Strong Baselines for Neural Semi-supervised Learning under Domain Shift

Sebastian Ruder

Insight

AYLIEN

Barbara Plank

IT UNIVERSITY OF COPENHAGEN

rijksuniversiteit groningen

Learning under Domain Shift

Learning under Domain Shift

- State-of-the-art domain adaptation approaches

Learning under Domain Shift

- State-of-the-art domain adaptation approaches
  - leverage task-specific features

Learning under Domain Shift

- State-of-the-art domain adaptation approaches
  - leverage task-specific features
  - evaluate on proprietary datasets or on a single benchmark

Learning under Domain Shift

- State-of-the-art domain adaptation approaches
  - leverage task-specific features
  - evaluate on proprietary datasets or on a single benchmark
- Only compare against weak baselines

Learning under Domain Shift

- State-of-the-art domain adaptation approaches
  - leverage task-specific features
  - evaluate on proprietary datasets or on a single benchmark
- Only compare against weak baselines
- Almost none evaluate against approaches from the extensive semi-supervised learning (SSL) literature

Revisiting Semi-Supervised Learning Classics in a Neural World

Revisiting Semi-Supervised Learning Classics in a Neural World

- How do classics in SSL compare to recent advances?

Revisiting Semi-Supervised Learning Classics in a Neural World

- How do classics in SSL compare to recent advances?
- Can we combine the best of both worlds?

Revisiting Semi-Supervised Learning Classics in a Neural World

- How do classics in SSL compare to recent advances?
- Can we combine the best of both worlds?
- How well do these approaches work on out-of-distribution data?

Bootstrapping algorithms

Bootstrapping algorithms

- Self-training

Bootstrapping algorithms

- Self-training
- (Co-training)

Bootstrapping algorithms

- Self-training
- (Co-training)
- Tri-training

Bootstrapping algorithms

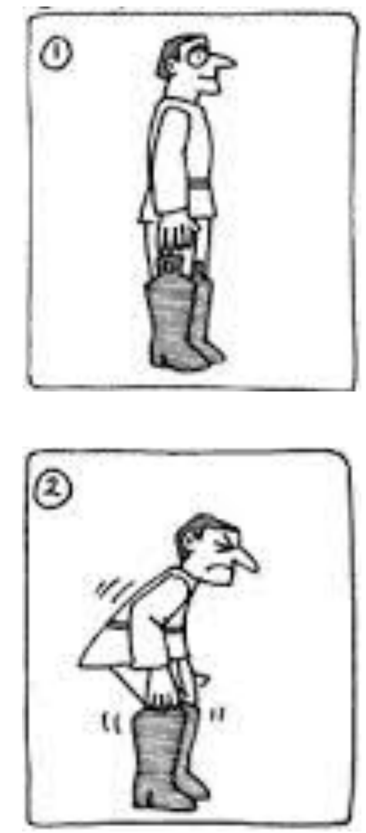
- Self-training
- (Co-training)
- Tri-training
- Tri-training with disagreement

Self-training

Self-training

1. Train model on labeled data.

Self-training



1. Train model on labeled data.
2. Use confident predictions on unlabeled data as training examples. Repeat.

Self-training

1. Train model on labeled data.
2. Use confident predictions on unlabeled data as training examples. Repeat.

- Error amplification

Self-training

1. Train model on labeled data.
2. Use confident predictions on unlabeled data as training examples. Repeat.

– Error amplification

Self-training variants

Self-training variants

- **Calibration**

Self-training variants

- **Calibration**
  - Output probabilities in neural networks are poorly calibrated.

Self-training variants

- **Calibration**
  - Output probabilities in neural networks are poorly calibrated.
  - Throttling (Abney, 2007), i.e. selecting the top *n* highest confidence unlabeled examples works best.

Self-training variants

- **Calibration**
  - Output probabilities in neural networks are poorly calibrated.
  - Throttling (Abney, 2007), i.e. selecting the top *n* highest confidence unlabeled examples works best.
- **Online learning**

Self-training variants

- **Calibration**
  - Output probabilities in neural networks are poorly calibrated.
  - Throttling (Abney, 2007), i.e. selecting the top *n* highest confidence unlabeled examples works best.
- **Online learning**
  - Training until convergence on labeled data and then on unlabeled data works best.

Tri-training

Tri-training

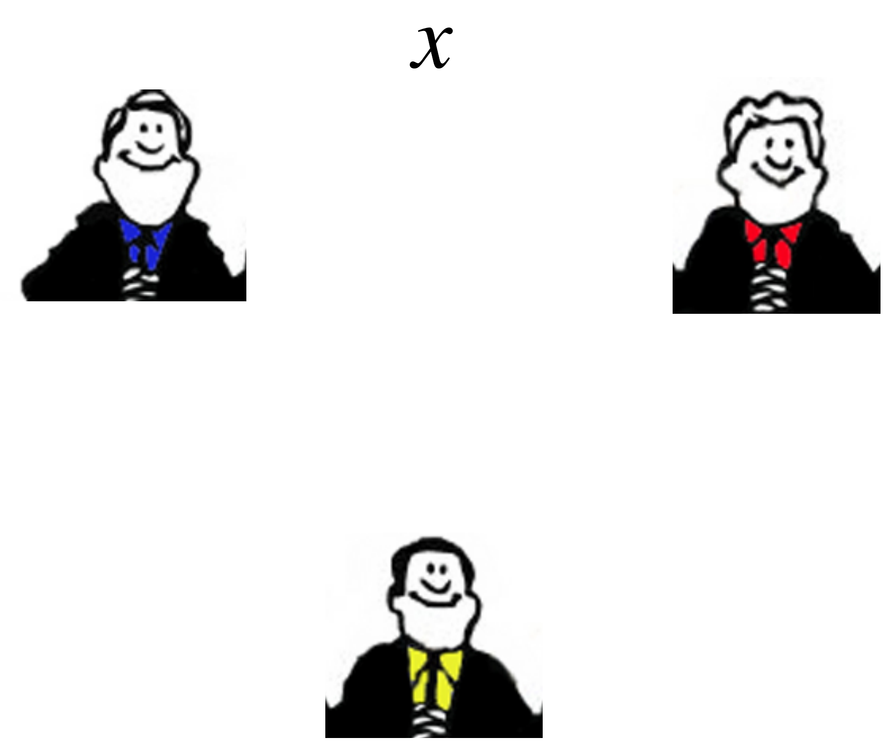
1. Train three models on bootstrapped samples.

Tri-training

1. Train three models on bootstrapped samples.
2. Use predictions on unlabeled data for third if two agree.

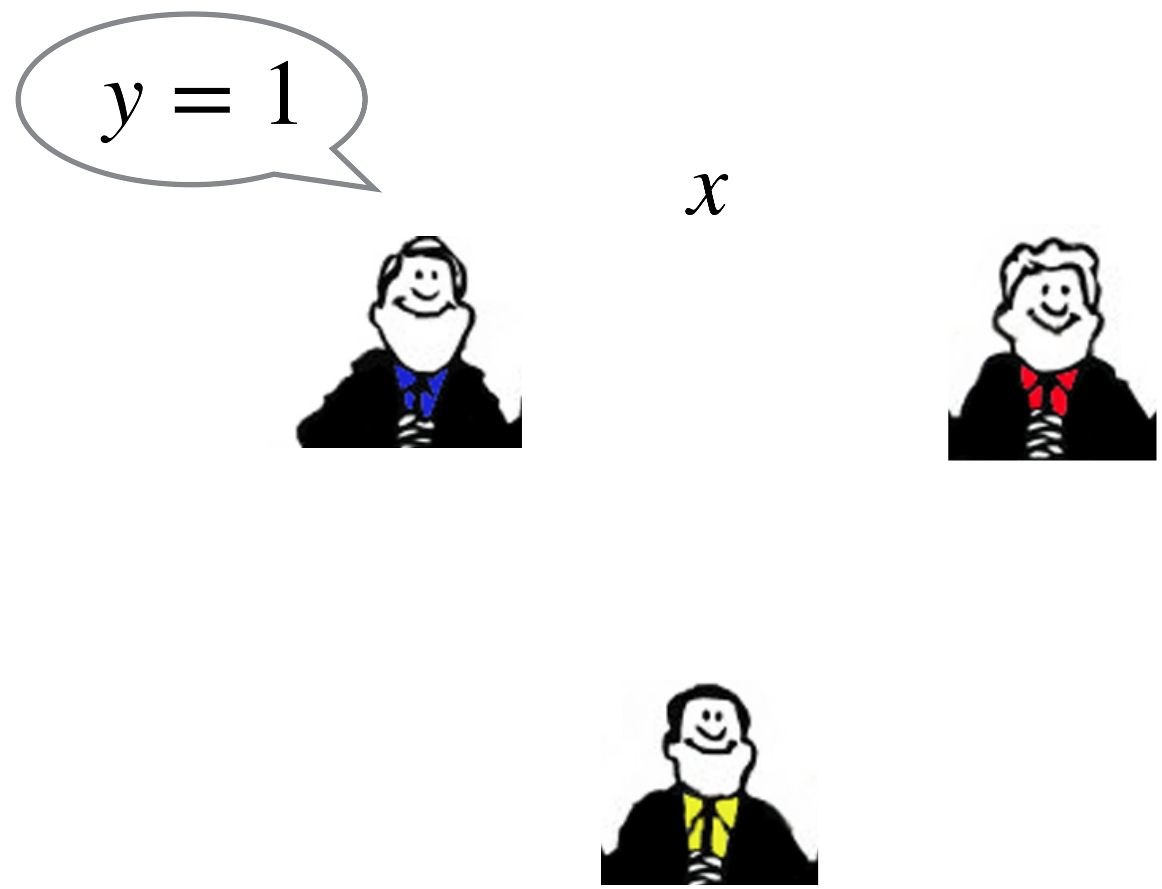
Tri-training

1. Train three models on bootstrapped samples.
2. Use predictions on unlabeled data for third if two agree.



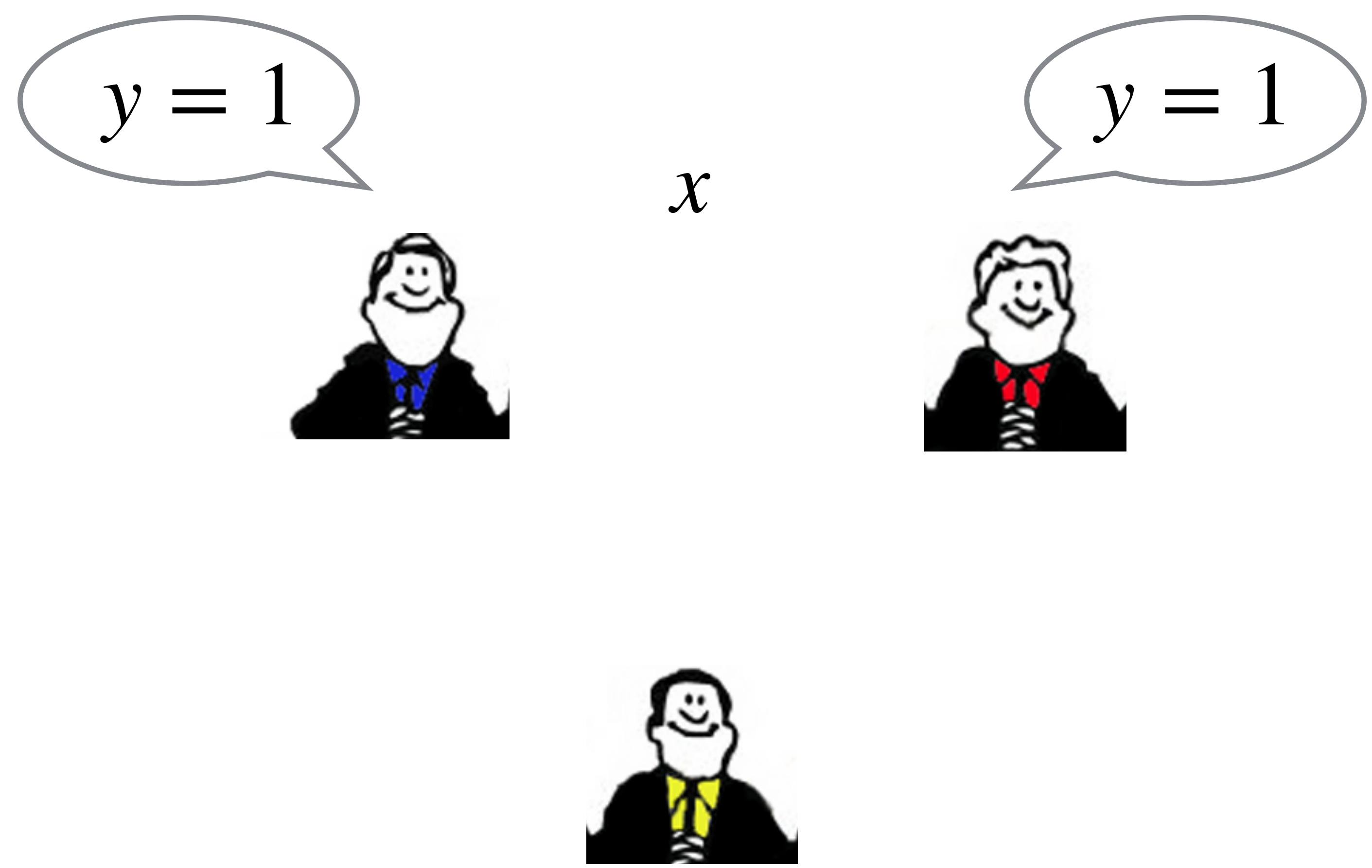
Tri-training

1. Train three models on bootstrapped samples.
2. Use predictions on unlabeled data for third if two agree.



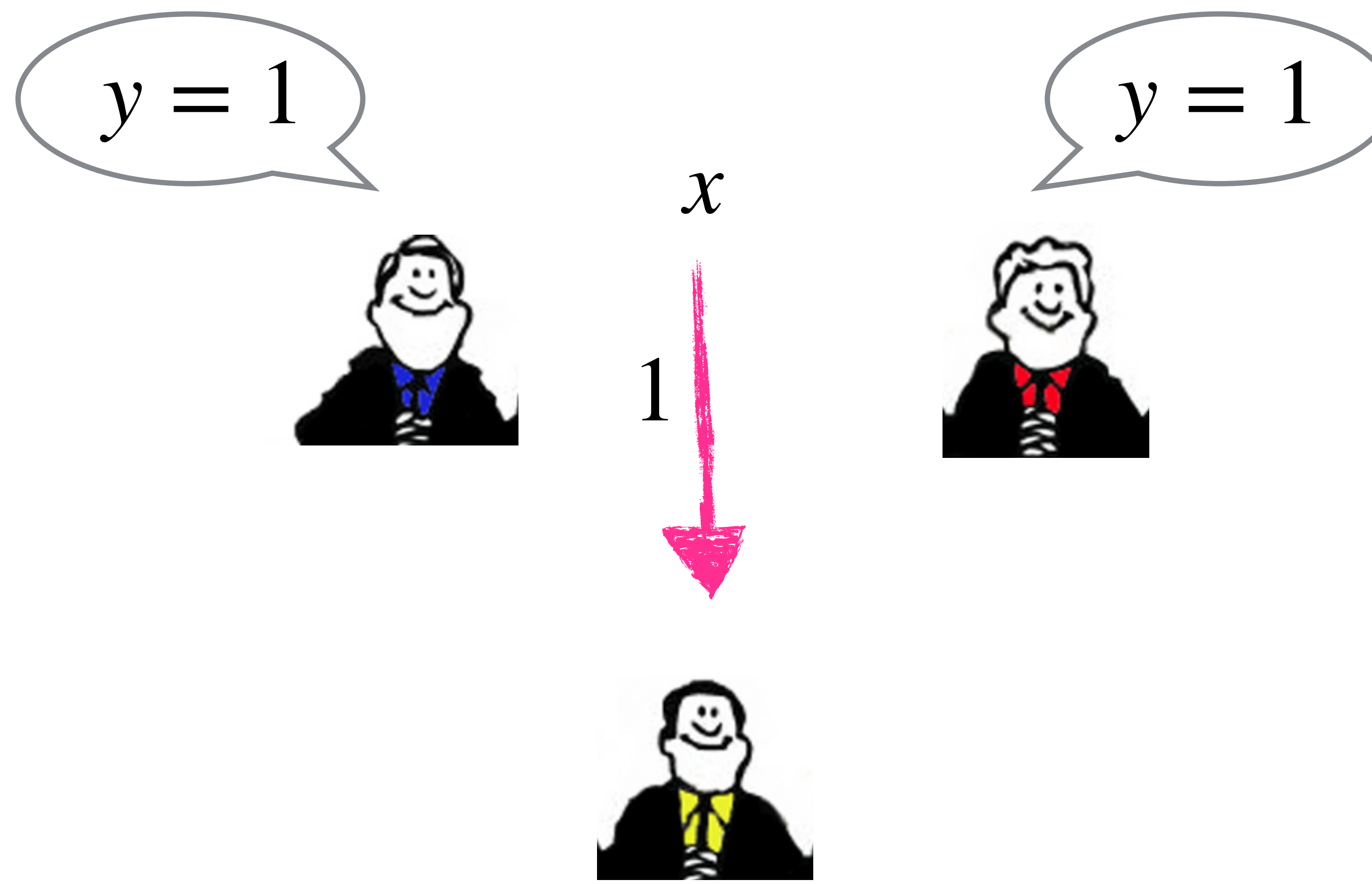
Tri-training

1. Train three models on bootstrapped samples.
2. Use predictions on unlabeled data for third if two agree.



Tri-training

1. Train three models on bootstrapped samples.
2. Use predictions on unlabeled data for third if two agree.



Tri-training

1. Train three models on bootstrapped samples.
2. Use predictions on unlabeled data for third if two agree.
3. Final prediction: majority voting

Tri-training

1. Train three models on bootstrapped samples.
2. Use predictions on unlabeled data for third if two agree.
3. Final prediction: majority voting

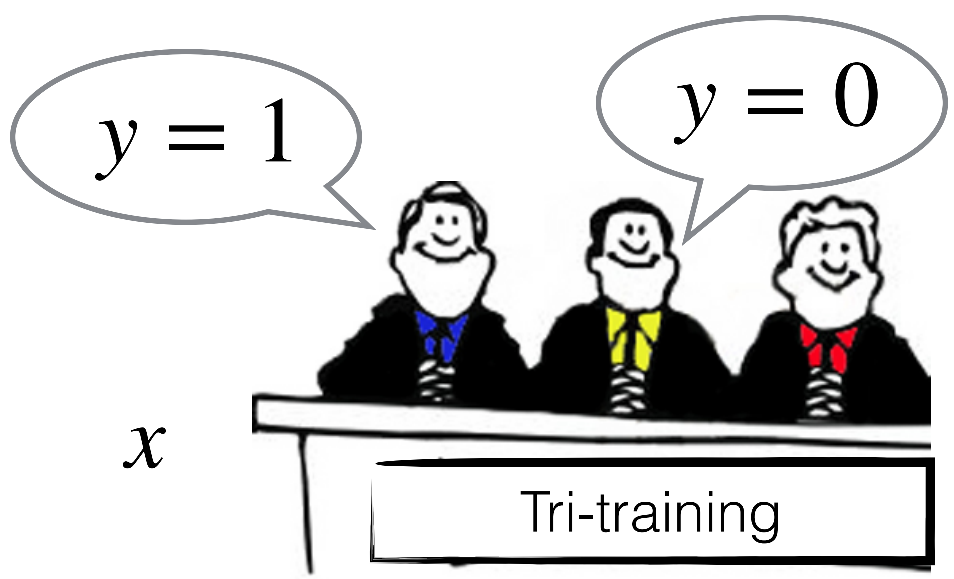
x

Tri-training

1. Train three models on bootstrapped samples.
2. Use predictions on unlabeled data for third if two agree.
3. Final prediction: majority voting

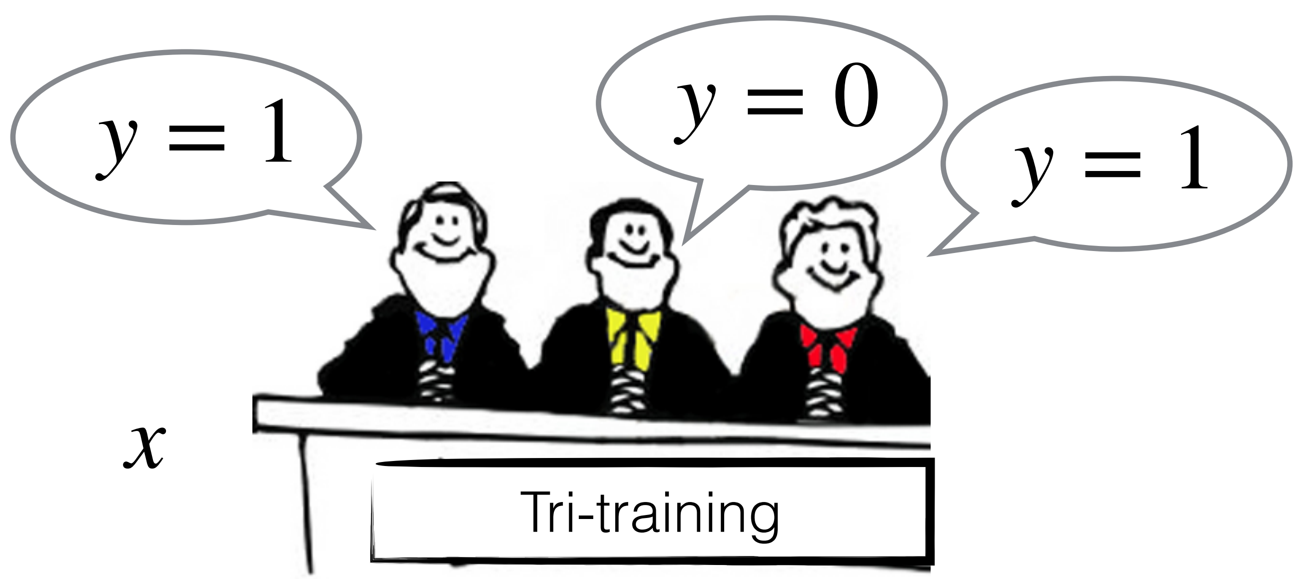
Tri-training

1. Train three models on bootstrapped samples.
2. Use predictions on unlabeled data for third if two agree.
3. Final prediction: majority voting



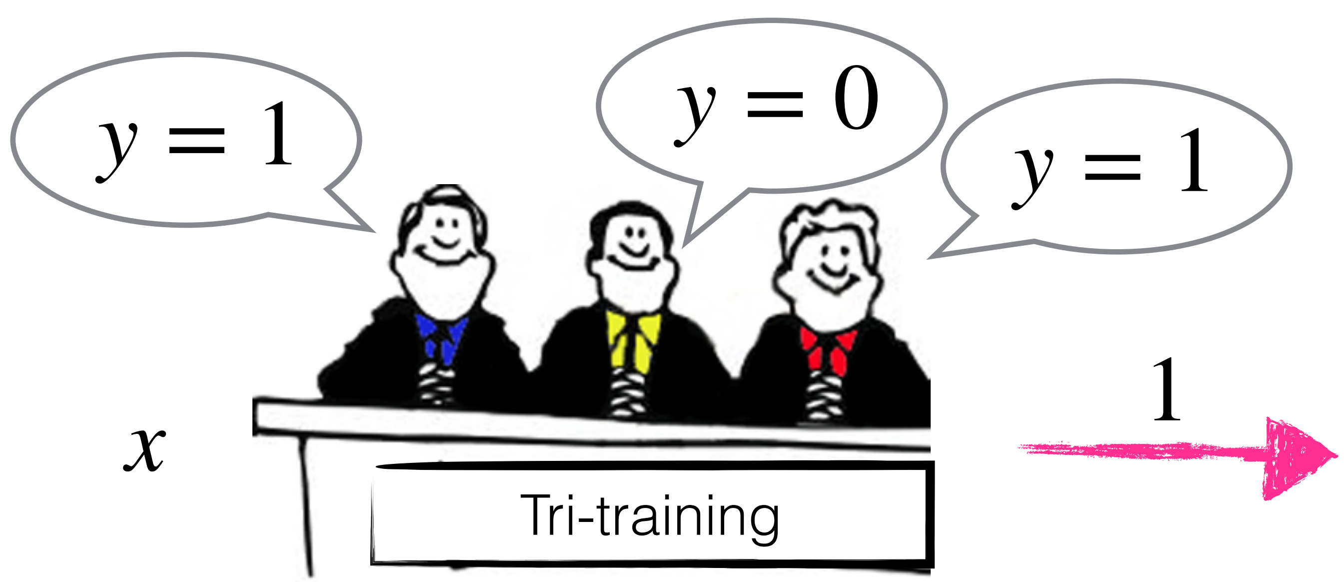
Tri-training

1. Train three models on bootstrapped samples.
2. Use predictions on unlabeled data for third if two agree.
3. Final prediction: majority voting



Tri-training

1. Train three models on bootstrapped samples.
2. Use predictions on unlabeled data for third if two agree.
3. Final prediction: majority voting



