translate the MPE inference in a ground MLN into an Integer Linear Programming (ILP) problem:
  - Convert all the ground clauses into clausal form
  - Assign a binary variable y_i to each unknown ground atom and a binary variable z_j to each non-deterministic ground clause
  - Translate each ground clause into linear constraints of y_i's and z_j's

Relaxation MPE inference for MLNs (cont.)

Ground MLN

3 InField(B1,Fauthor,P01)

0.5 InField(B1,Fauthor,P01) v InField(B1,Fvenue,P01)

-1 InField(B1,Ftitle,P01) v InField(B1,Fvenue,P01)

!InField(B1,Fauthor,P01) v !InField(a1,Ftitle,P01).
!InField(B1,Fauthor,P01) v !InField(a1,Fvenue,P01).
!InField(B1,Ftitle,P01) v !InField(a1,Fvenue,P01).

Translated ILP problem

$$\max_{y,z} 3y_1 + 0.5z_1 + z_2$$

$$st. \quad y_1 + y_2 \geq z_1$$

$$1 - y_2 \geq z_2$$

$$1 - y_3 \geq z_2$$

$$(1 - y_1) + (1 - y_2) \geq 1$$

$$(1 - y_1) + (1 - y_3) \geq 1$$

$$(1 - y_2) + (1 - y_3) \geq 1$$

$$y_i, z_j \in \{0,1\}$$

Relaxation MPE inference for MLNs (cont.)

- LP-relaxation: relax the integer constraints {0,1} to linear constraints [0,1].
- Adapt the ROUNDUP [Boros and Hammer, 2002] procedure to round the solution of the LP problem
  - Pick a non-integral component and round it in each step

Loss-augmented LP-relaxation MPE inference

- Represent the loss function as a linear function of y_i's:

$$\Delta_{\text{Hammming}}(y^T, y) = \sum_{i:y_i^T=0} y_i + \sum_{i:y_i^T=1} (1 - y_i)$$

- Add the loss term to the objective of the LP-relaxation → the problem is still a LP problem → can be solved by the previous algorithm

Experiments

Collective multi-label webpage classification

- WebKB dataset [Craven and Slattery, 2001] [Lowd and Domingos, 2007]
- 4,165 web pages and 10,935 web links of 4 departments
- Each page is labeled with a subset of 7 categories: *Course, Department, Faculty, Person, Professor, Research Project, Student*
- MLN [Lowd and Domingos, 2007] :

Has(+word,page) → PageClass(+class,page)
¬Has(+word,page) → PageClass(+class,page)
PageClass(+c1,p1) ^ Linked(p1,p2) → PageClass(+c2,p2)

Collective multi-label webpage classification (cont.)

- Largest ground MLN for one department:
  - 8,876 query atoms
  - 174,594 ground clauses