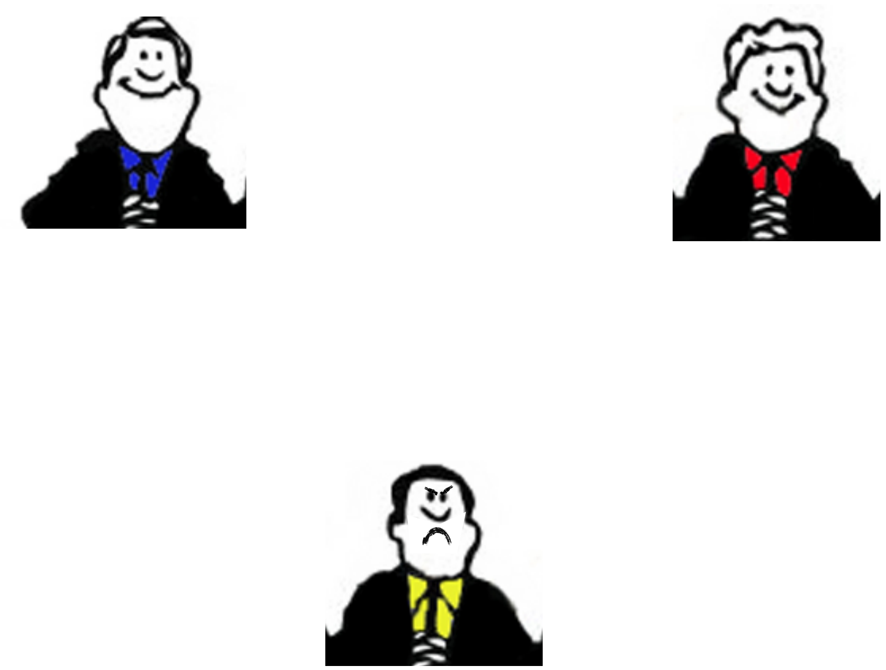
Tri-training with disagreement

Tri-training with disagreement

1. Train three models on bootstrapped samples.

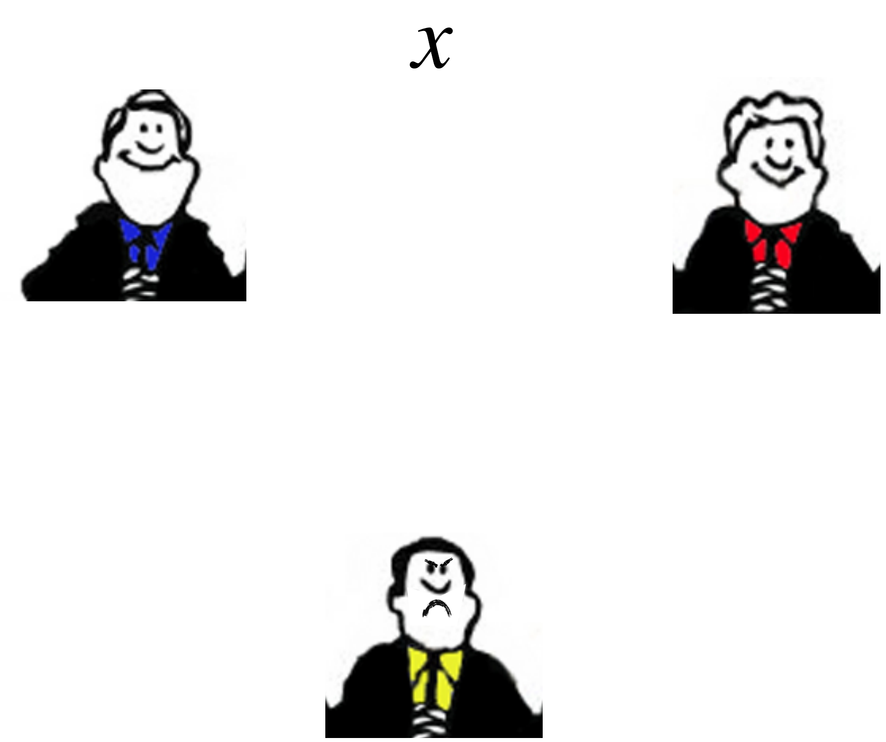
Tri-training with disagreement

1. Train three models on bootstrapped samples.
2. Use predictions on unlabeled data for third if two agree and prediction differs.



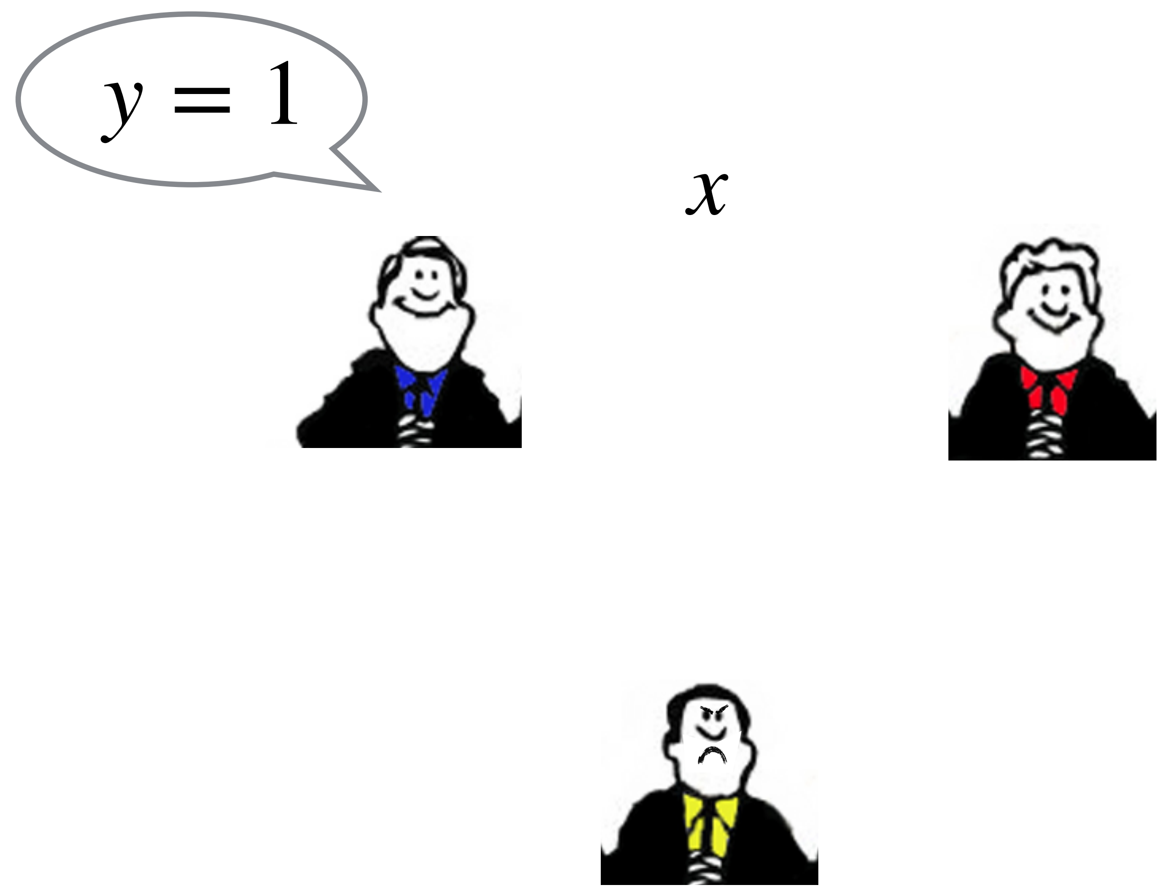
Tri-training with disagreement

1. Train three models on bootstrapped samples.
2. Use predictions on unlabeled data for third if two agree and prediction differs.



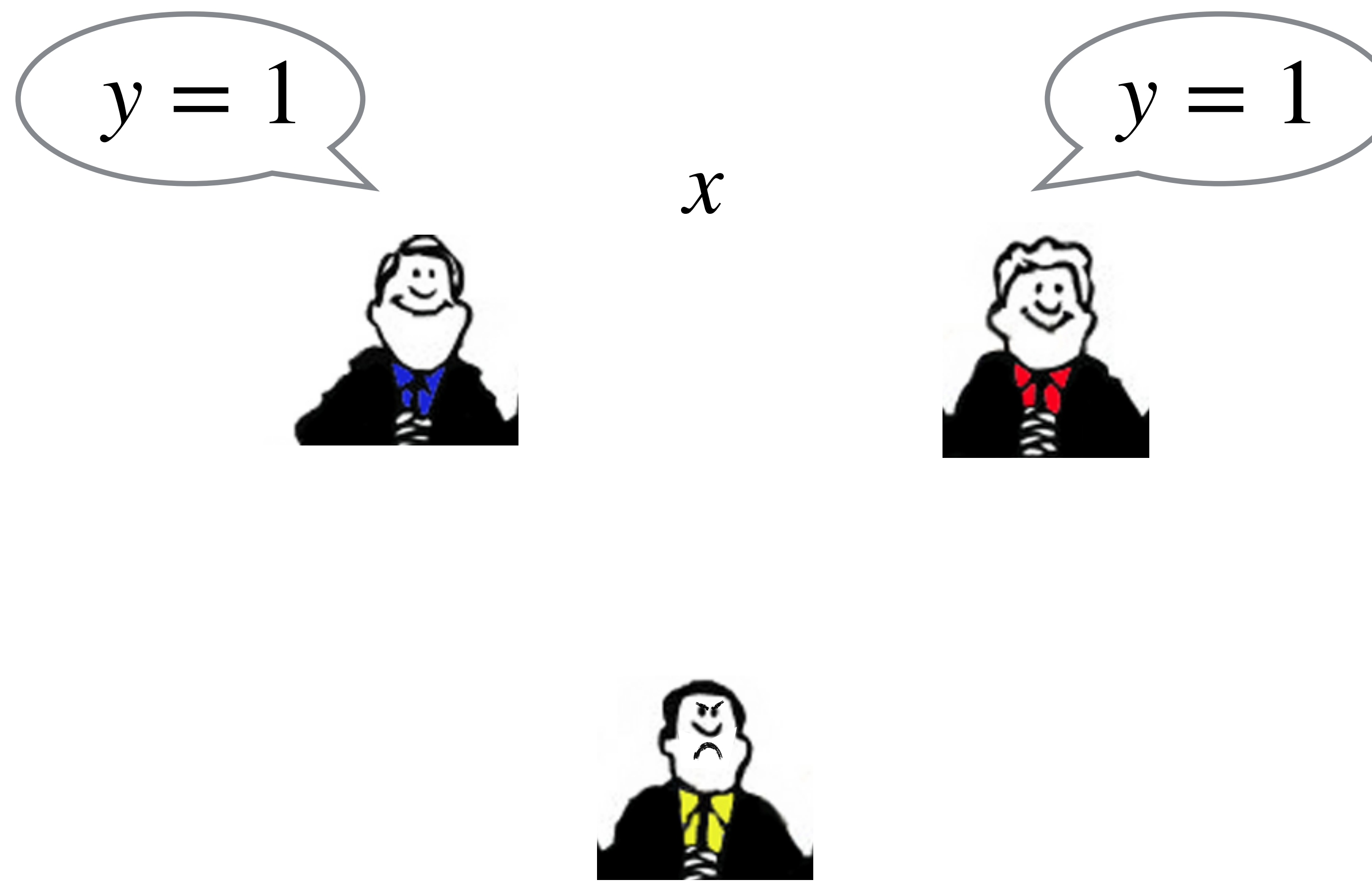
Tri-training with disagreement

1. Train three models on bootstrapped samples.
2. Use predictions on unlabeled data for third if two agree and prediction differs.



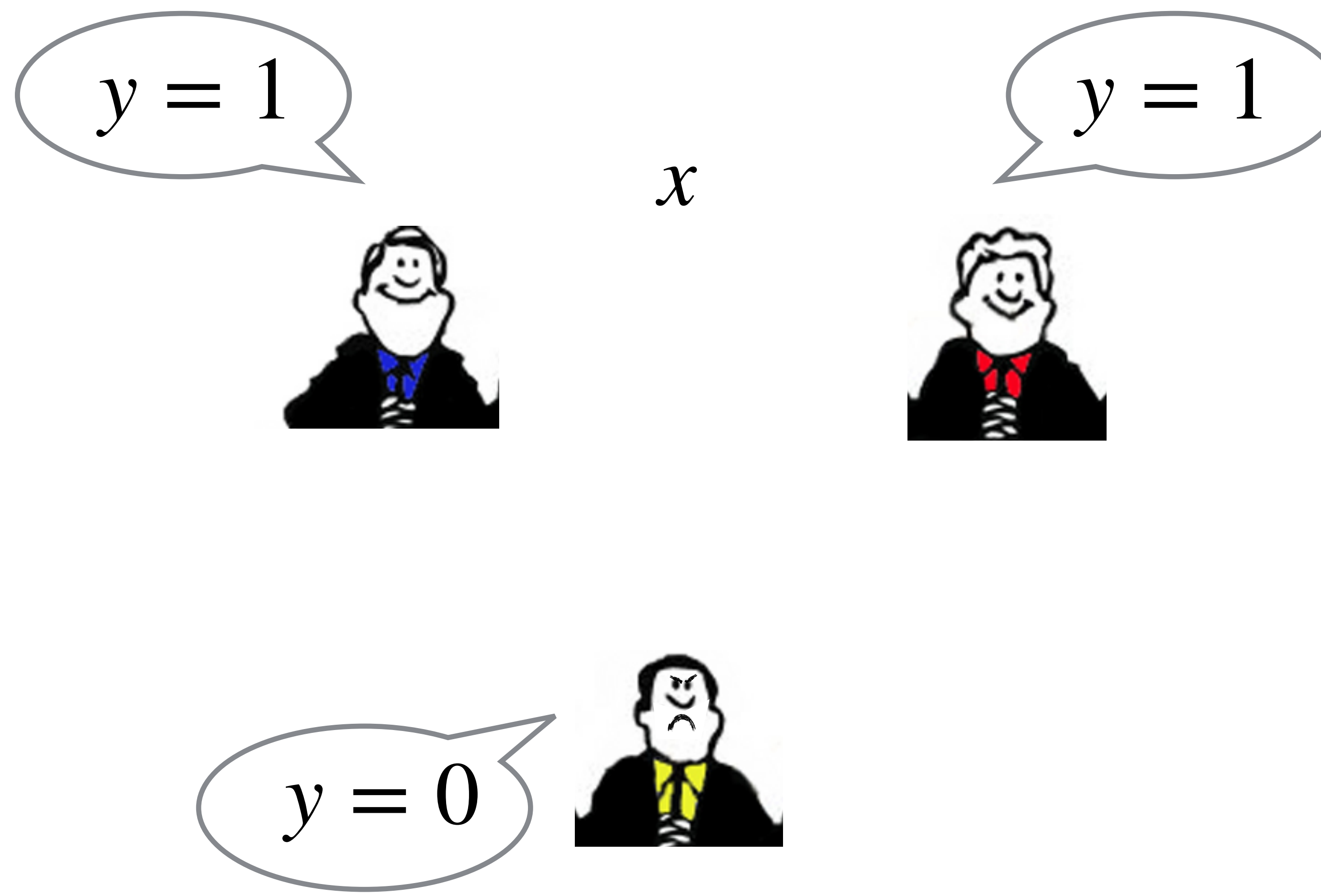
Tri-training with disagreement

1. Train three models on bootstrapped samples.
2. Use predictions on unlabeled data for third if two agree and prediction differs.



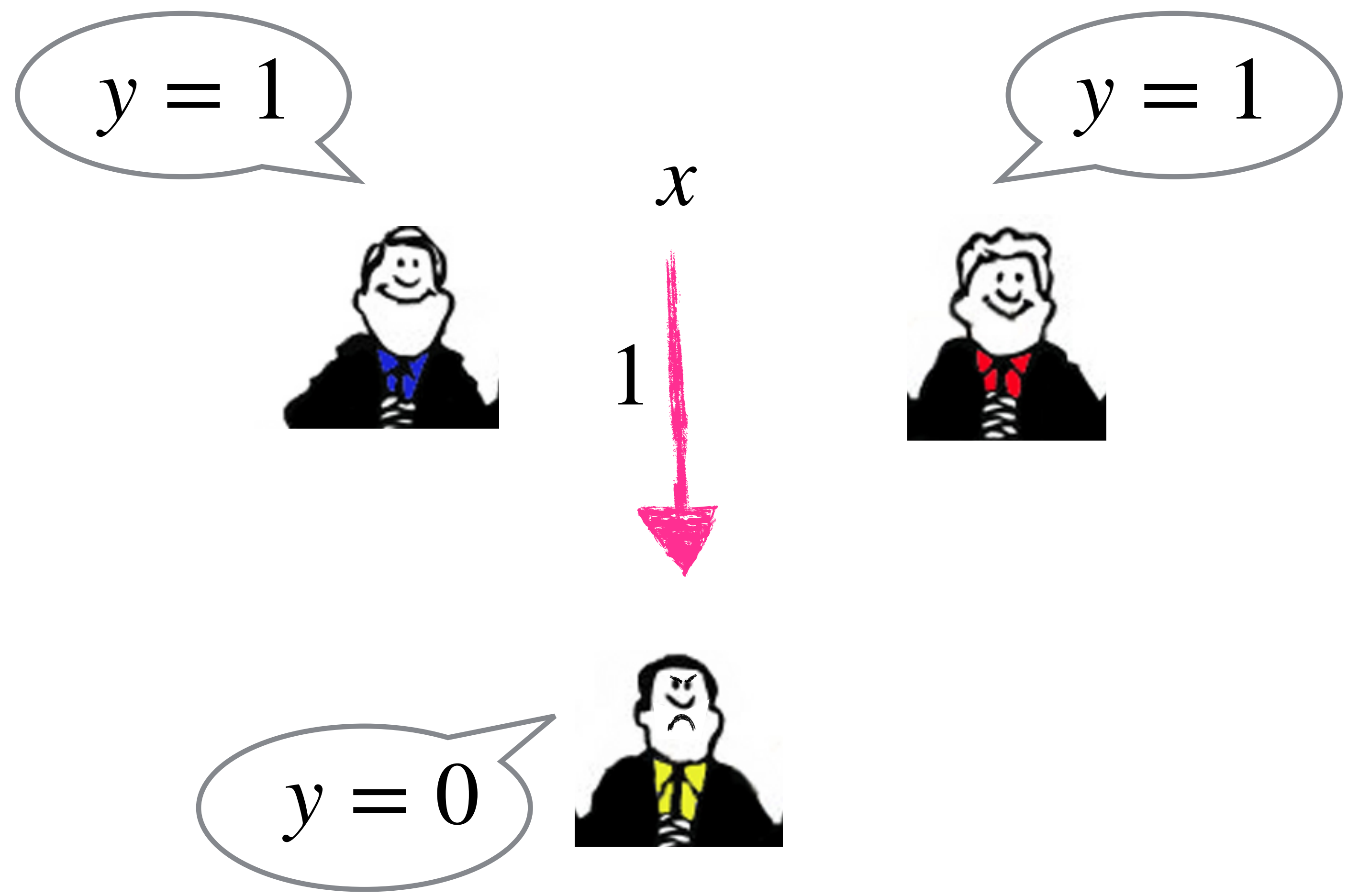
Tri-training with disagreement

1. Train three models on bootstrapped samples.
2. Use predictions on unlabeled data for third if two agree and prediction differs.



Tri-training with disagreement

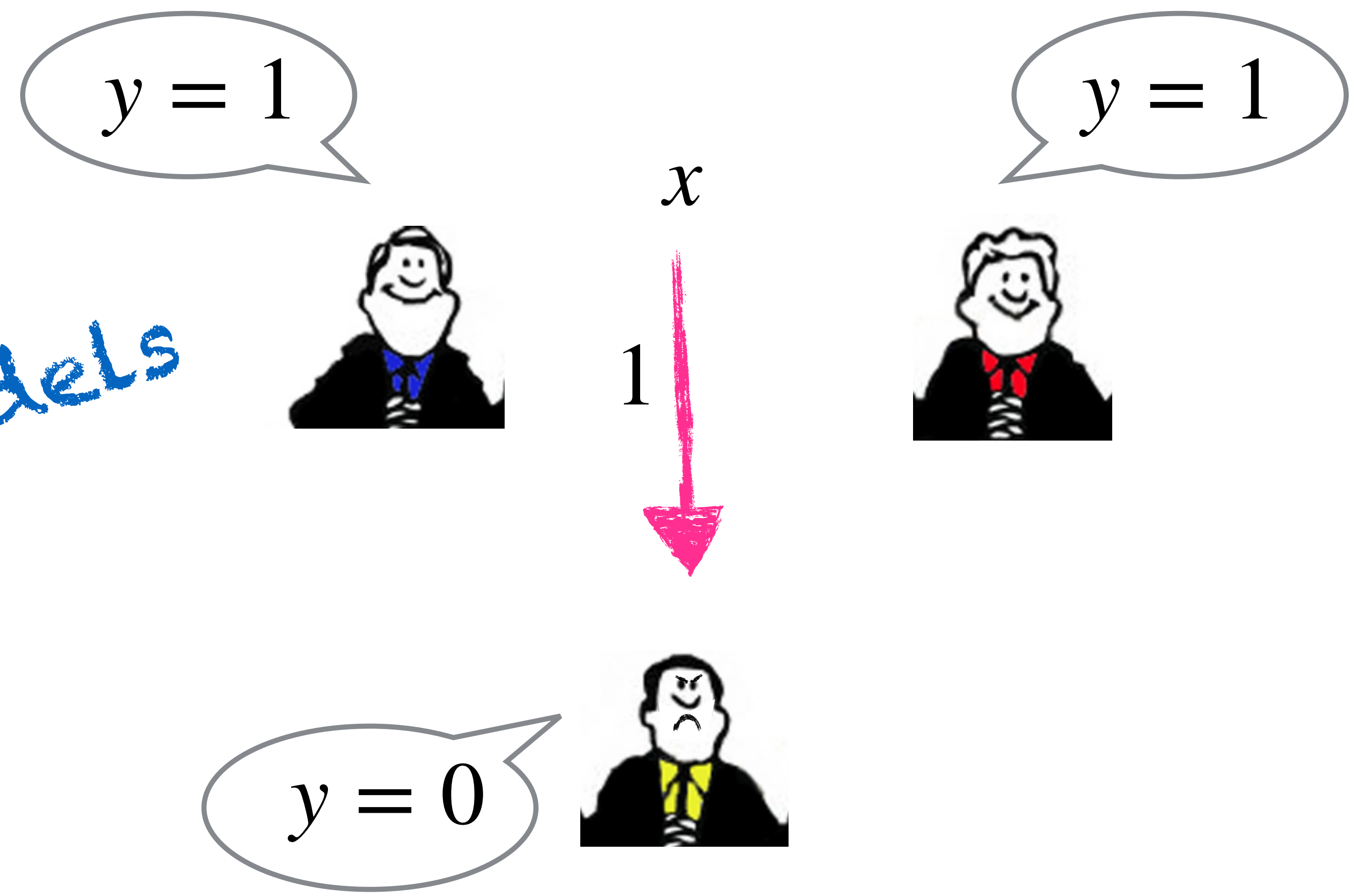
1. Train three models on bootstrapped samples.
2. Use predictions on unlabeled data for third if two agree and prediction differs.



Tri-training with disagreement

1. Train three models on bootstrapped samples.
2. Use predictions on unlabeled data for third if two agree and prediction differs.

- 3 independent models



Tri-training hyper-parameters

Tri-training hyper-parameters

- Sampling unlabeled data

Tri-training hyper-parameters

- **Sampling unlabeled data**
  - Producing predictions for all unlabeled examples is expensive

Tri-training hyper-parameters

- **Sampling unlabeled data**
  - Producing predictions for all unlabeled examples is expensive
  - Sample number of unlabeled examples

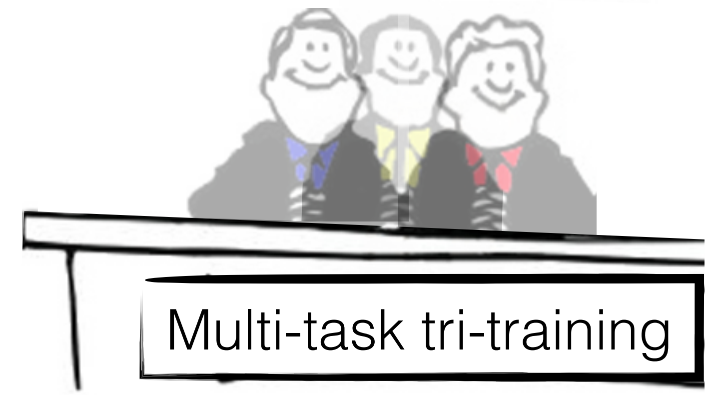
Tri-training hyper-parameters

- **Sampling unlabeled data**
  - Producing predictions for all unlabeled examples is expensive
  - Sample number of unlabeled examples
- **Confidence thresholding**

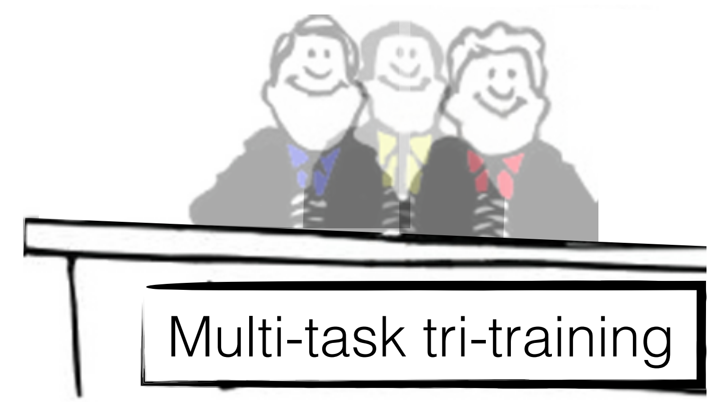
Tri-training hyper-parameters

- **Sampling unlabeled data**
  - Producing predictions for all unlabeled examples is expensive
  - Sample number of unlabeled examples
- **Confidence thresholding**
  - Not effective for classic approaches, but essential for our method

Multi-task Tri-training

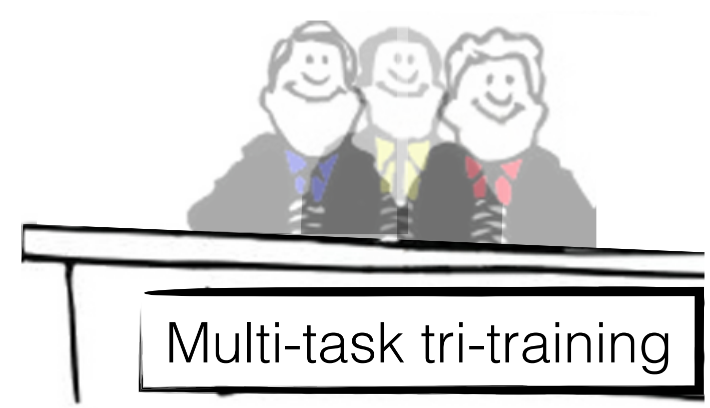


Multi-task Tri-training



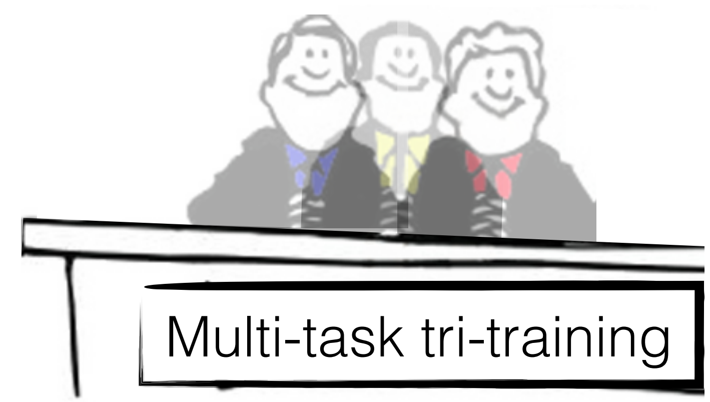
1. Train one model with 3 objective functions.

Multi-task Tri-training

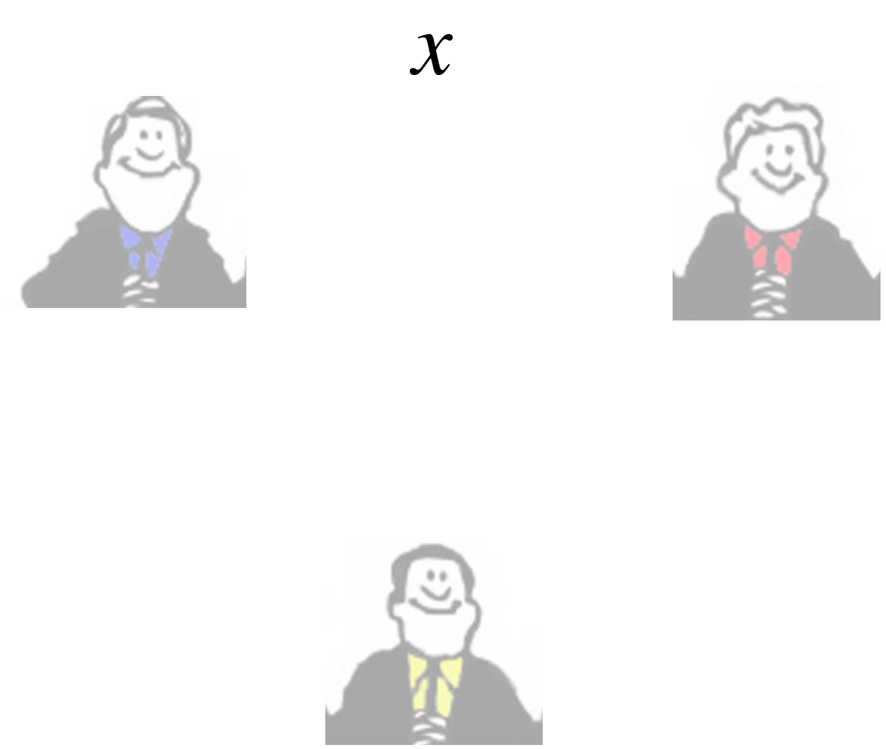


1. Train one model with 3 objective functions.
2. Use predictions on unlabeled data for third if two agree.

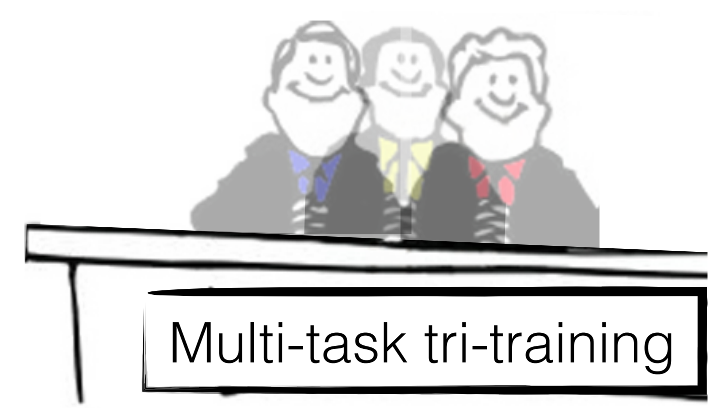
Multi-task Tri-training



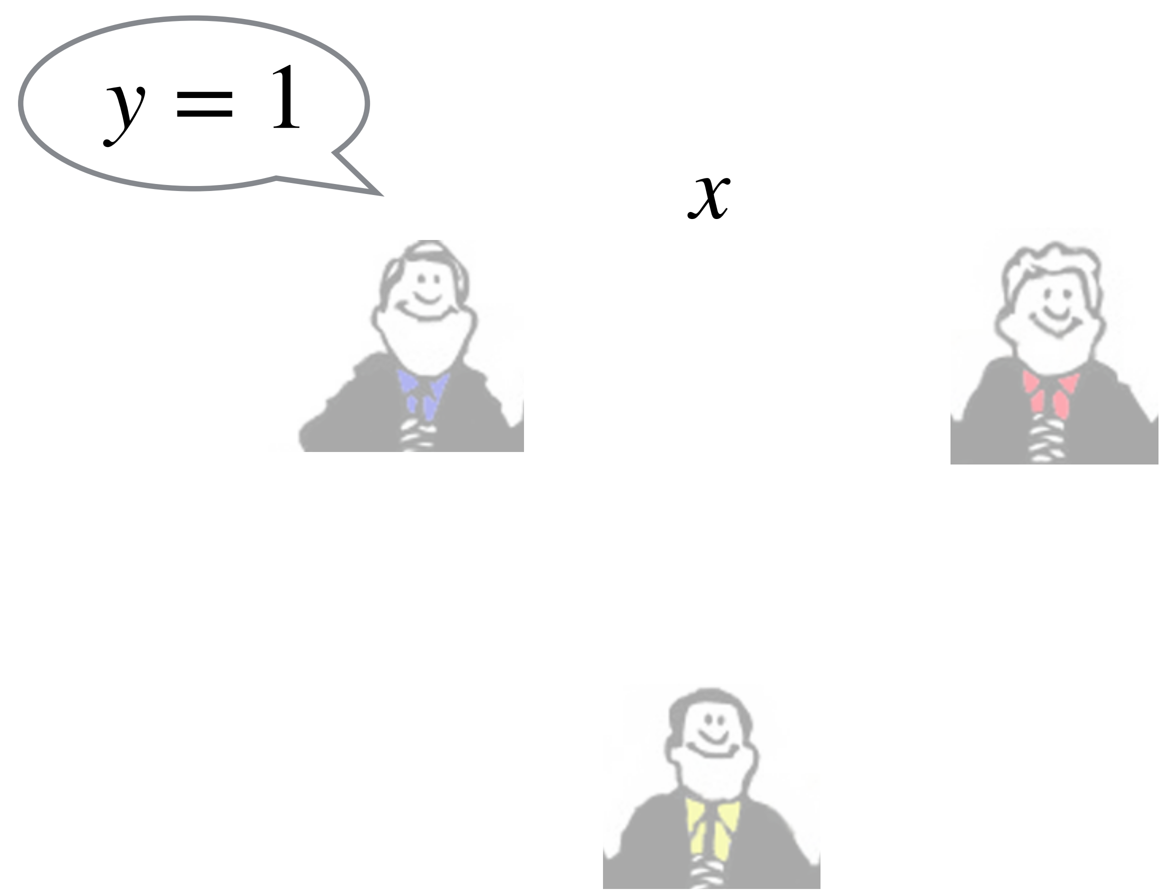
1. Train one model with 3 objective functions.
2. Use predictions on unlabeled data for third if two agree.



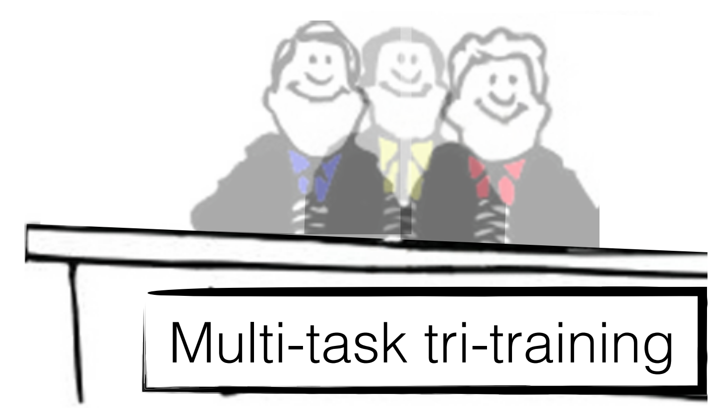
Multi-task Tri-training



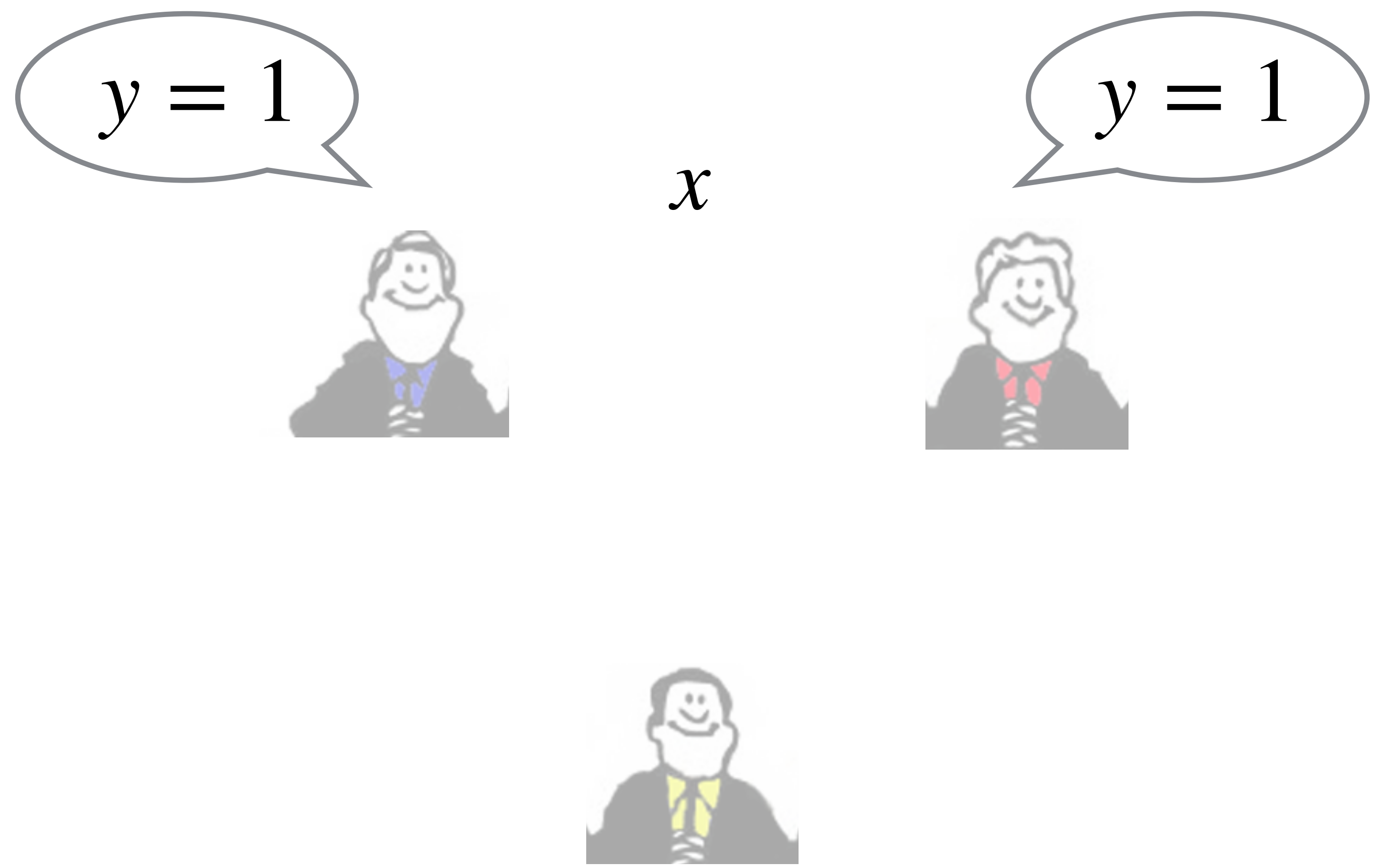
1. Train one model with 3 objective functions.
2. Use predictions on unlabeled data for third if two agree.



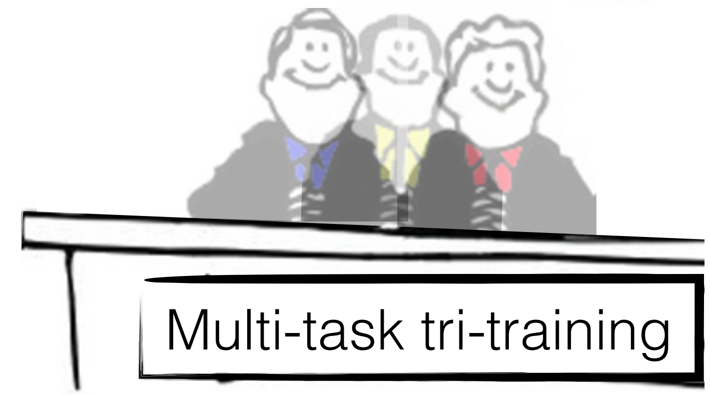
Multi-task Tri-training



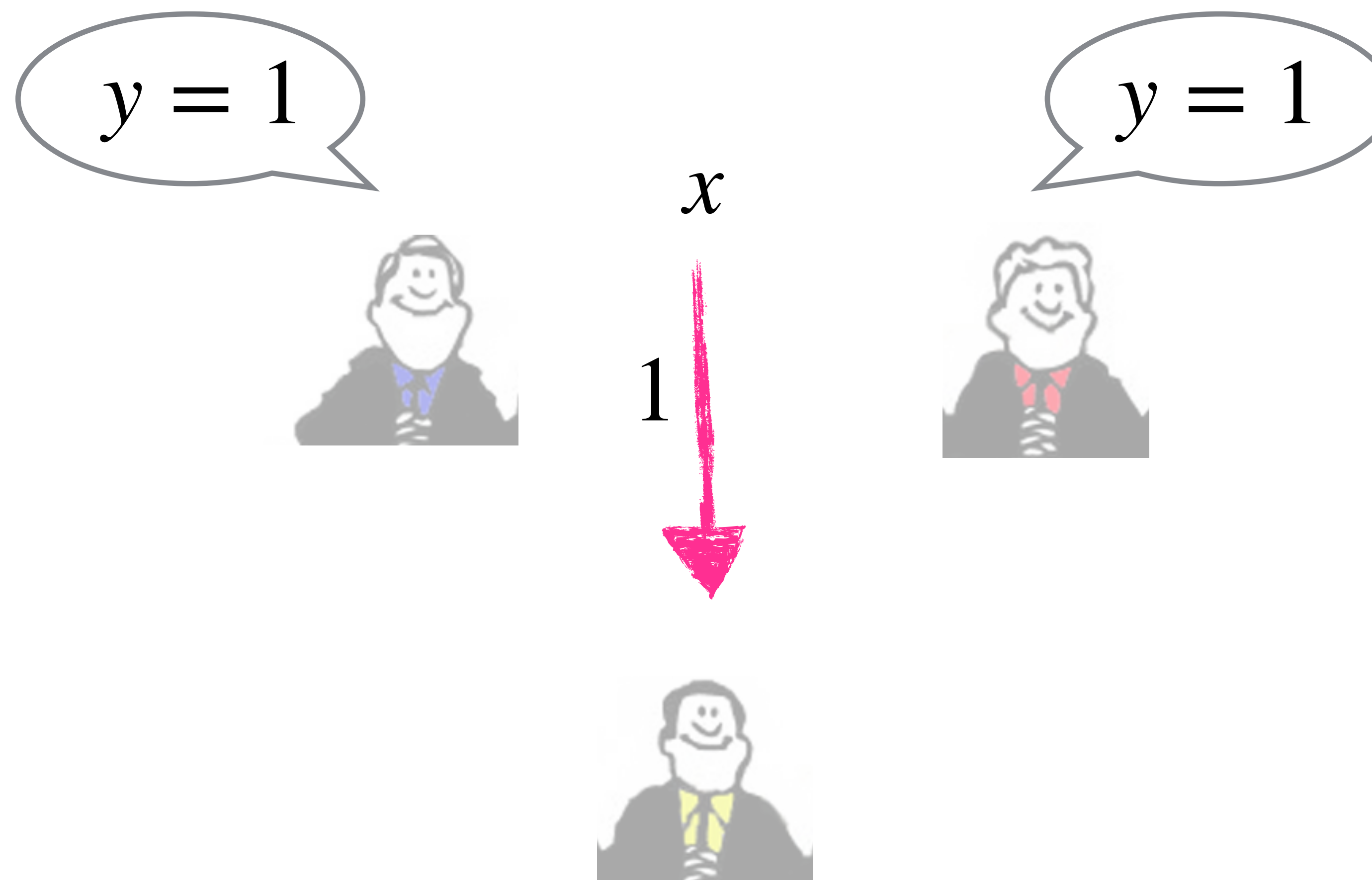
1. Train one model with 3 objective functions.
2. Use predictions on unlabeled data for third if two agree.



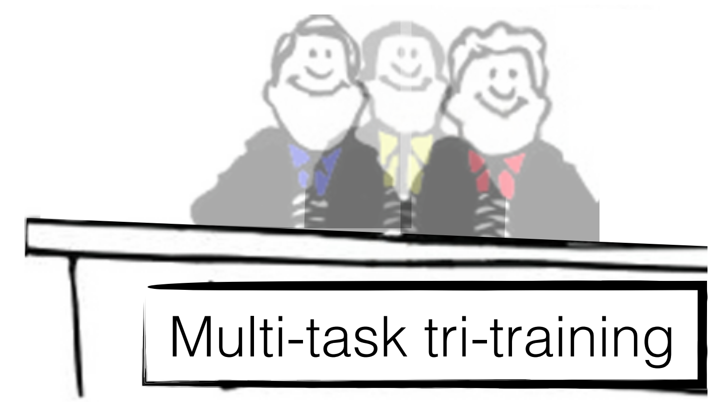
Multi-task Tri-training



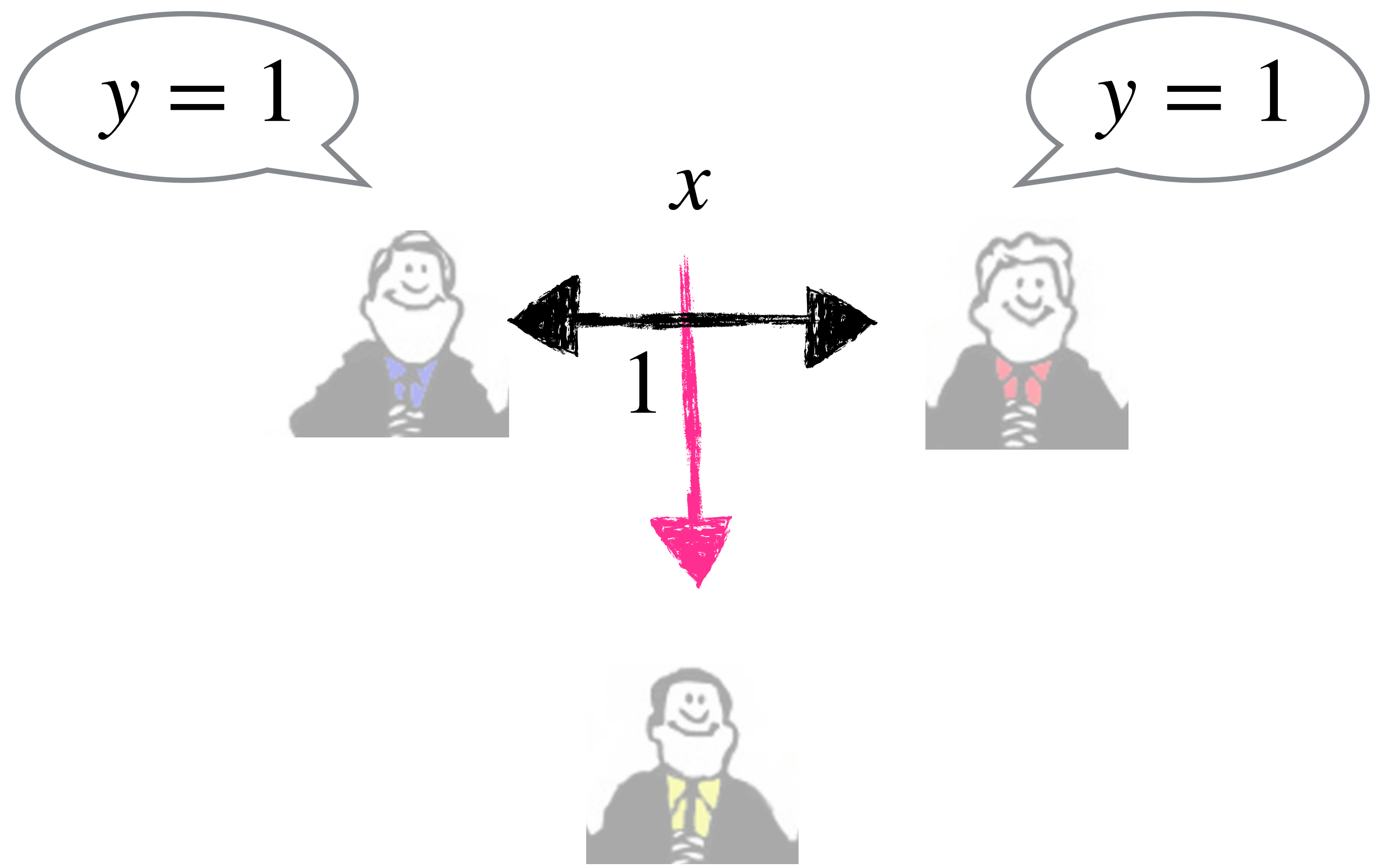
1. Train one model with 3 objective functions.
2. Use predictions on unlabeled data for third if two agree.