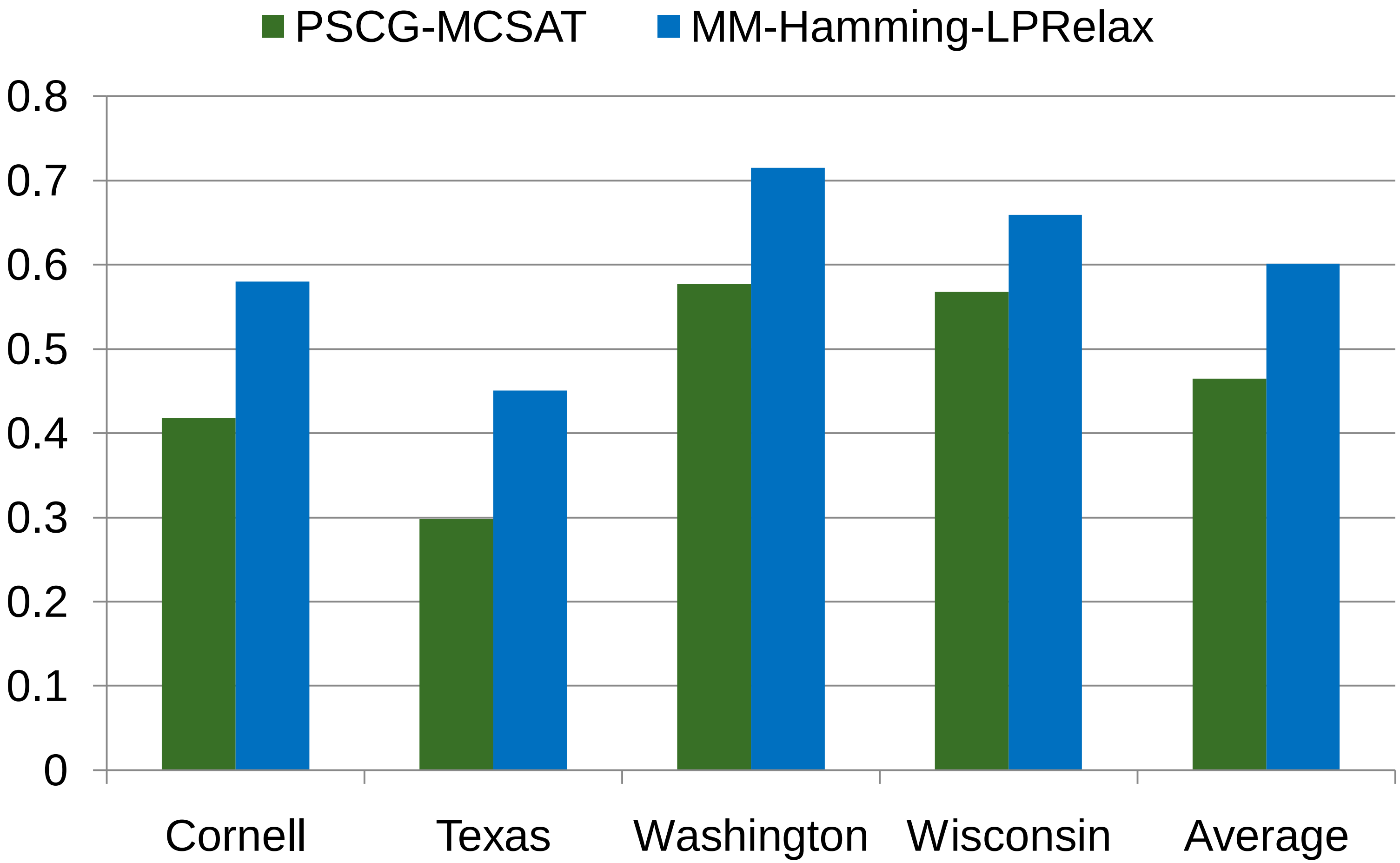
Citation segmentation

- Citeseer dataset [Lawrence et.al., 1999] [Poon and Domingos, 2007]
- 1,563 citations, divided into 4 research topics
- Each citation is segmented into 3 fields: *Author, Title, Venue*
- Used the simplest MLN in [Poon and Domingos, 2007]
- Largest ground MLN for one topic:
  - 37,692 query atoms
  - 131,573 ground clauses

Experimental setup

- 4-fold cross-validation
- Metric: F_1 score
- Compare against the Preconditioned Scaled Conjugated Gradient (PSCG) algorithm
- Train with 5 different values of C: 1, 10, 100, 1000, 10000 and test with the one that performs best on training
- Use Mosek to solve the QP and LP problems