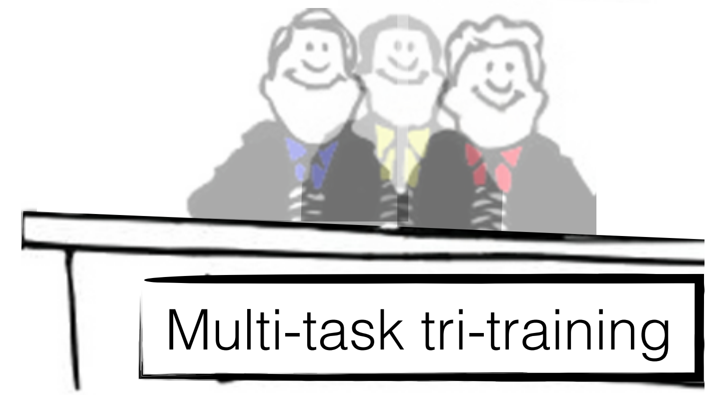
Multi-task Tri-training



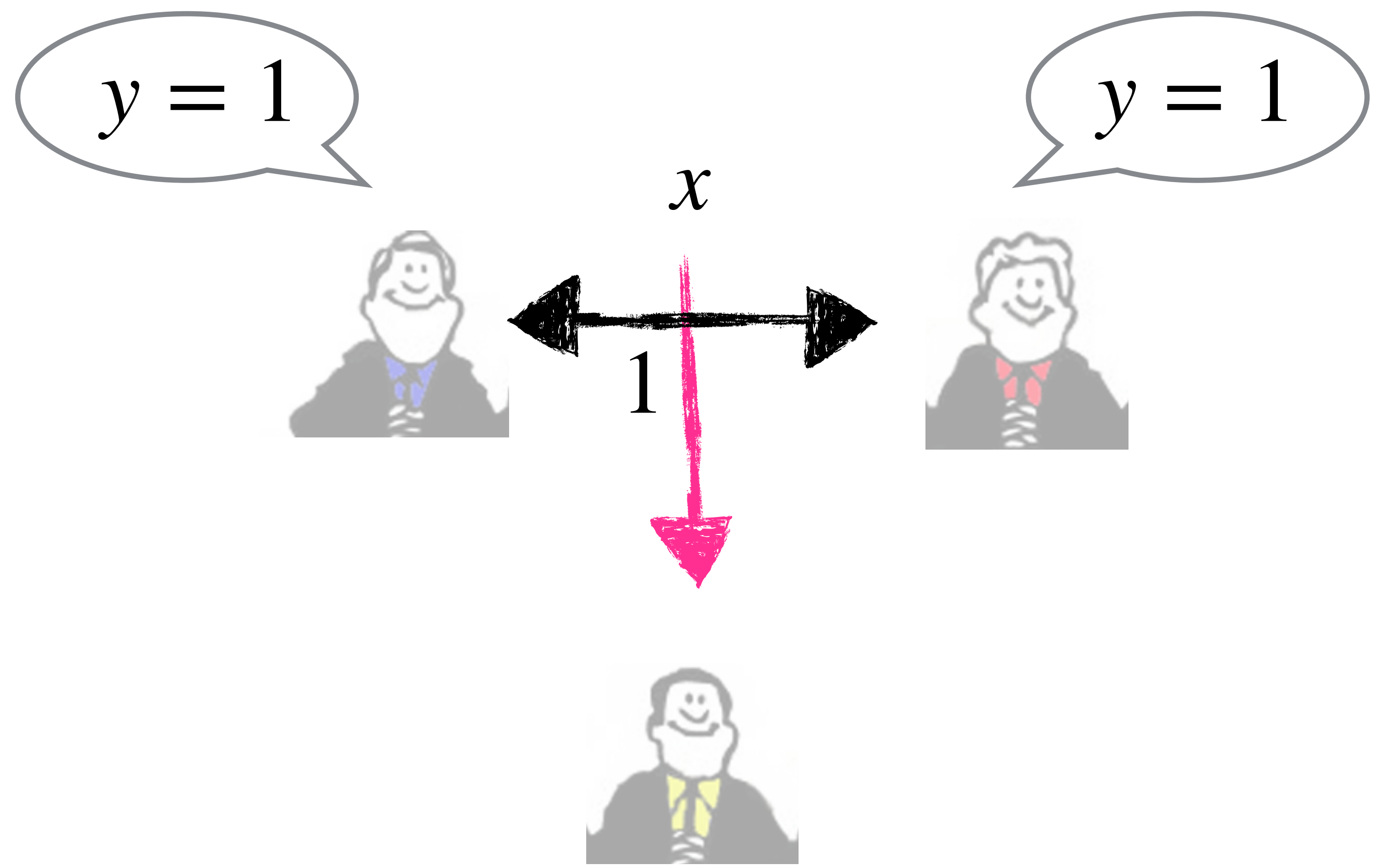
1. Train one model with 3 objective functions.
2. Use predictions on unlabeled data for third if two agree.
3. Restrict final layers to use different representations.



Multi-task Tri-training



1. Train one model with 3 objective functions.
2. Use predictions on unlabeled data for third if two agree.
3. Restrict final layers to use different representations.
4. Train third objective function only on pseudo labeled to bridge domain shift.



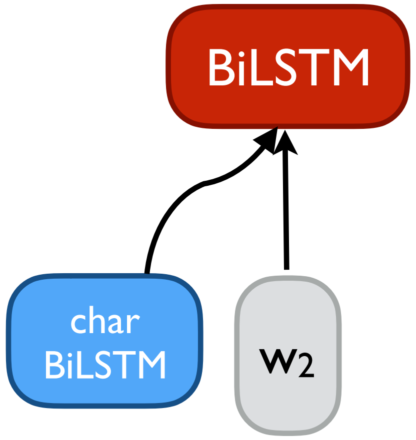
Multi-task
Tri-training

Multi-task Tri-training

BiLSTM

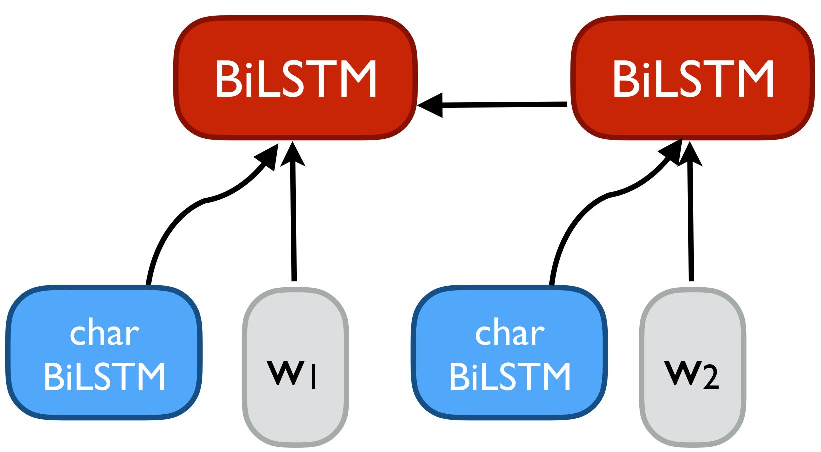
(Plank et al., 2016)

Multi-task
Tri-training



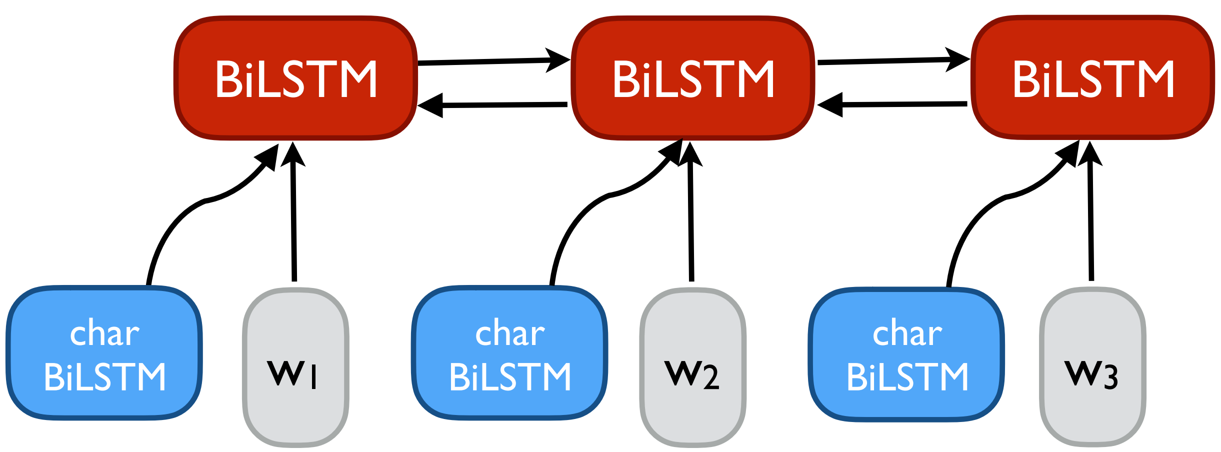
(Plank et al., 2016)

Multi-task Tri-training



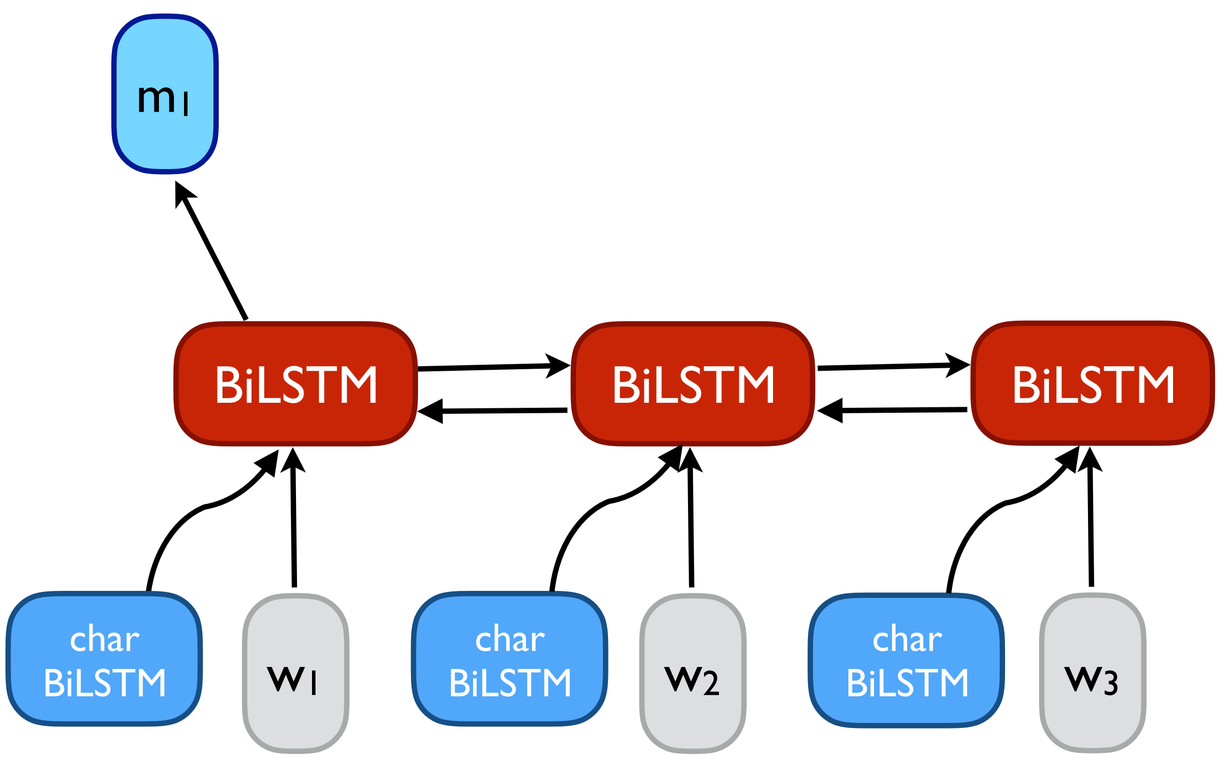
(Plank et al., 2016)

Multi-task Tri-training



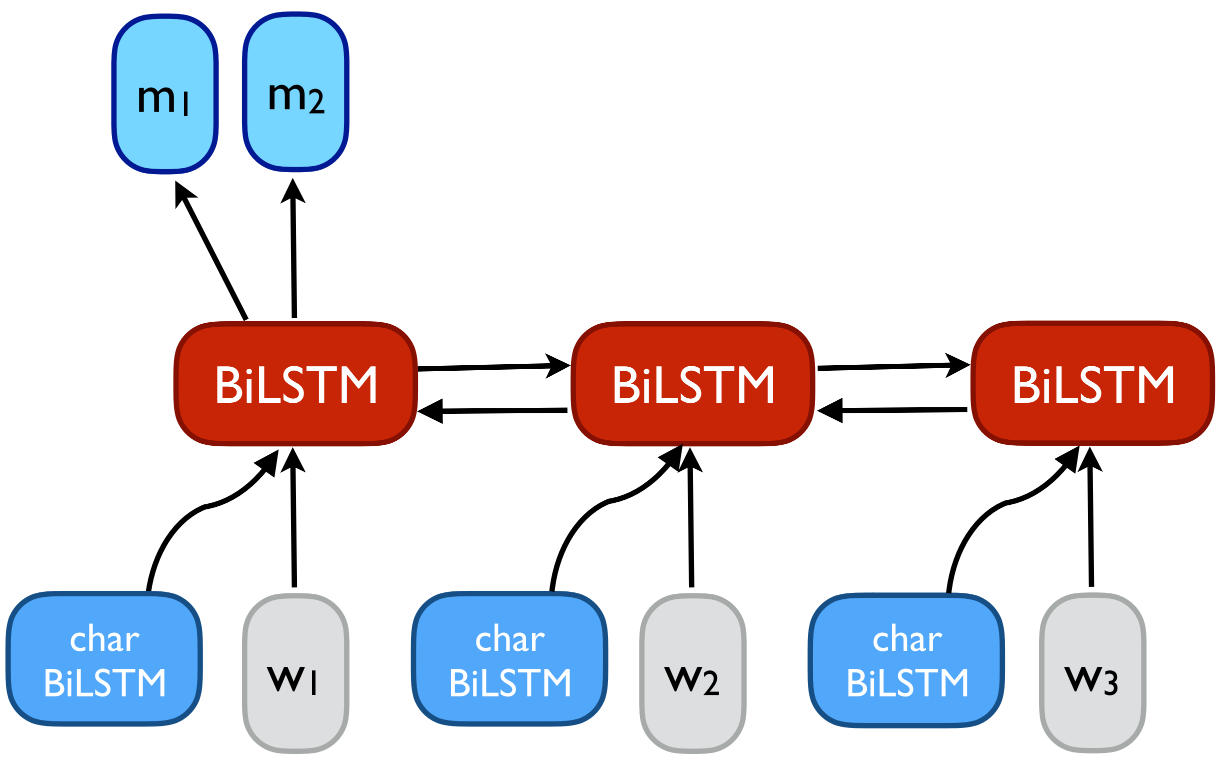
(Plank et al., 2016)

Multi-task Tri-training



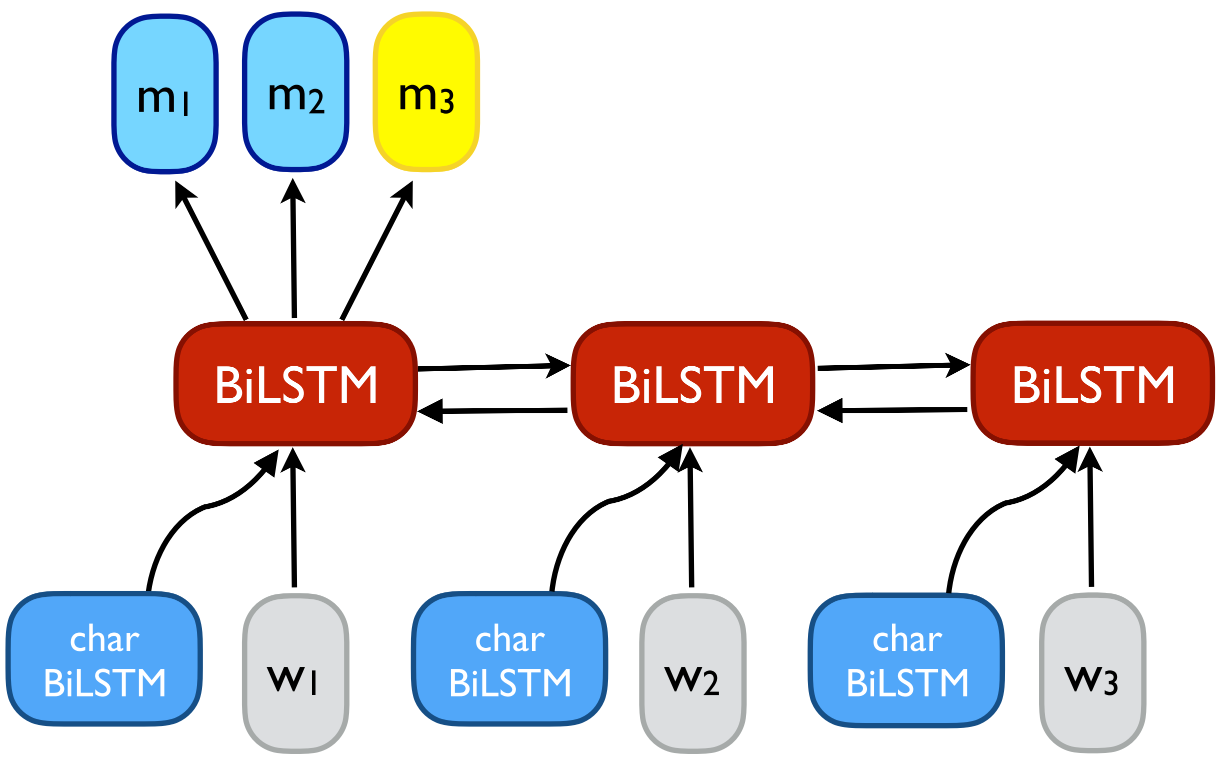
(Plank et al., 2016)

Multi-task Tri-training



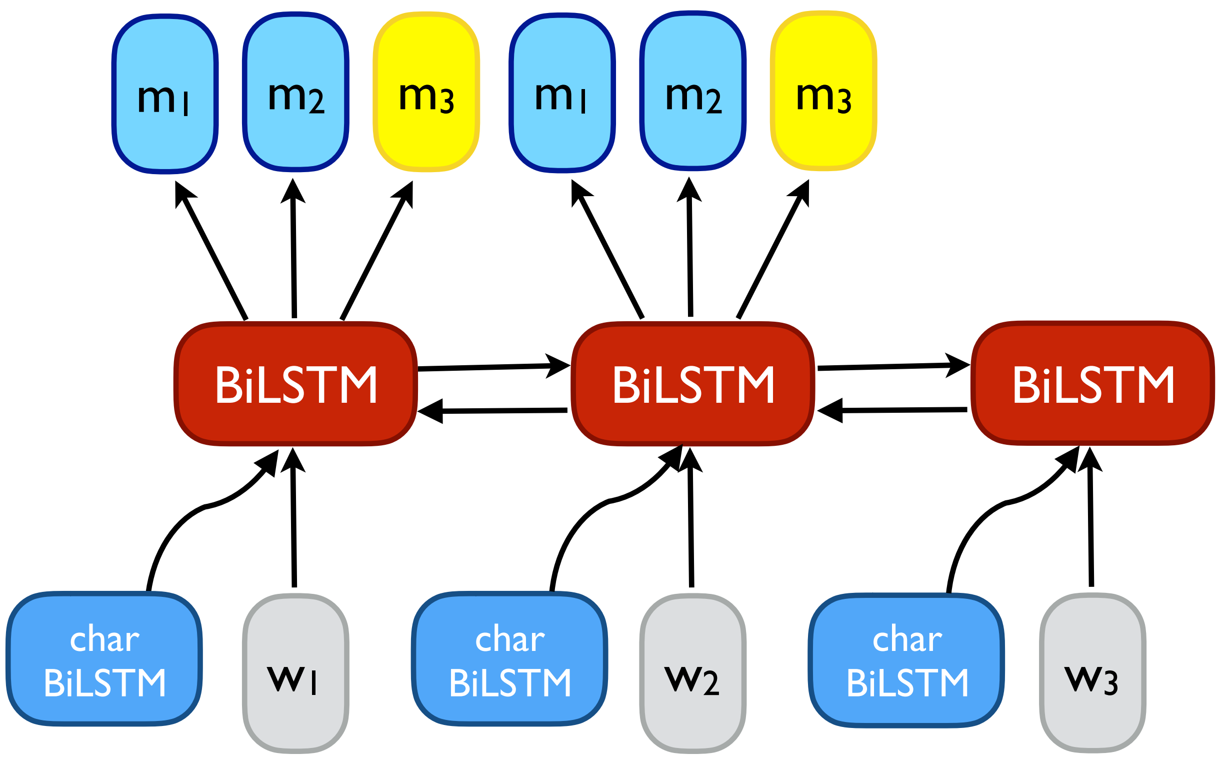
(Plank et al., 2016)

Multi-task Tri-training



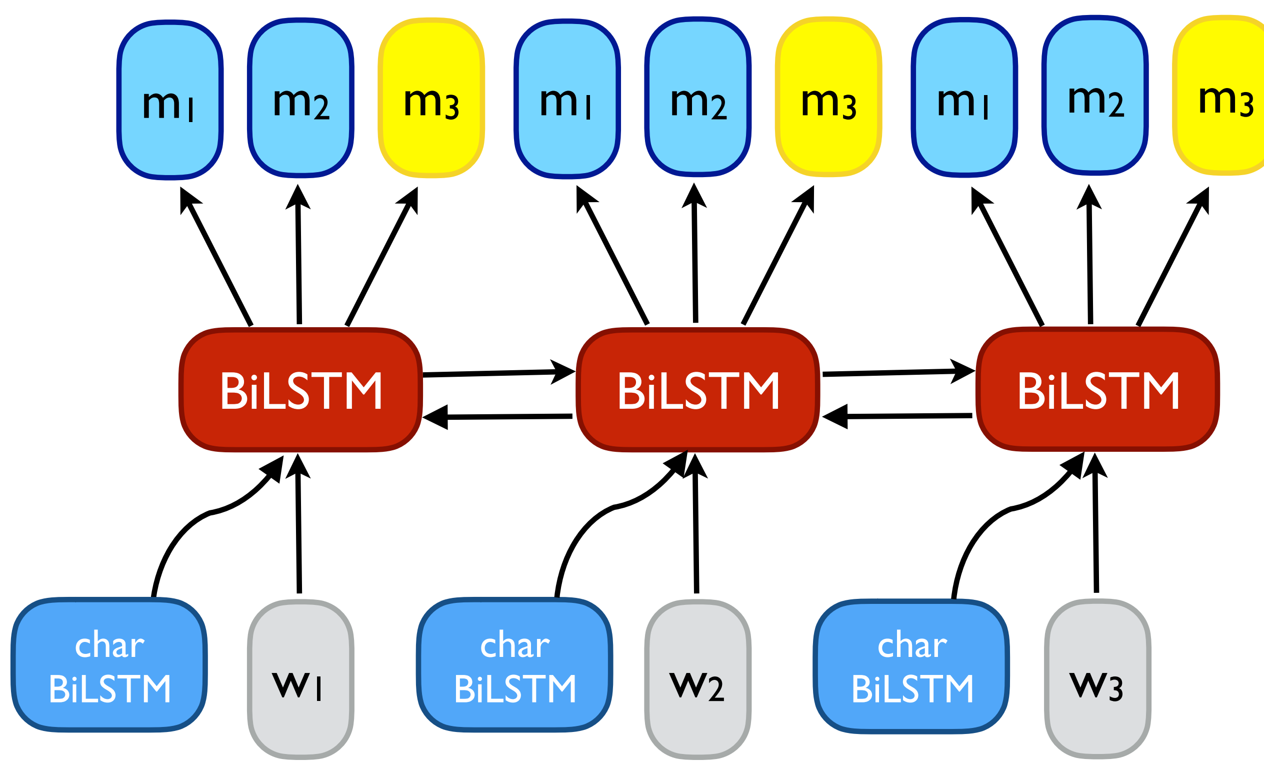
(Plank et al., 2016)

Multi-task Tri-training



(Plank et al., 2016)

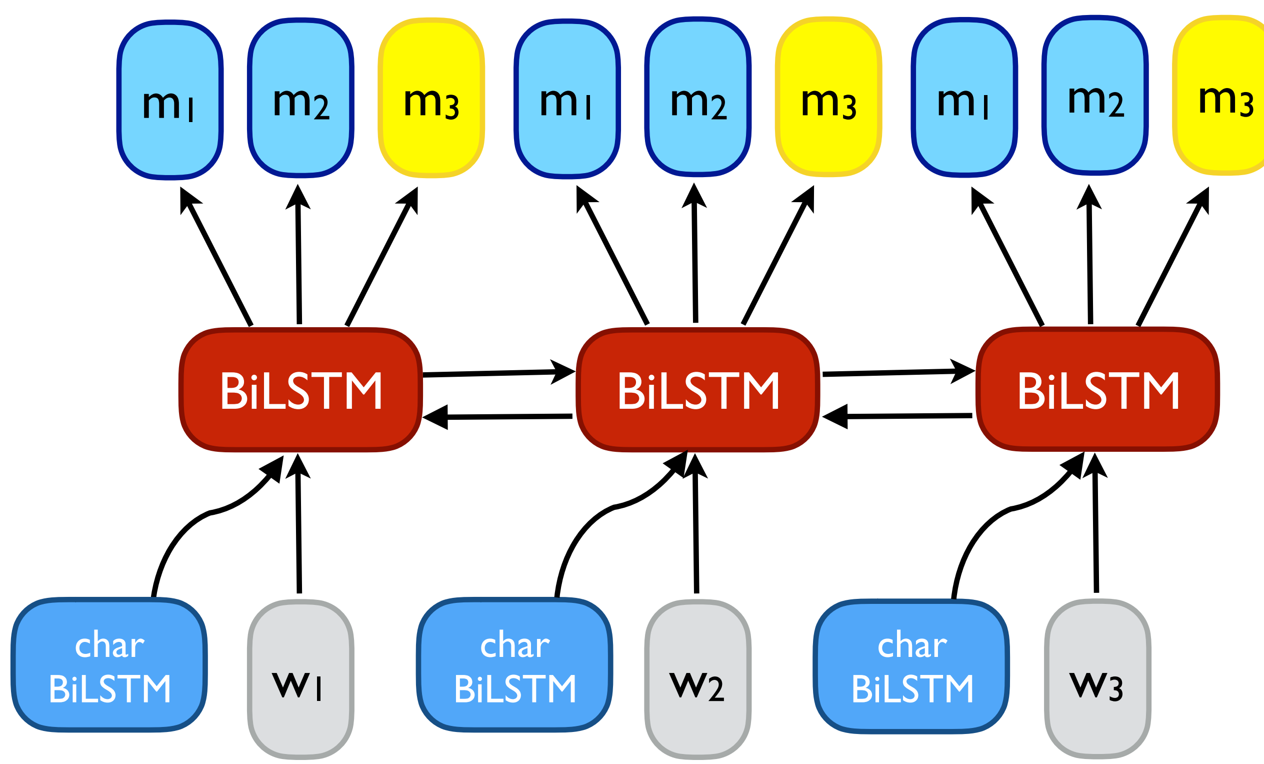
Multi-task Tri-training



(Plank et al., 2016)

Multi-task
Tri-training

$$L_{orth} = \|W_{m_1}^{\top} W_{m_2}\|_F^2$$ orthogonality constraint (Bousmalis et al., 2016)

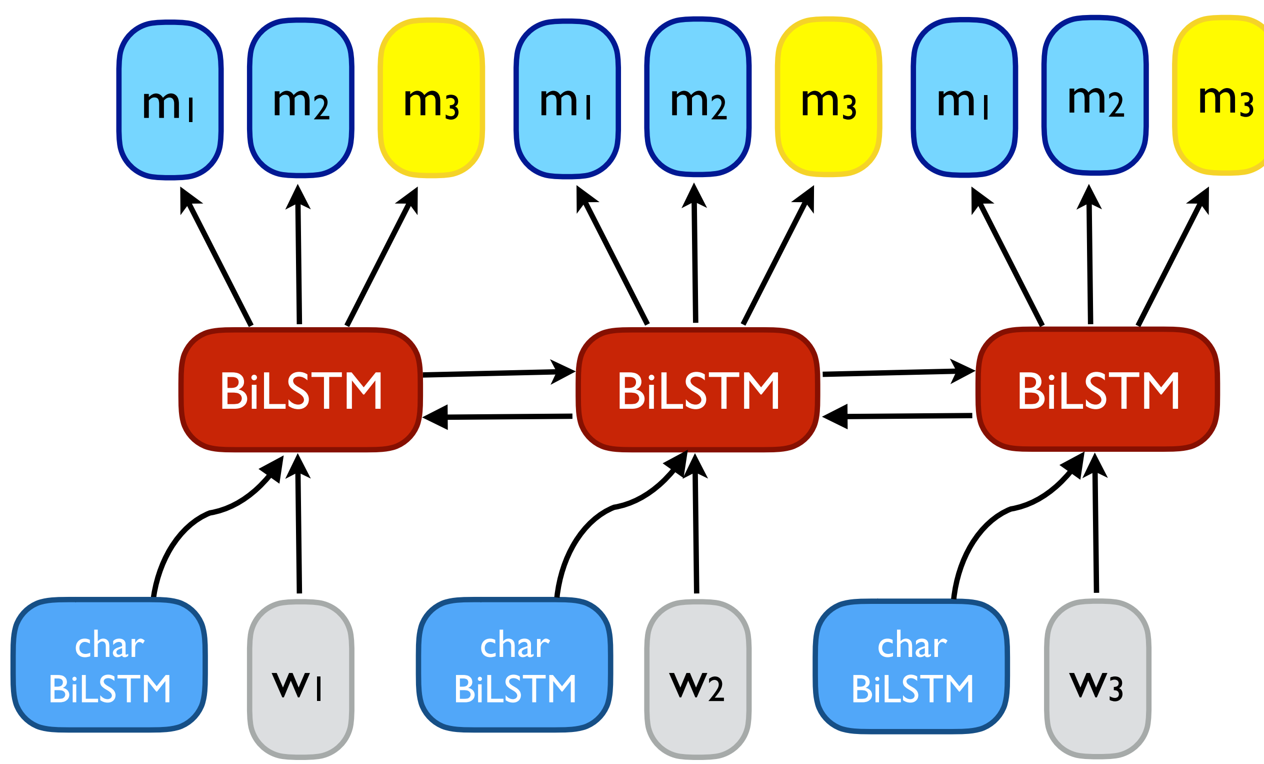


(Plank et al., 2016)

Multi-task Tri-training

$$L_{orth} = \| W_{m_1}^{\top} W_{m_2} \|_F^2$$

orthogonality constraint (Bousmalis et al., 2016)



(Plank et al., 2016)

Loss: $$L(\boldsymbol{\theta}) = -\sum_{i} \sum_{1,..,n} \log P_{m_i}(y \mid \vec{h}) + \gamma L_{orth}$$

Data & Tasks

Data & Tasks

Two tasks: **Domains:**

Data & Tasks

Two tasks:

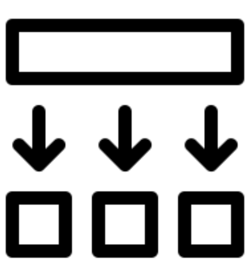
Domains:

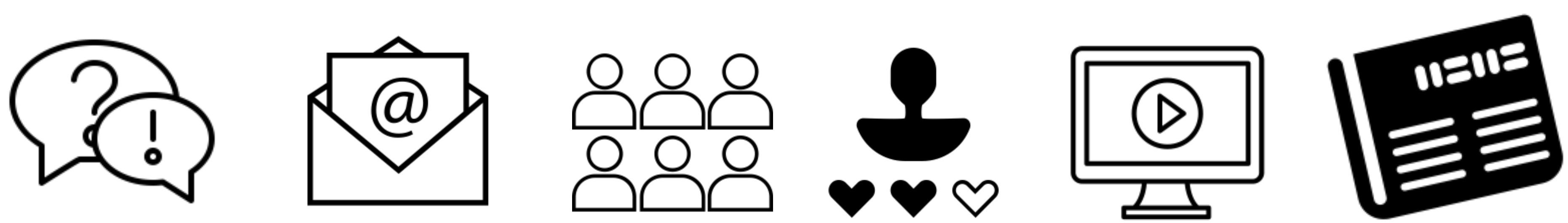
Sentiment analysis on Amazon reviews dataset (Blitzer et al, 2006)

Data & Tasks

Two tasks: **Domains:**

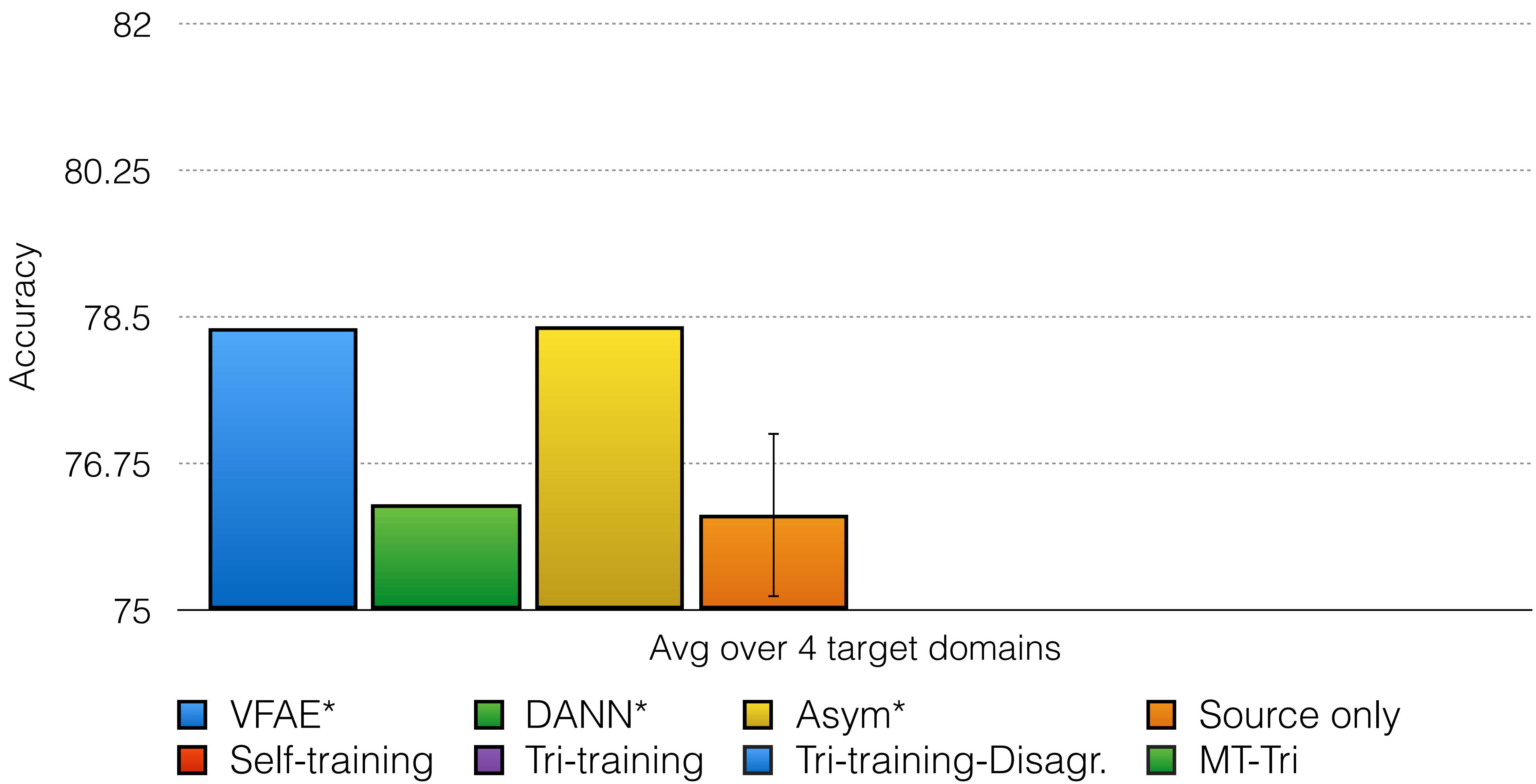
Sentiment analysis on Amazon reviews dataset (Blitzer et al, 2006)





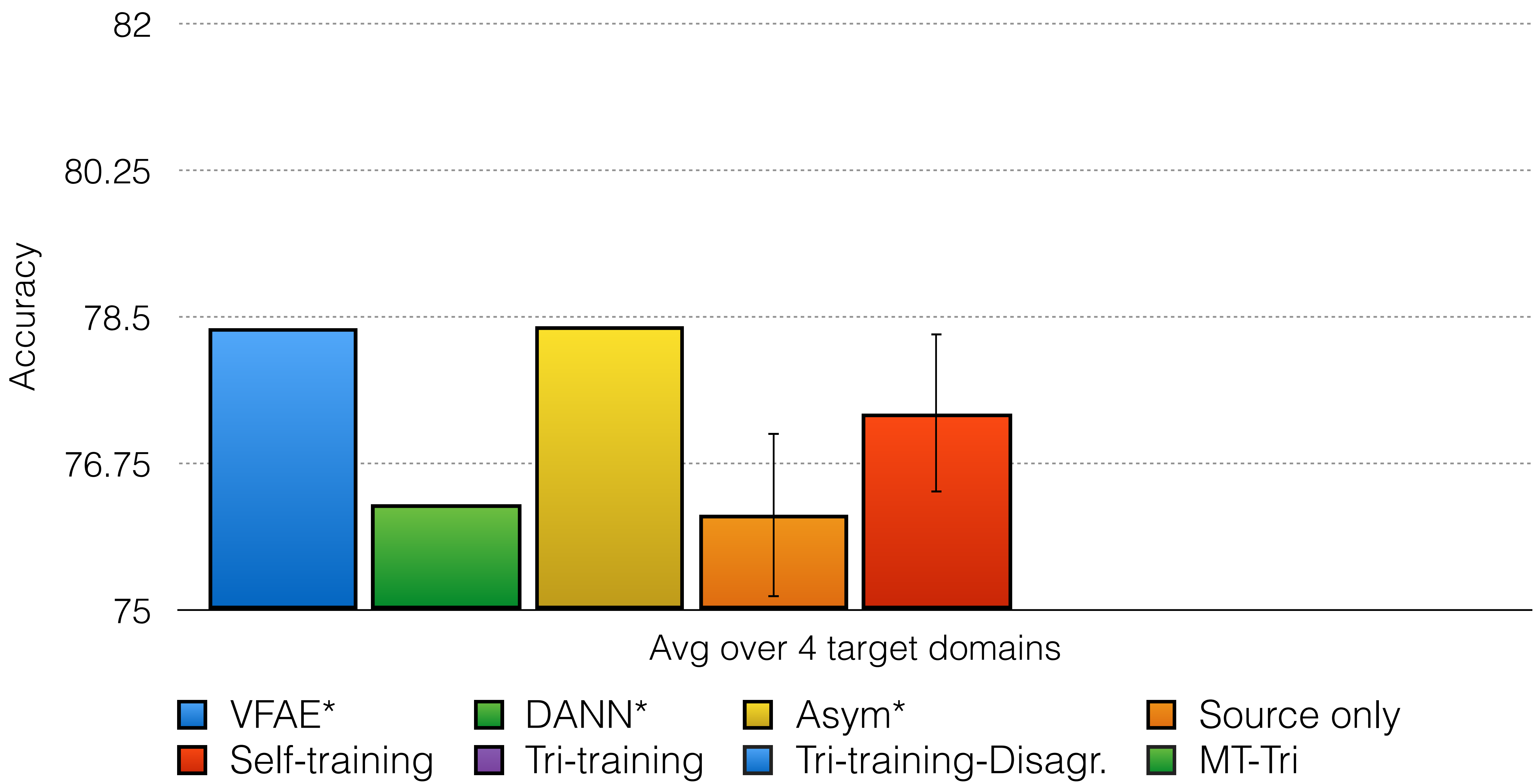
POS tagging on SANCL 2012 dataset (Petrov and McDonald, 2012)

Sentiment Analysis Results



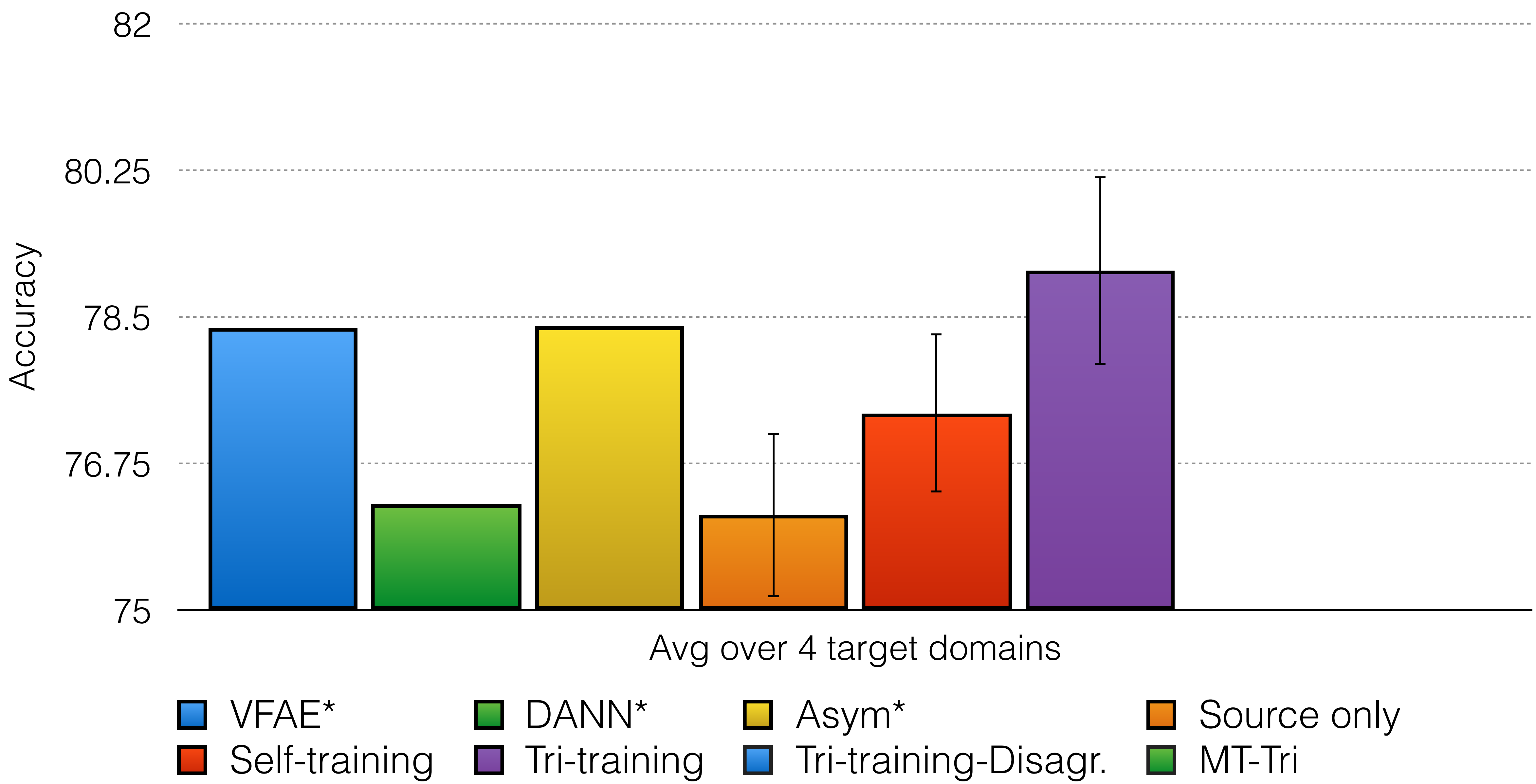
* result from Saito et al., (2017)

Sentiment Analysis Results



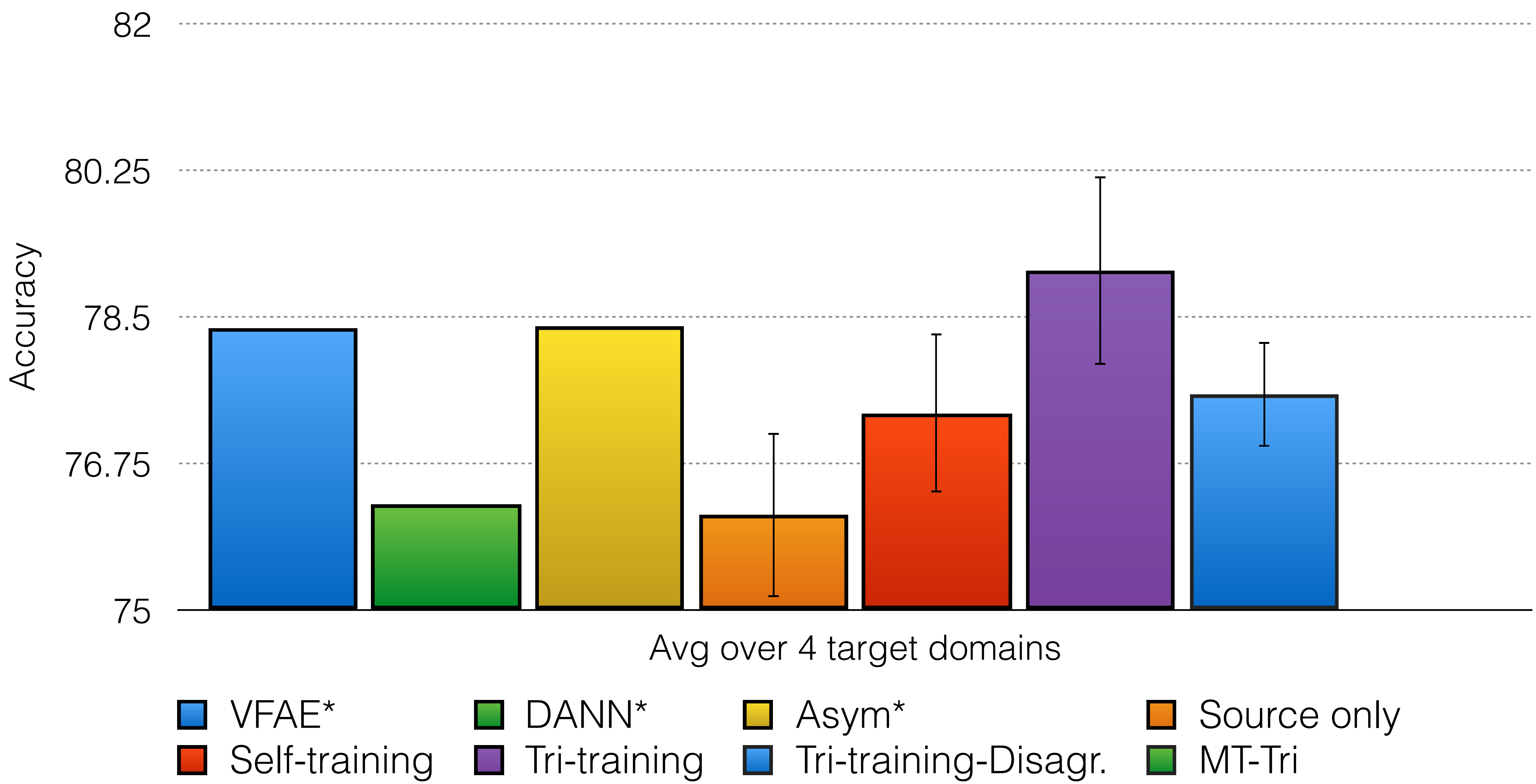
* result from Saito et al., (2017)

Sentiment Analysis Results



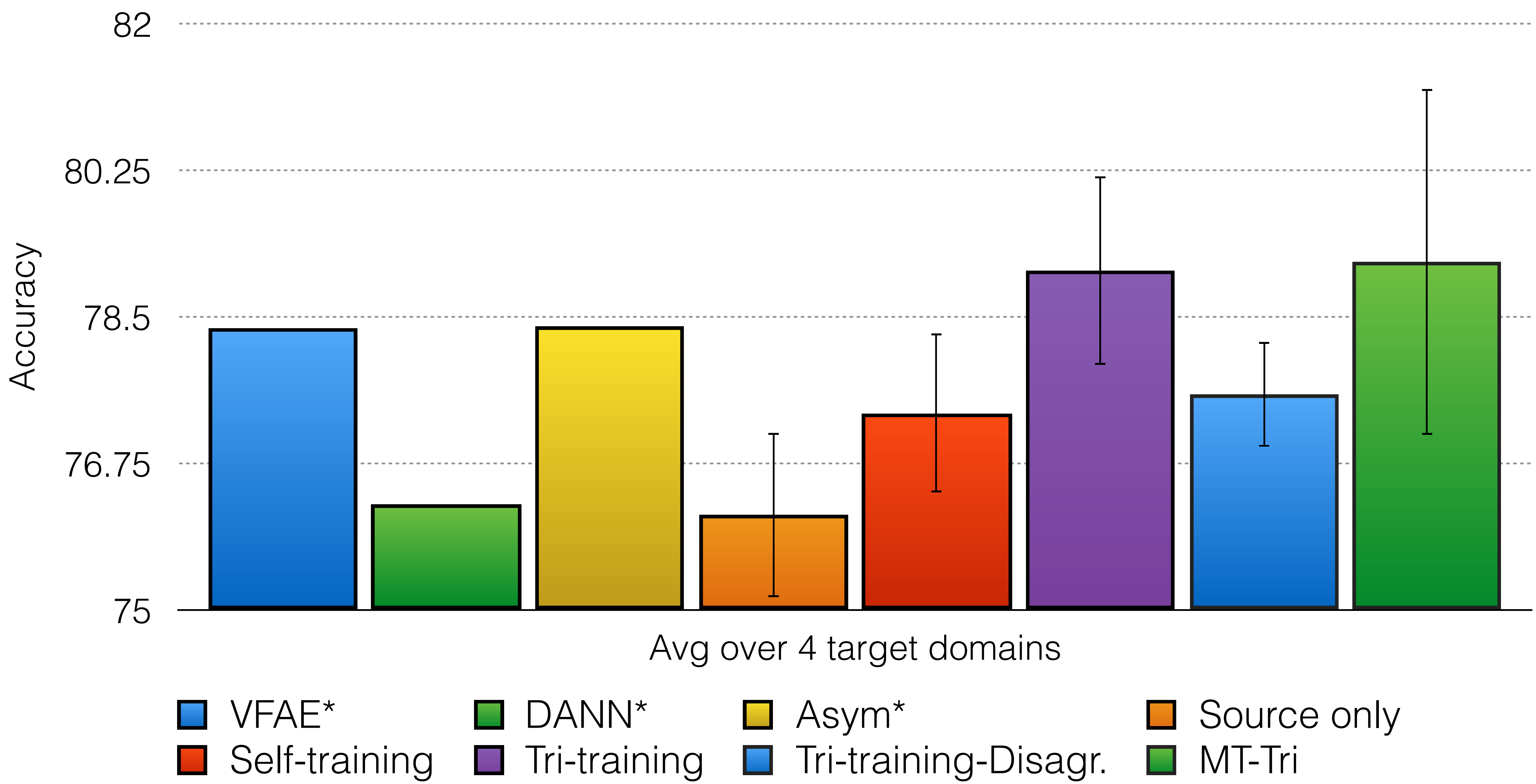
* result from Saito et al., (2017)

Sentiment Analysis Results



* result from Saito et al., (2017)