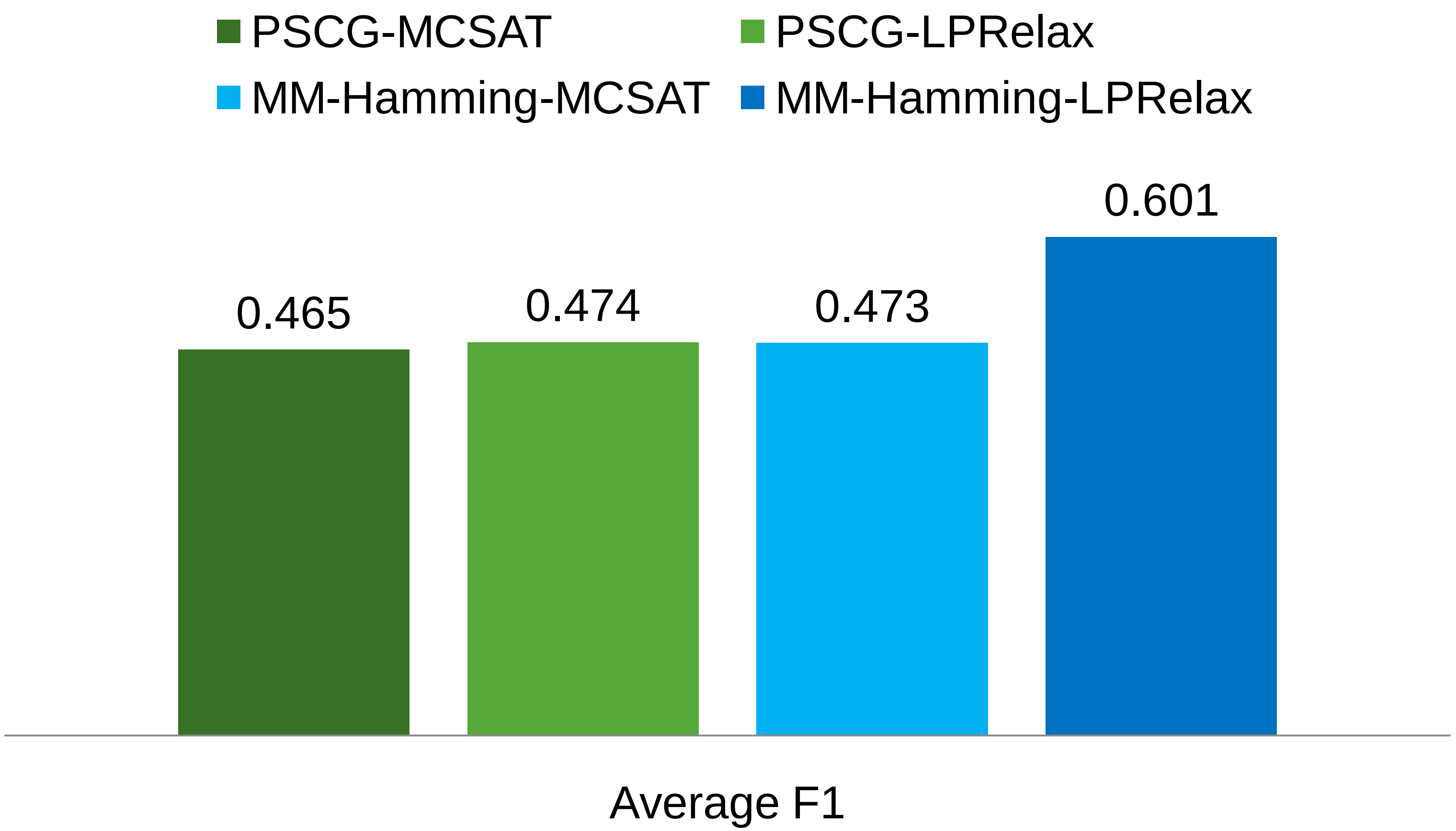
F_1 scores on WebKB



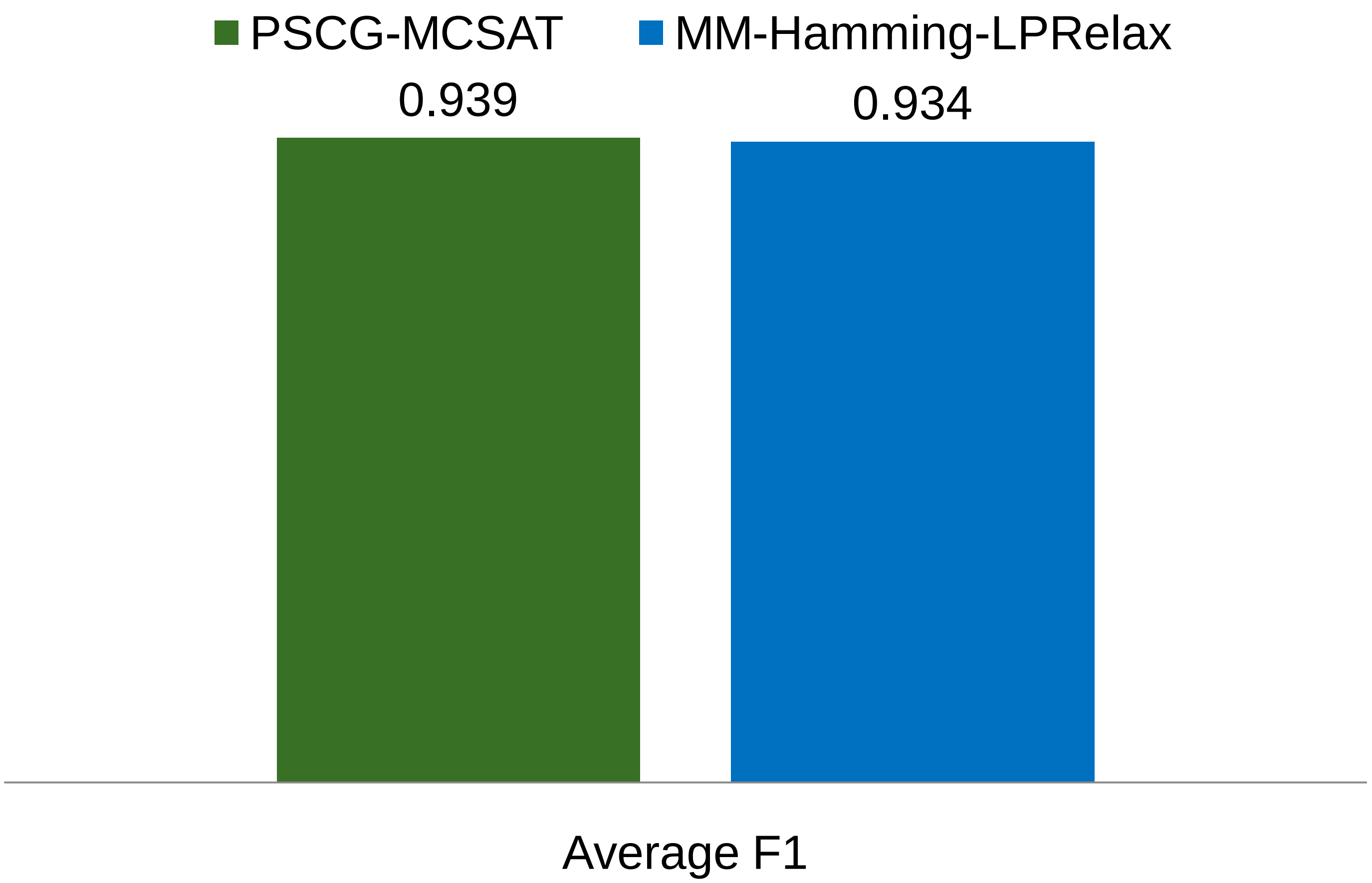
Where does the improvement come from?

- PSCG-LPRelax: run the new LP-relaxation MPE algorithm on the model learnt by PSCG-MCSAT
- MM-Hamming-MCSAT: run the MCSAT inference on the model learnt by MM-Hamming-LPRelax