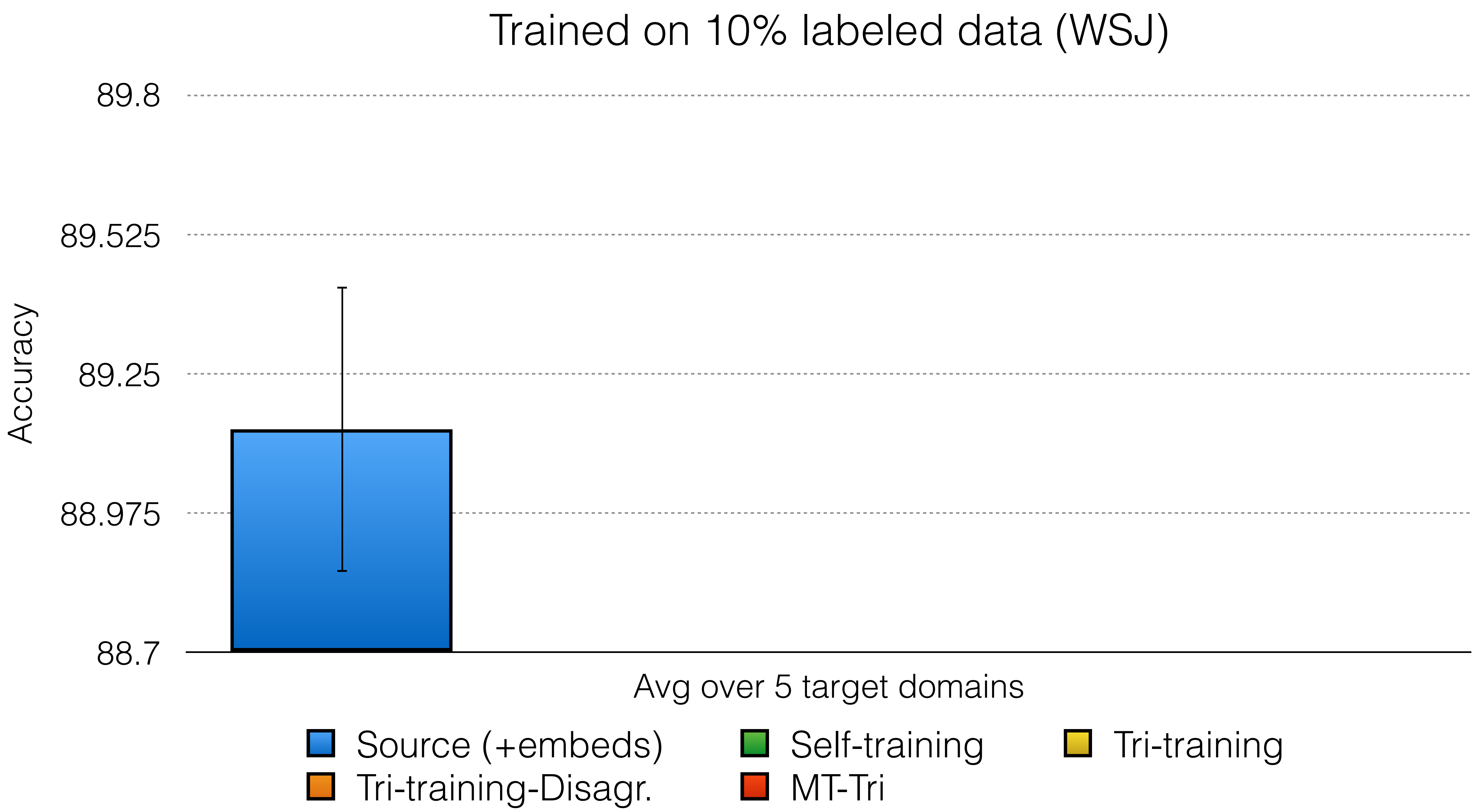
Sentiment Analysis Results



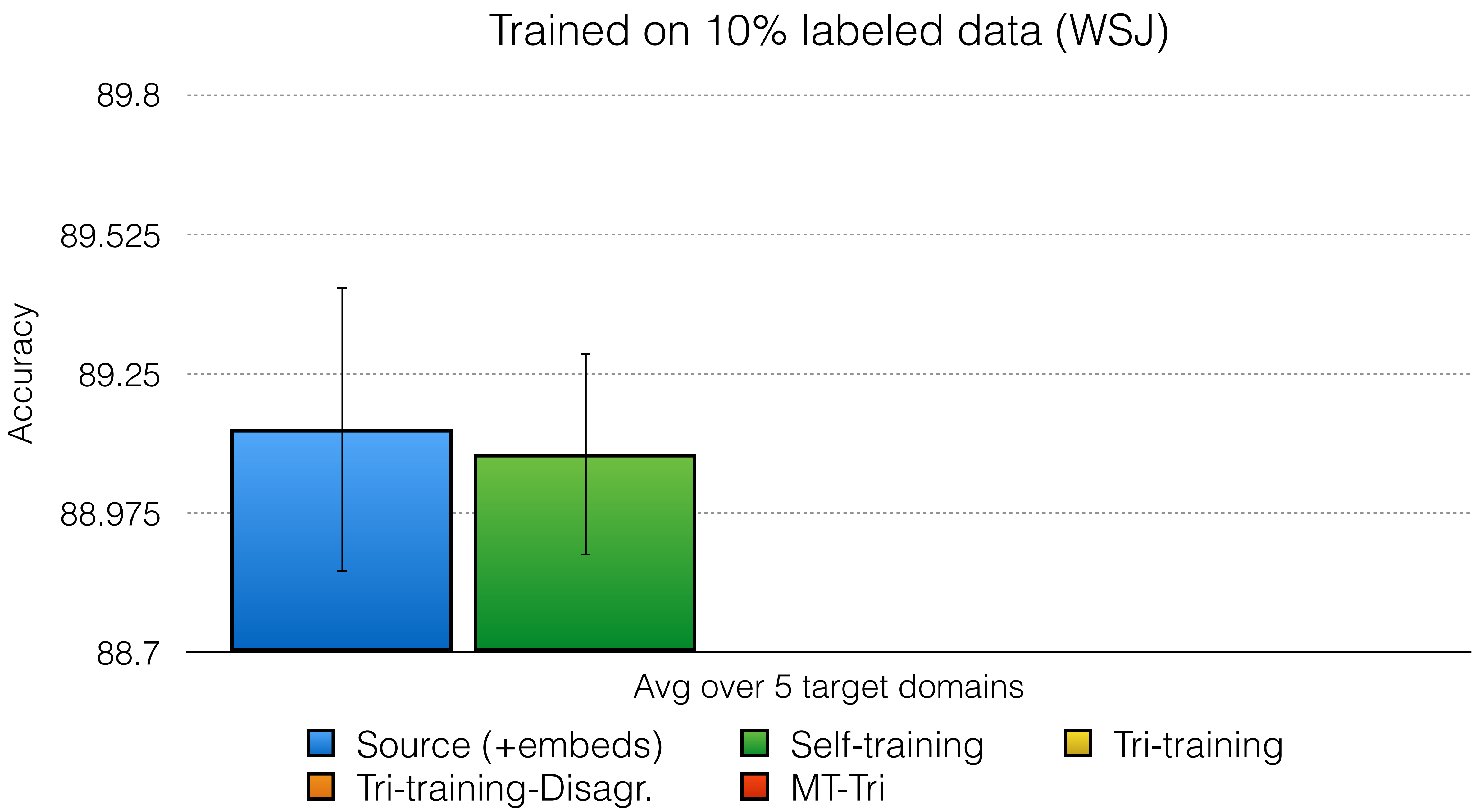
* result from Saito et al., (2017)

- Multi-task tri-training slightly outperforms tri-training, but has higher variance.

POS Tagging Results

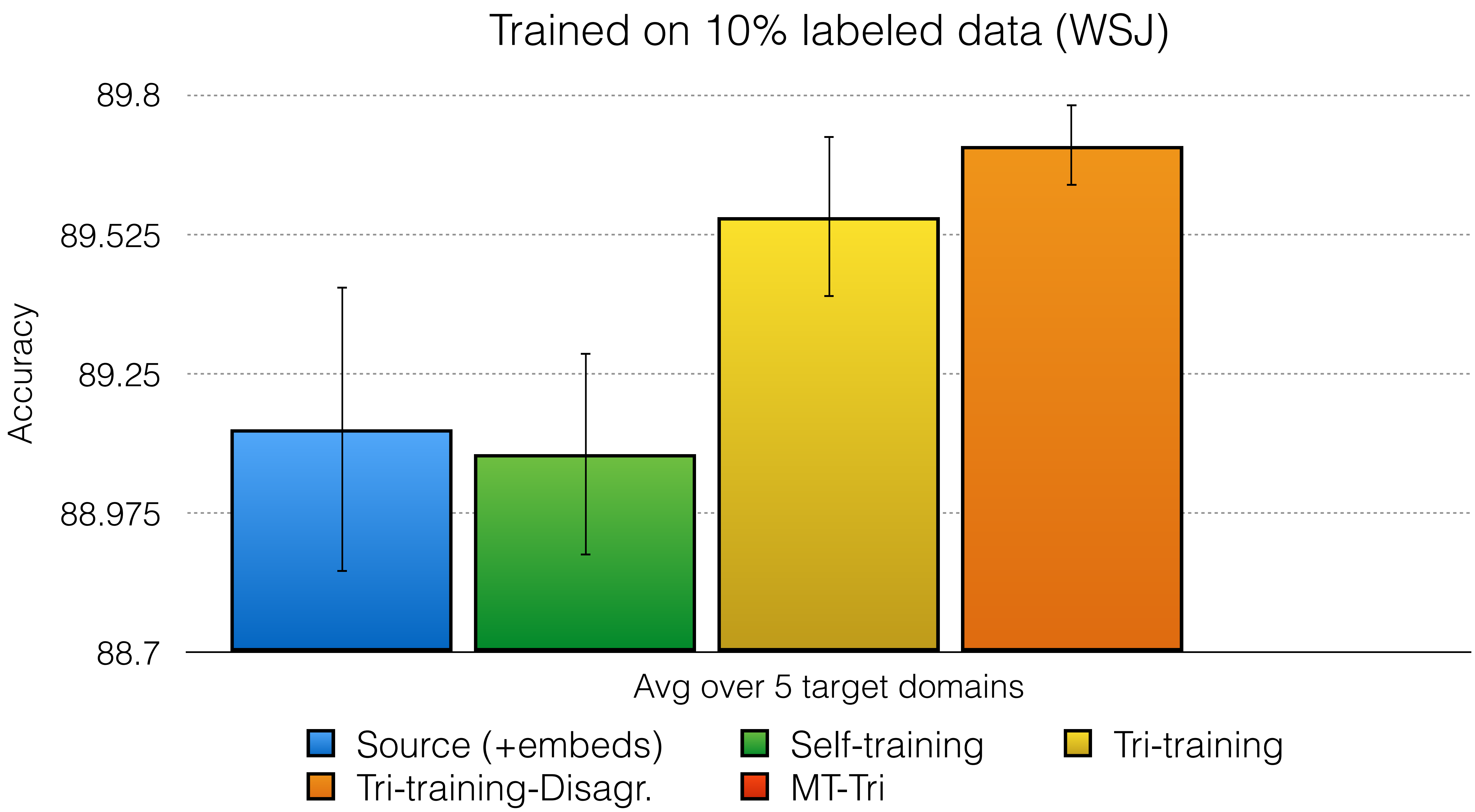


POS Tagging Results

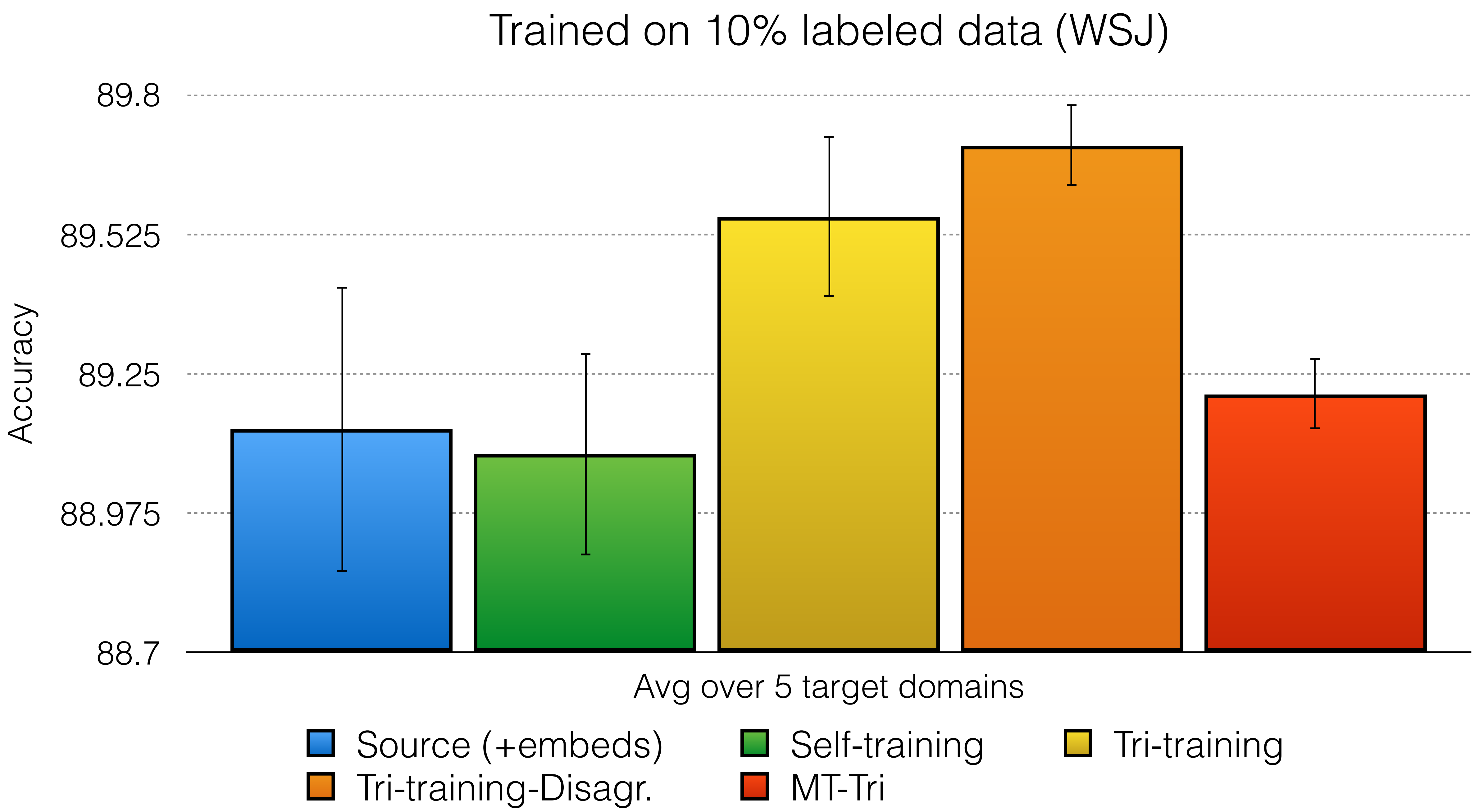


POS Tagging Results

POS Tagging Results

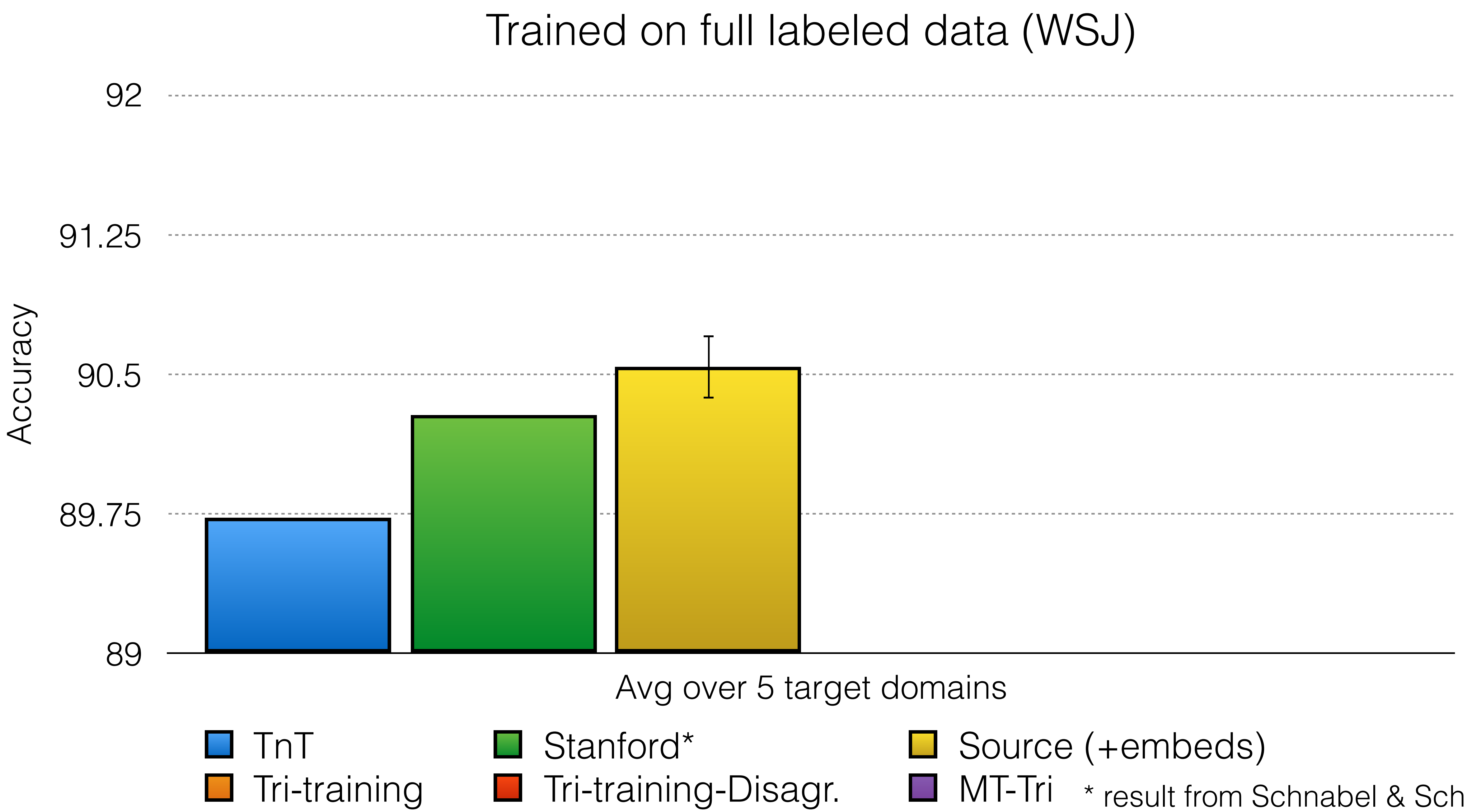


POS Tagging Results



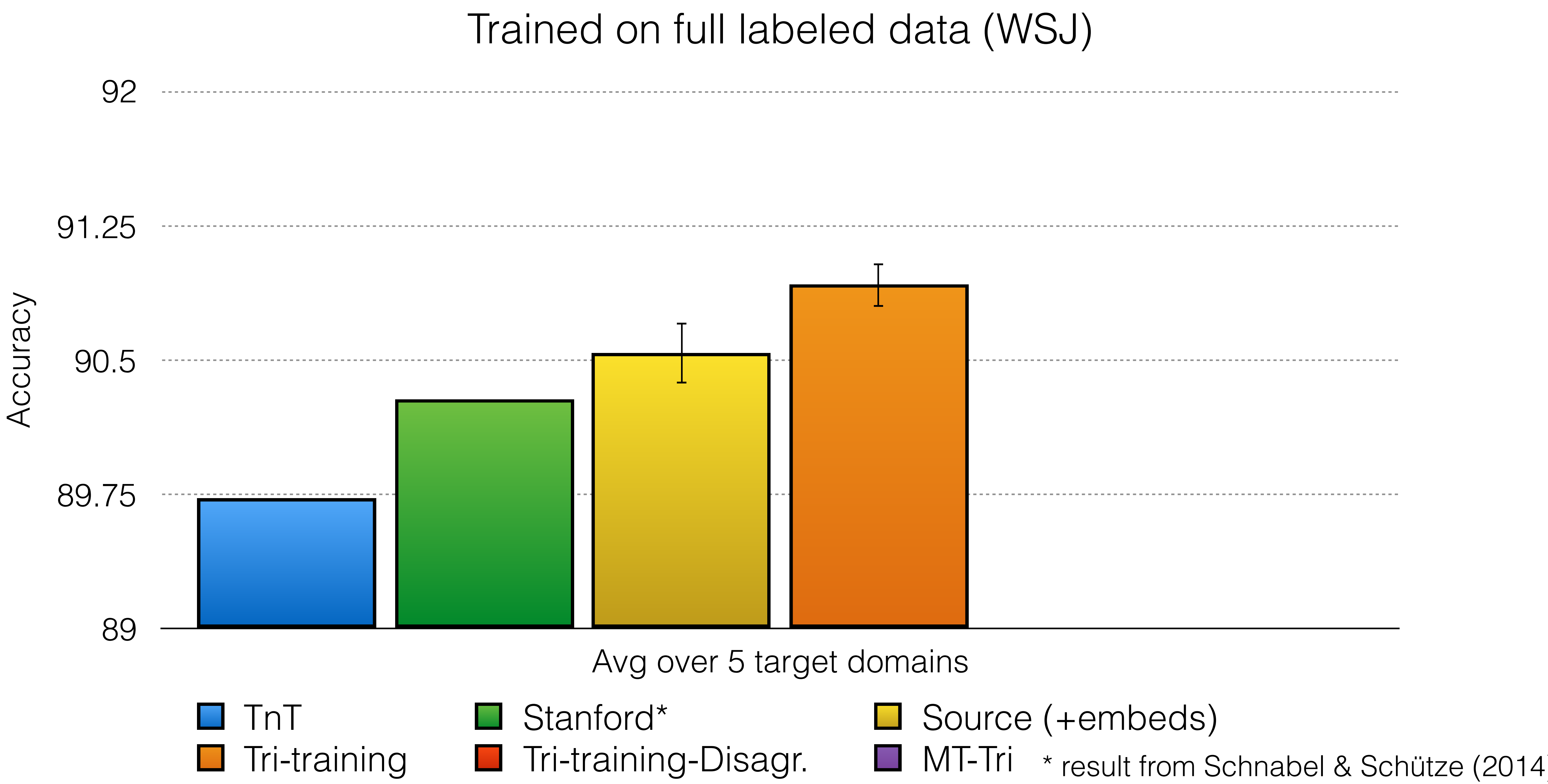
- Tri-training with disagreement works best with little data.

POS Tagging Results

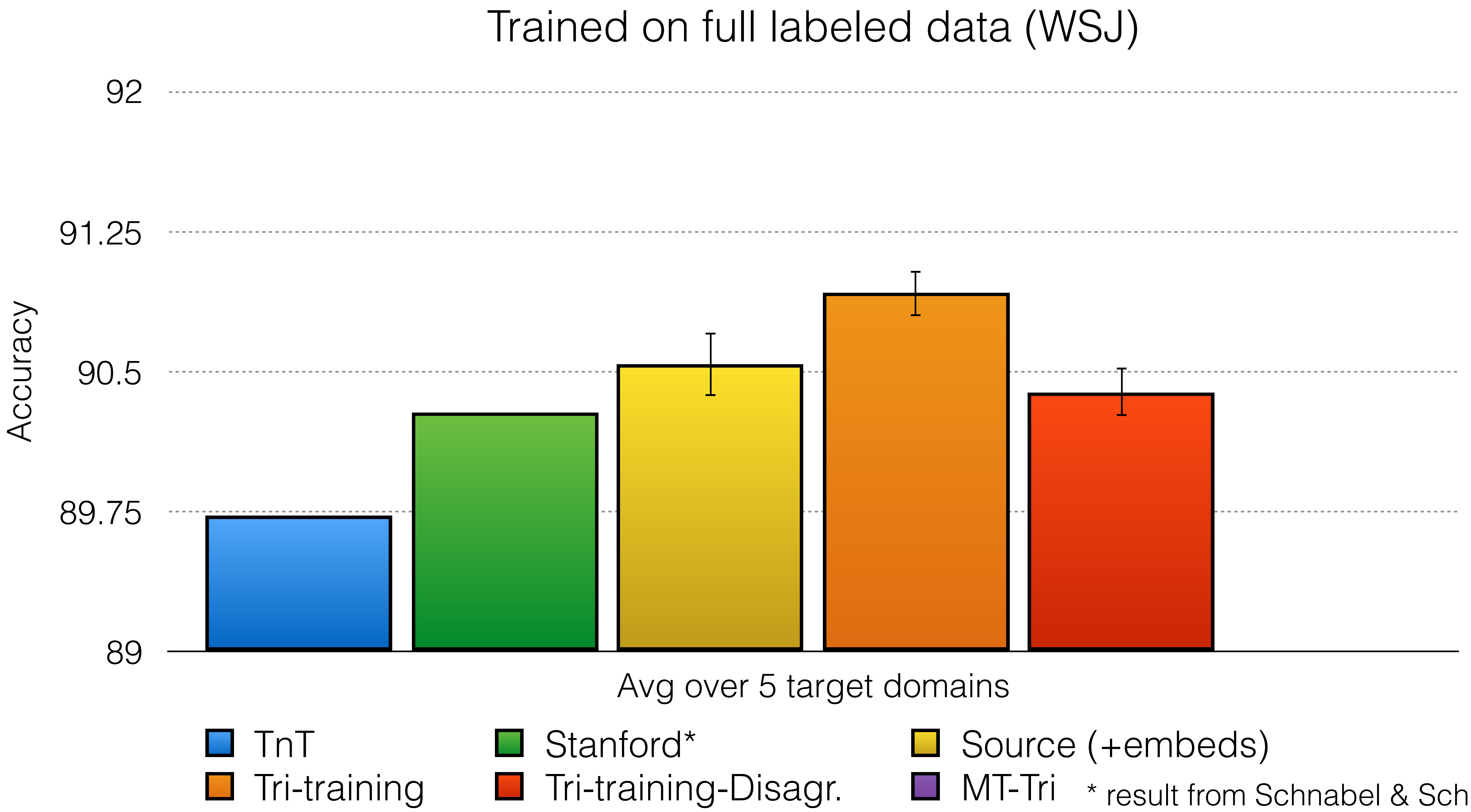


* result from Schnabel & Schütze (2014)

POS Tagging Results

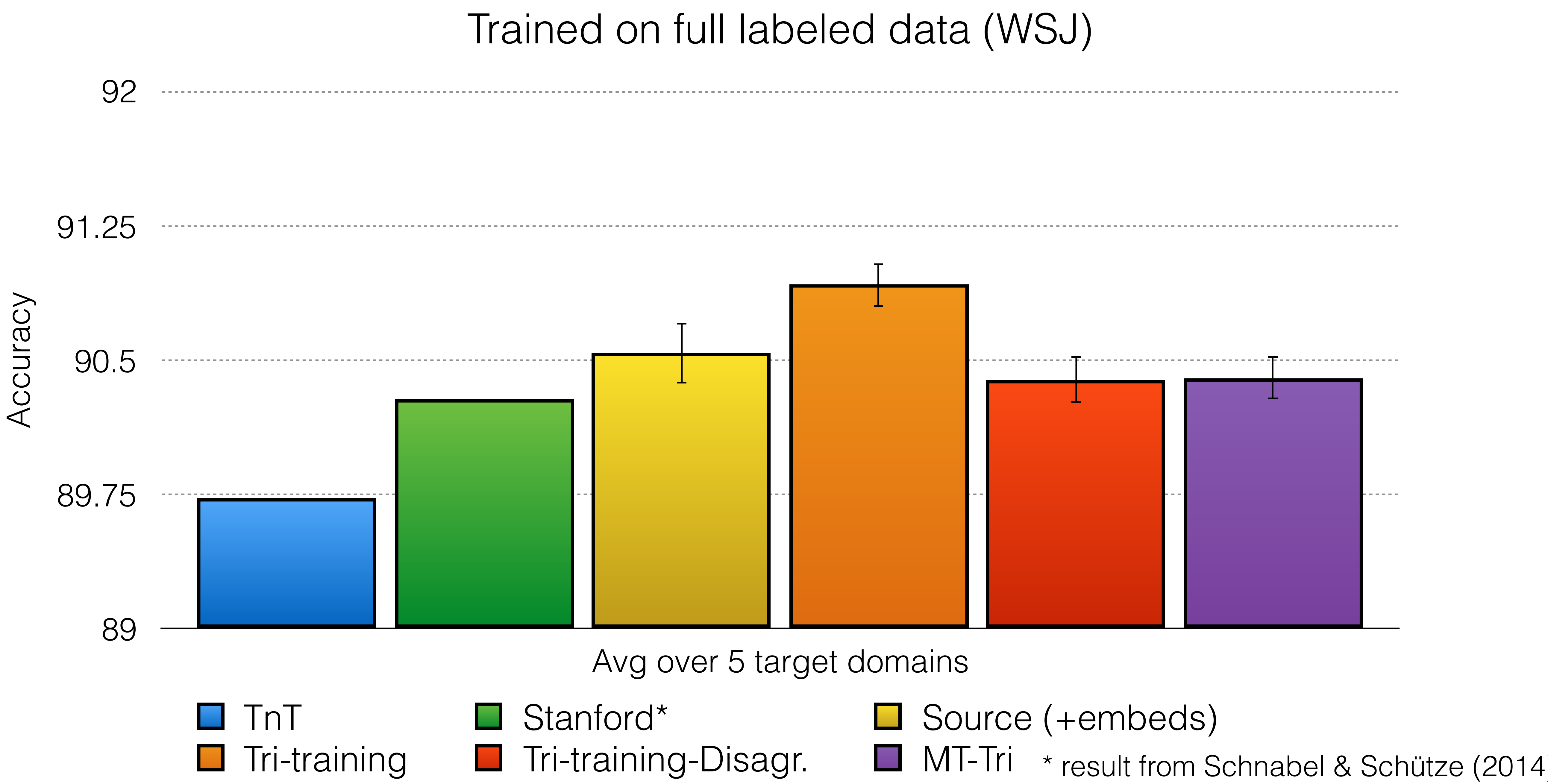


POS Tagging Results



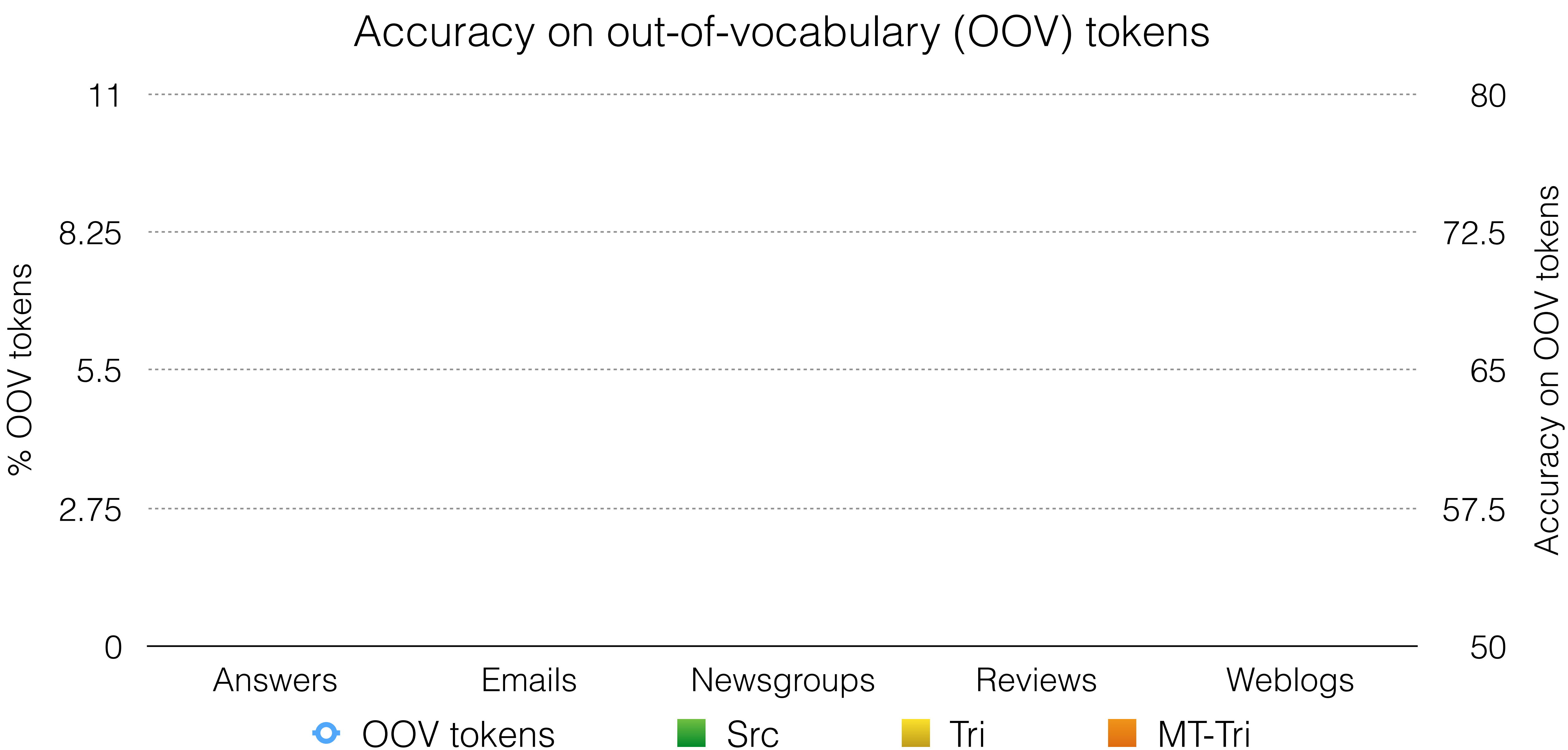
* result from Schnabel & Schütze (2014)

POS Tagging Results

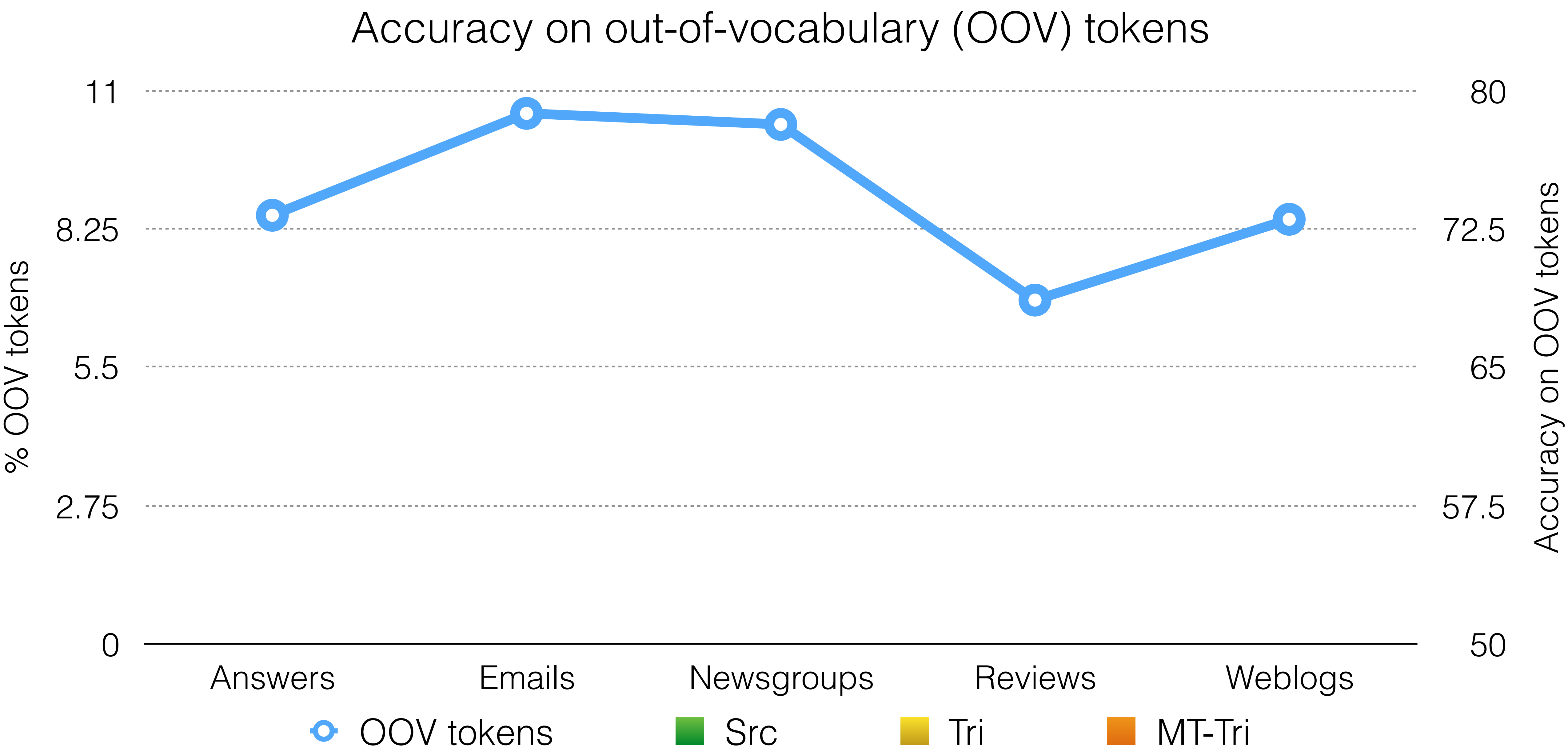


- Tri-training works best in the full data setting.

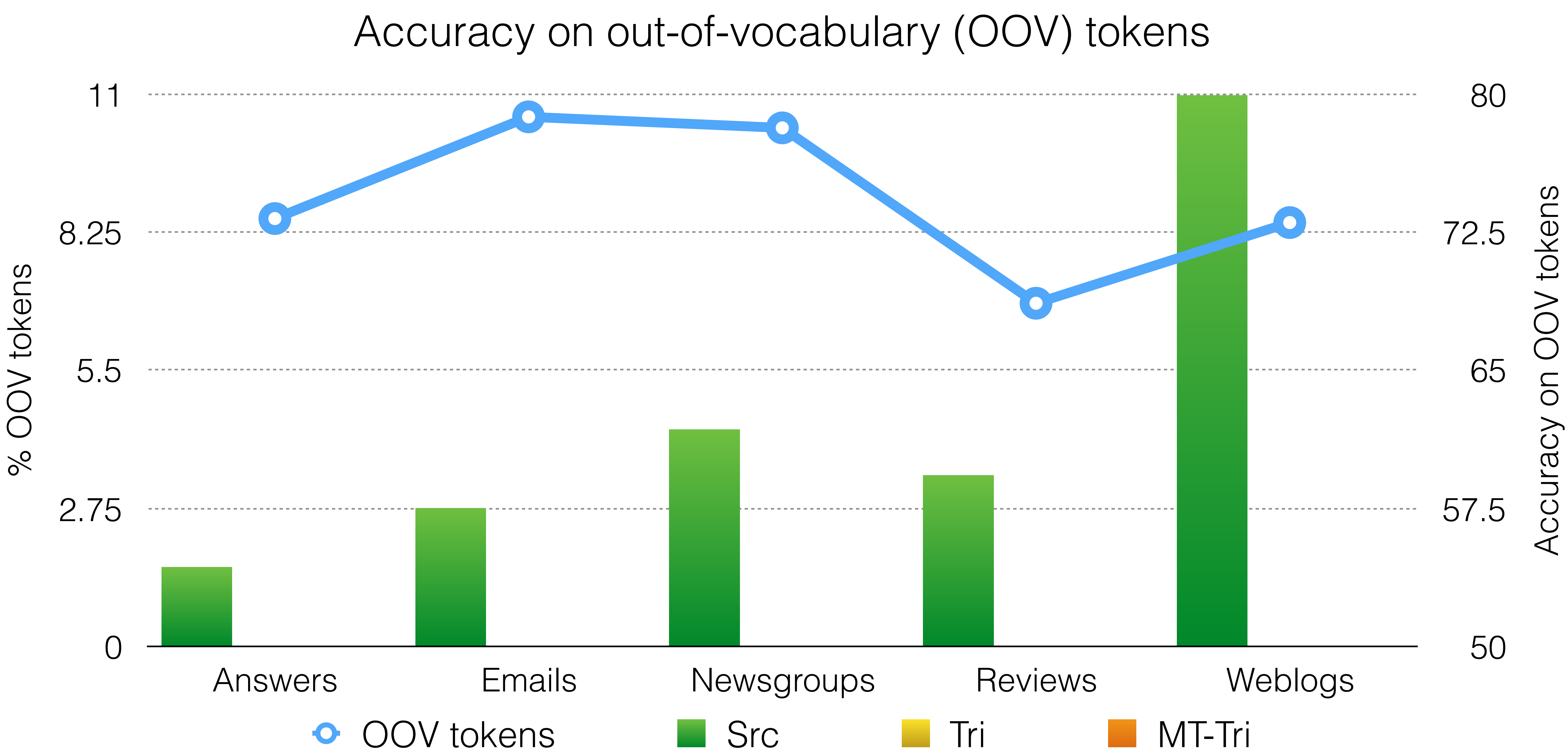
POS Tagging Analysis



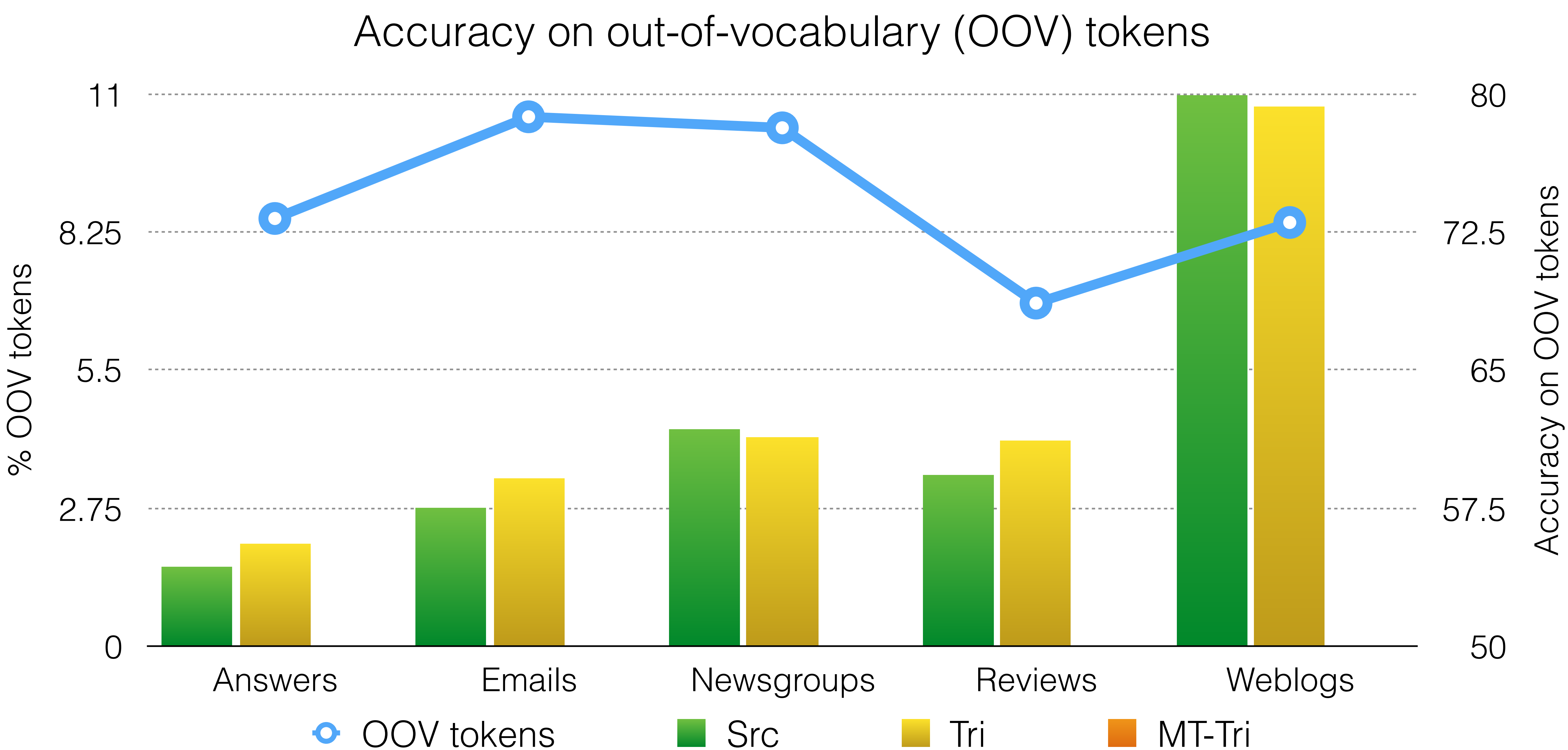
POS Tagging Analysis



POS Tagging Analysis

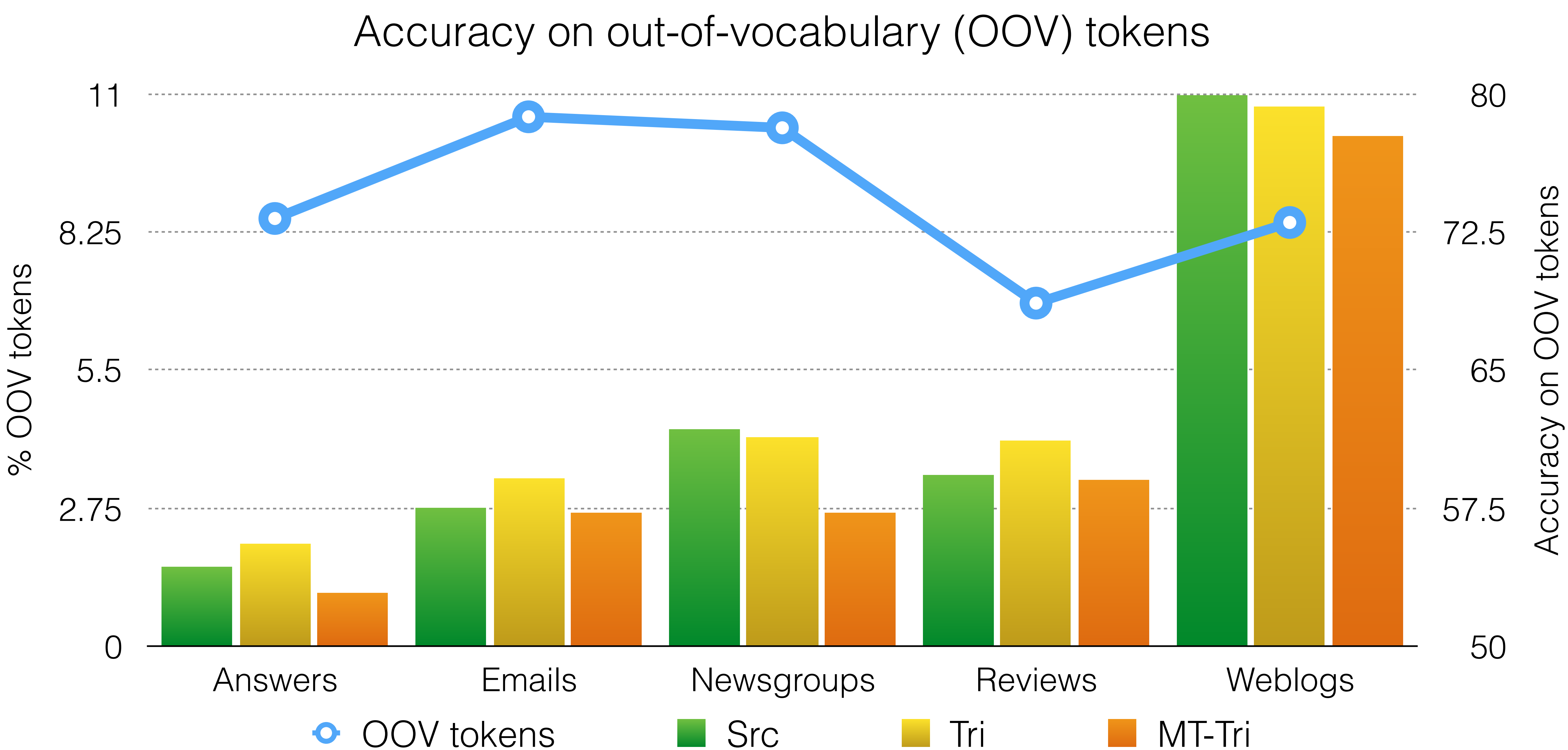


POS Tagging Analysis



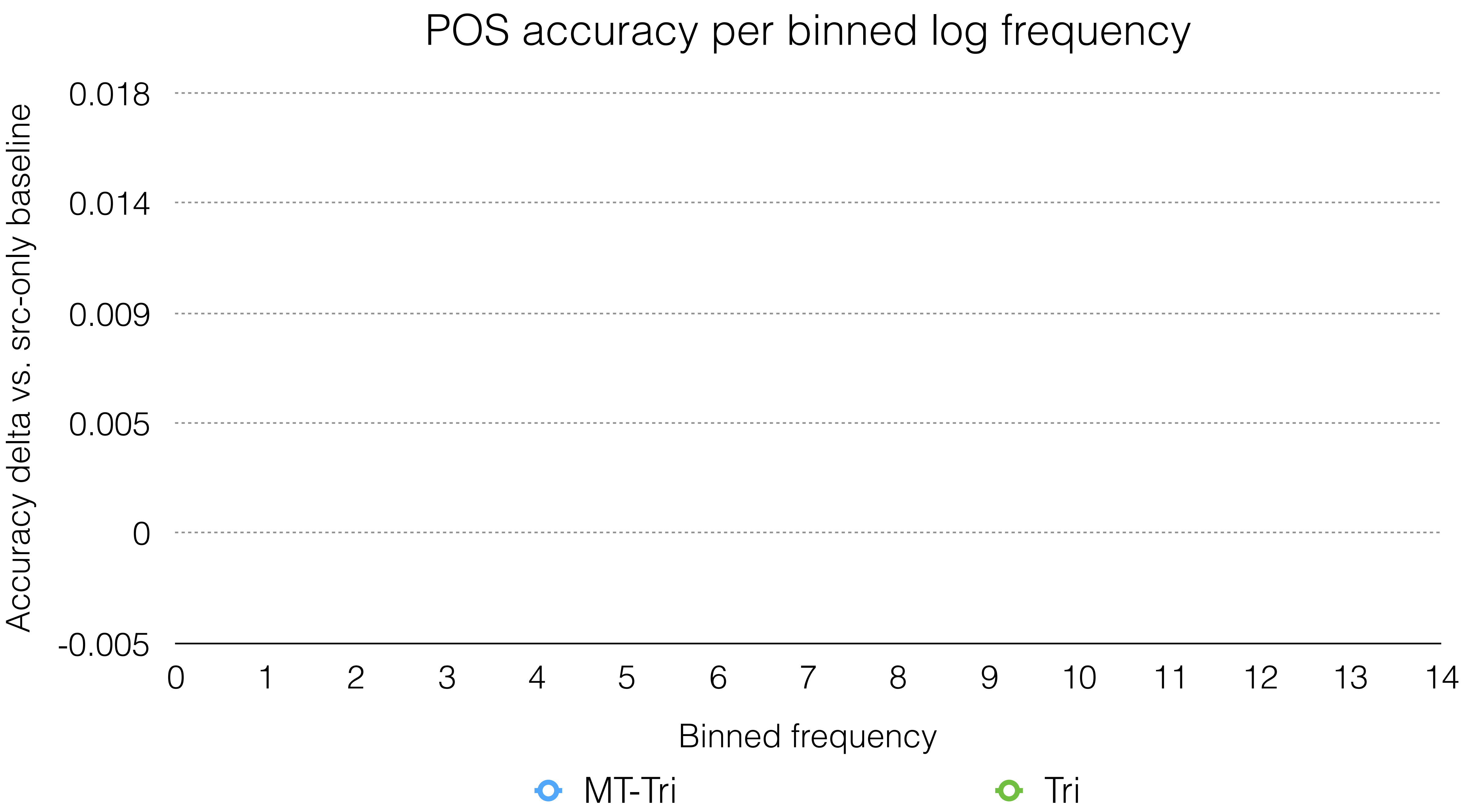
- Classic tri-training works best on OOV tokens.