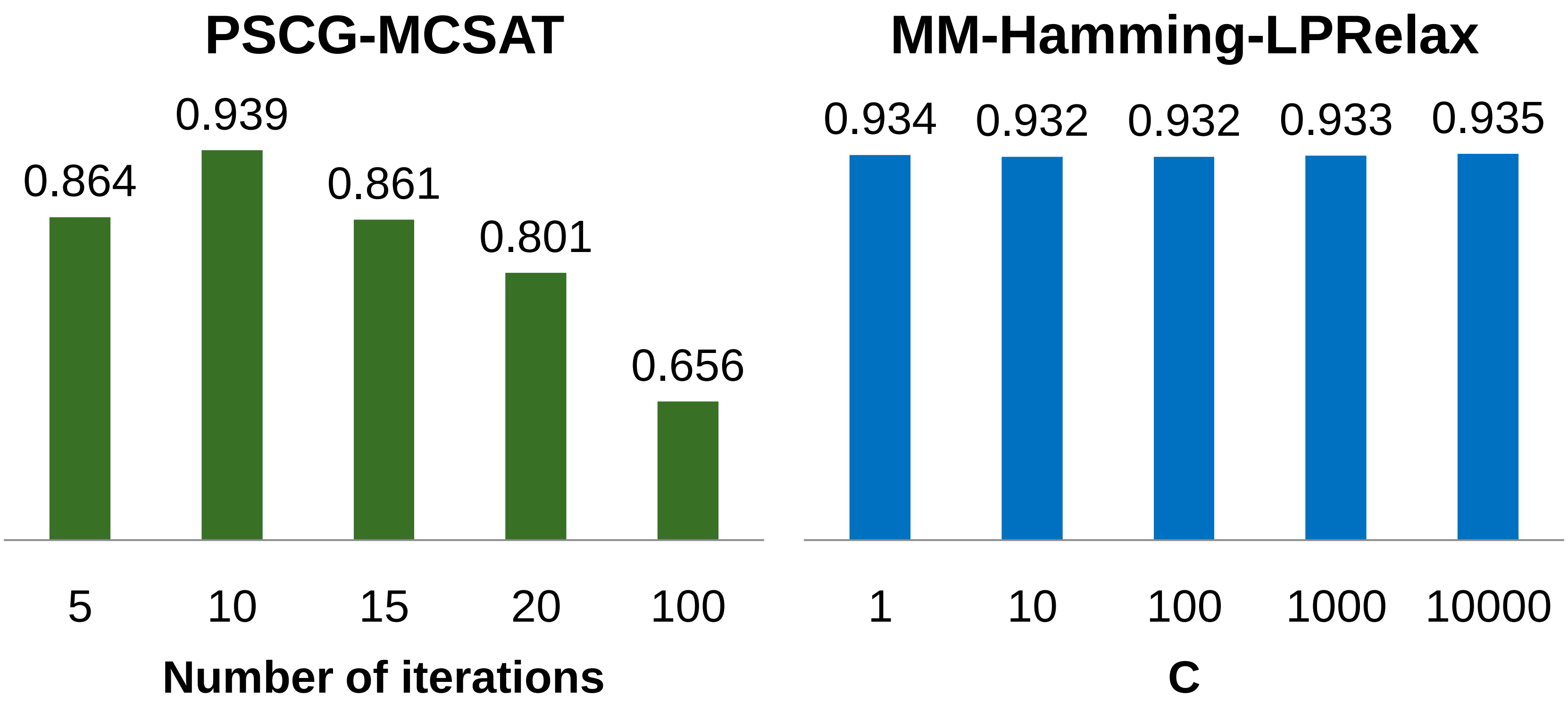
F_1 scores on WebKB(cont.)



F_1 scores on Citeseer



Sensitivity to the tuning parameter



Future work

- Approximation algorithms for optimizing other application specific loss functions
- More efficient inference algorithm
- Online max-margin weight learning
  - 1-best MIRA [Crammer et.al., 2005]
- More experiments on structured prediction and compare to other existing models

Summary

- All existing discriminative weight learners for MLNs try to optimize the CLL
- Proposed a max-margin approach to weight learning in MLNs, which can optimize application specific measures
- Developed a new LP-relaxation MPE inference for MLNs
- The max-margin weight learner achieves better or equally good but more stable performance.

Questions?

Thank you!

Cutting plane algorithm [Joachims et.al., 2009]

QP solver

$$(w,\xi) \leftarrow \arg\min_{w,\xi>0} \frac{1}{2} w^T w + C\xi$$

$$st. \quad \forall(\bar{y}_1,\ldots,\bar{y}_n) \in W : \frac{1}{n} w^T \sum_{i=1}^{n} [n(x_i,y_i) - n(x_i,\bar{y}_i)] \geq \frac{1}{n} \sum_{i=1}^{n} \Delta(y_i,\bar{y}_i) - \xi$$

$$(w,\xi)$$

Separation oracle

for i = 1,.., n do

$$\bar{y}_i \leftarrow \arg\max_{y \in Y} \left\{ \Delta(y_i,y) + w^T n(x_i,y) \right\}$$

end for

The most violated constraint

$$(\bar{y}_1,\ldots\bar{y}_n)$$

$$\frac{1}{n} w^T \sum_{i=1}^{n} [n(x_i,y_i) - n(x_i,\bar{y}_i)] \geq \frac{1}{n} \sum_{i=1}^{n} \Delta(y_i,\bar{y}_i) - \xi - \varepsilon$$

Stopping condition

$$W \leftarrow W \bigcup (\bar{y}_1,\ldots\bar{y}_n)$$