POS Tagging Analysis

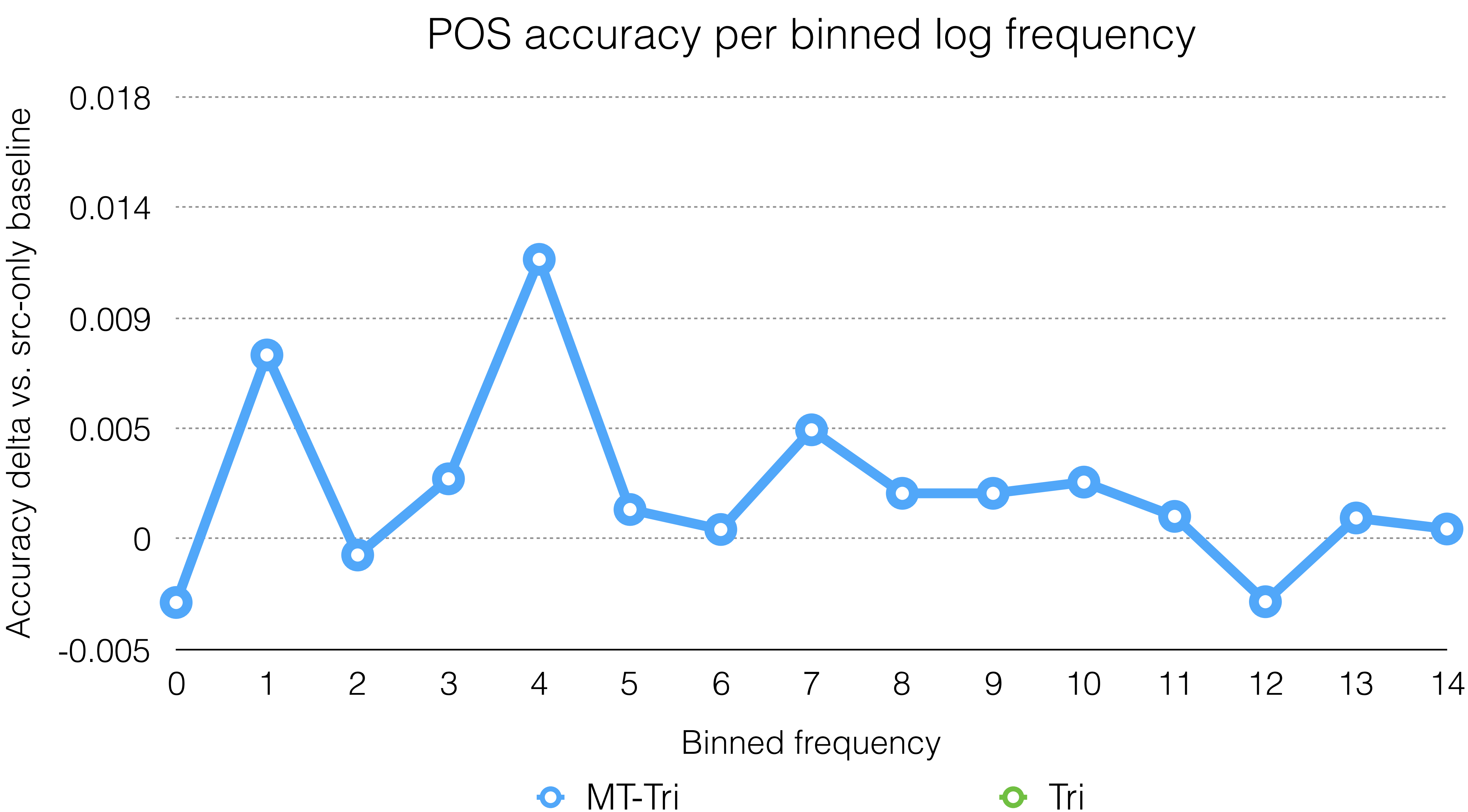


- Classic tri-training works best on OOV tokens.
- MT-Tri does worse than source-only baseline on OOV.

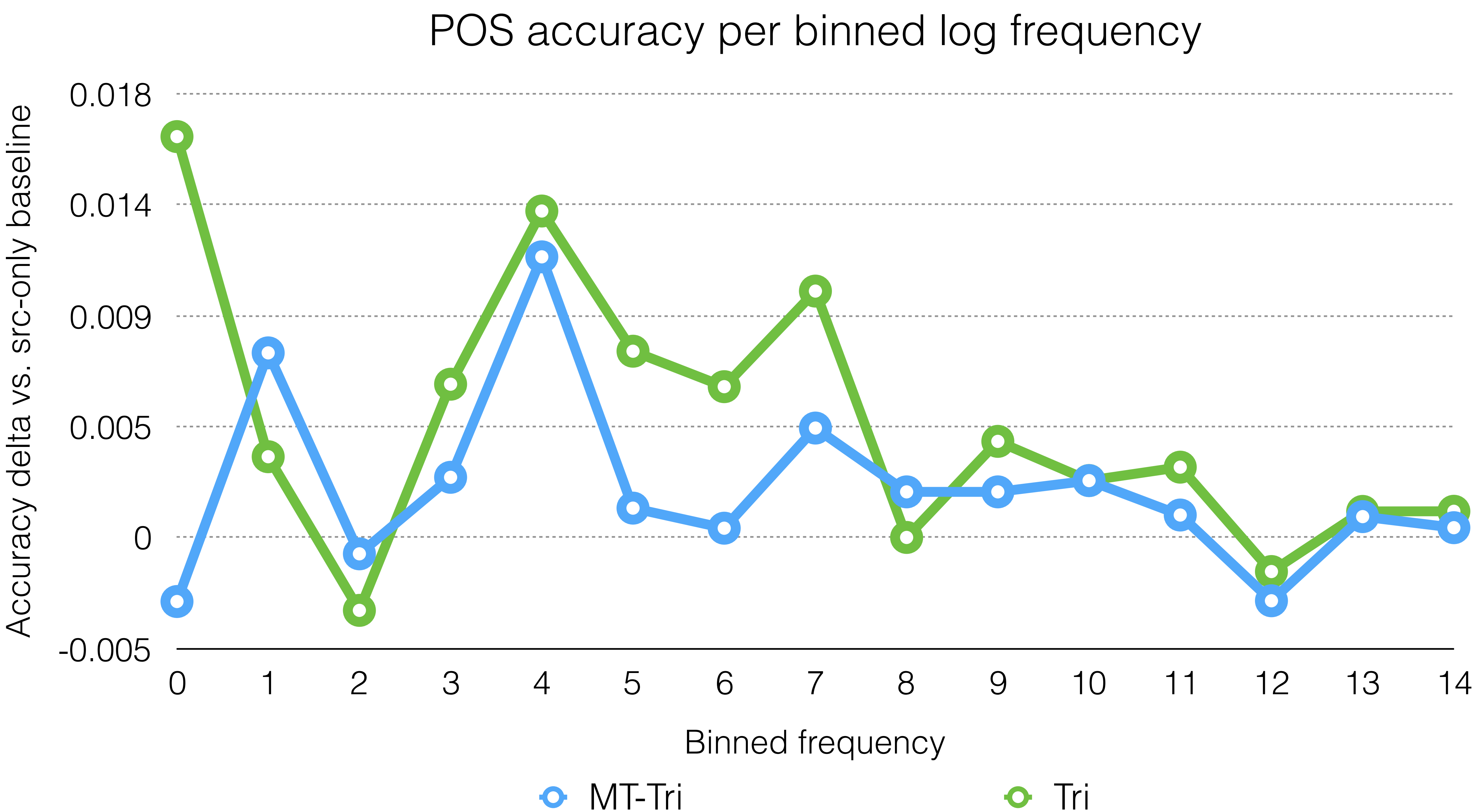
POS Tagging Analysis



POS Tagging Analysis

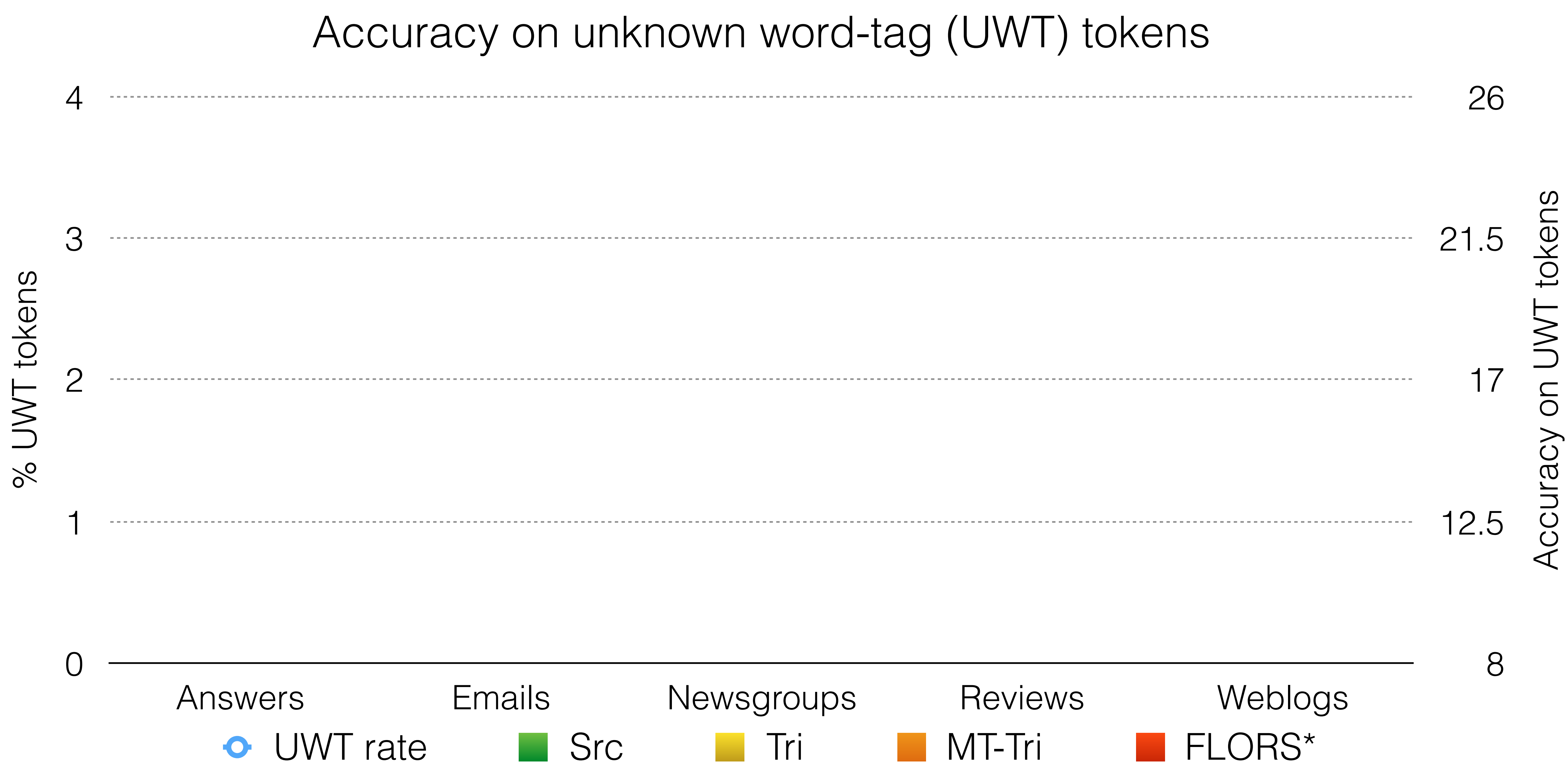


POS Tagging Analysis



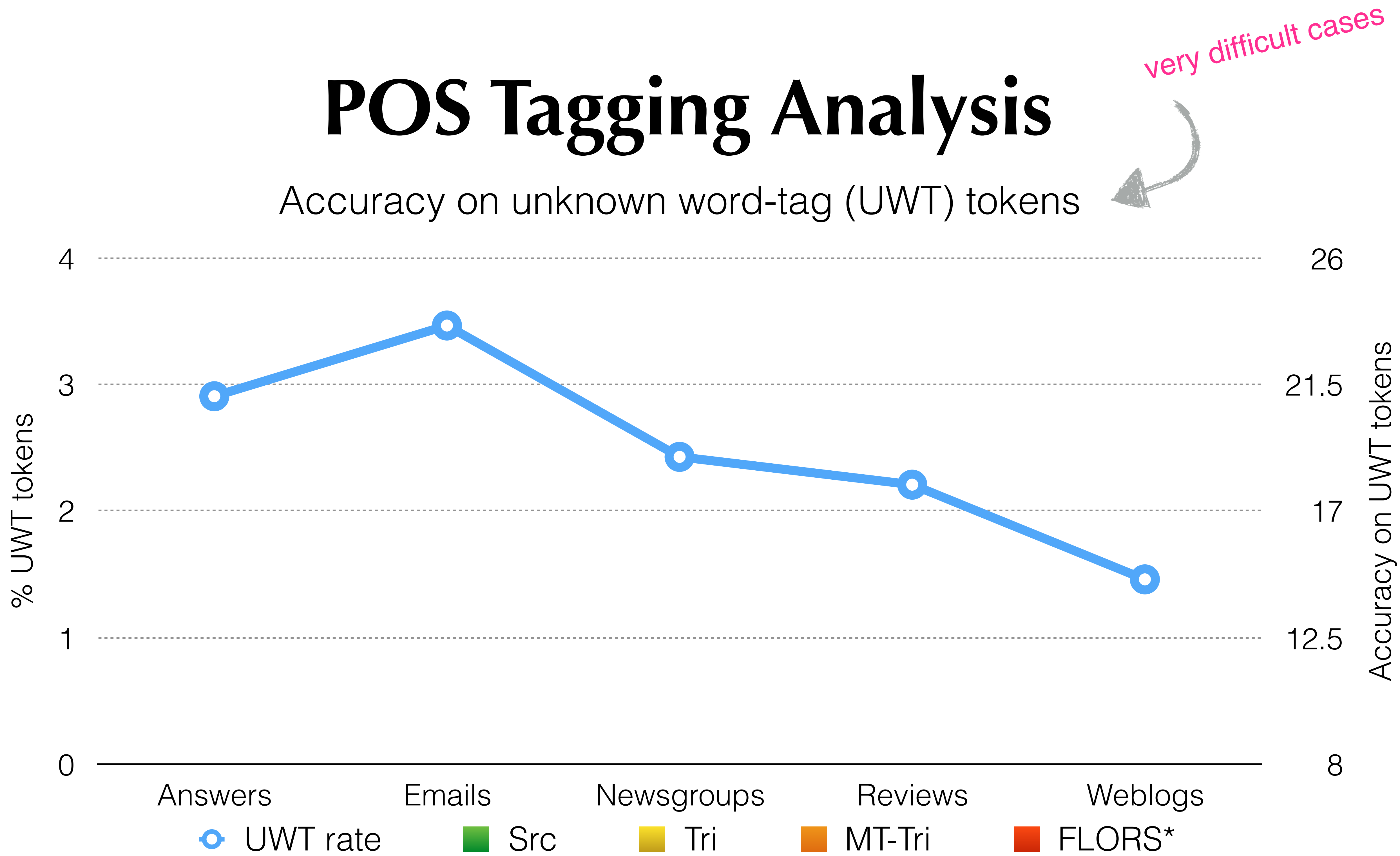
- Tri-training works best on low-frequency tokens (leftmost bins).

POS Tagging Analysis



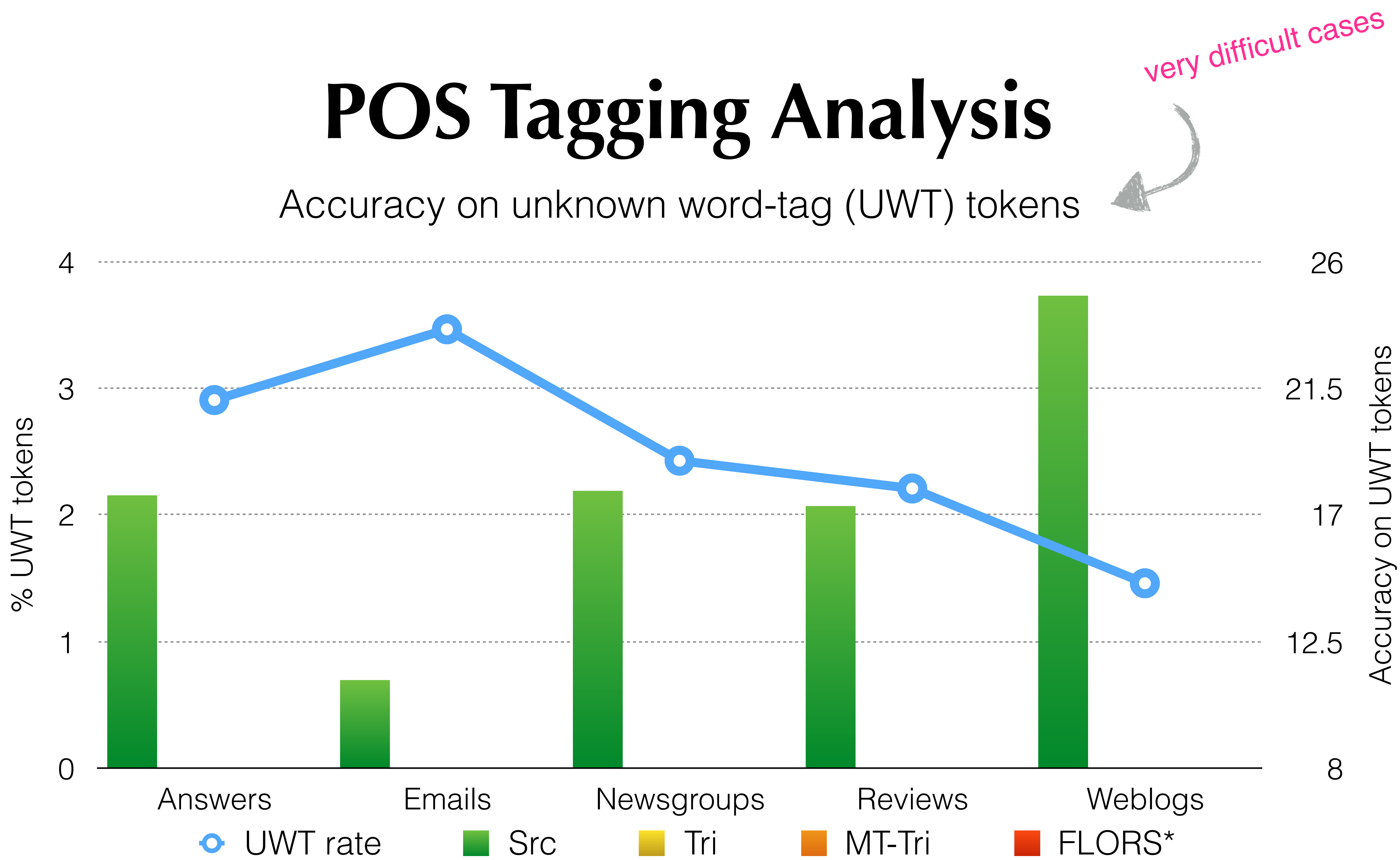
* result from Schnabel & Schütze (2014)

POS Tagging Analysis



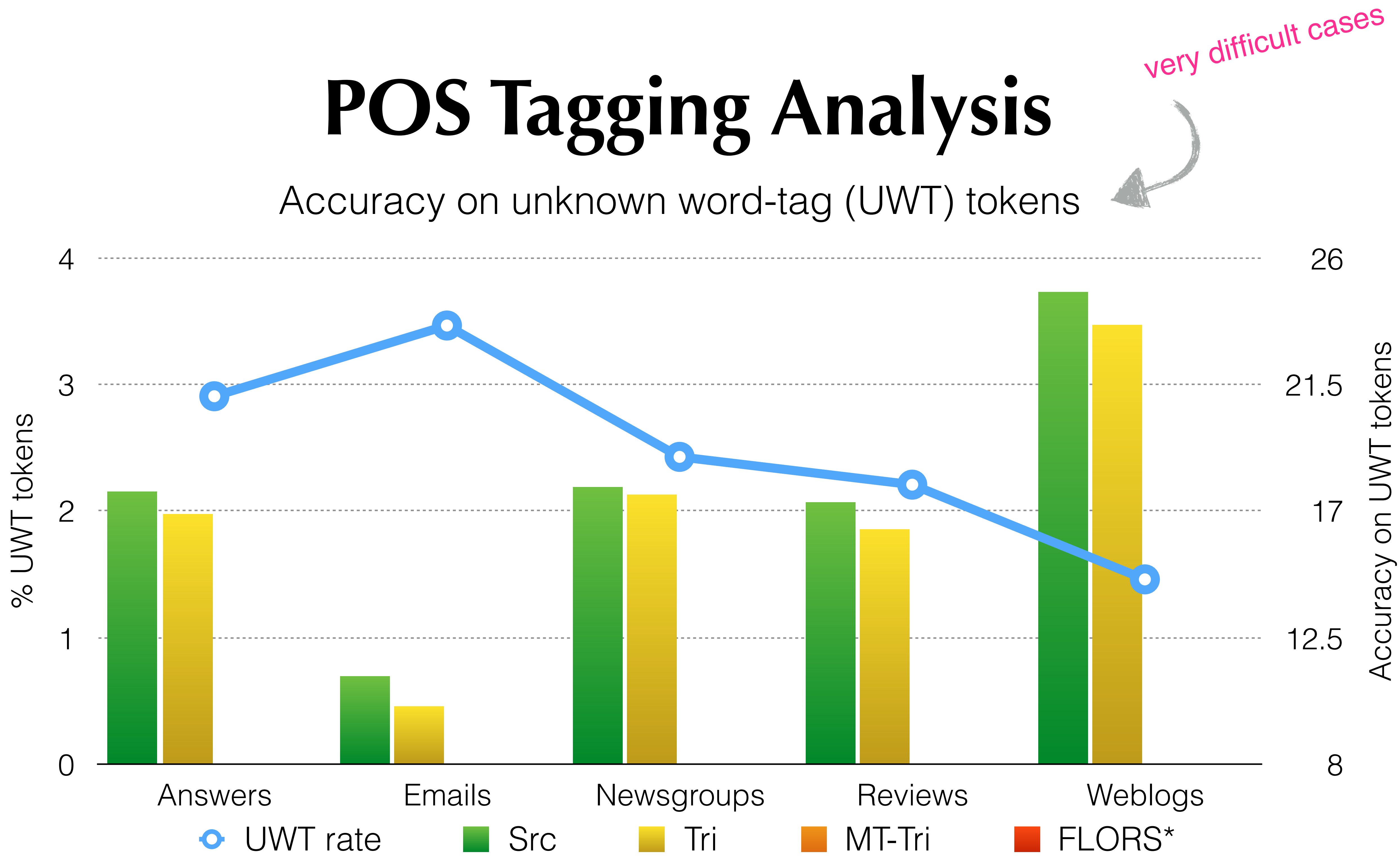
* result from Schnabel & Schütze (2014)

POS Tagging Analysis



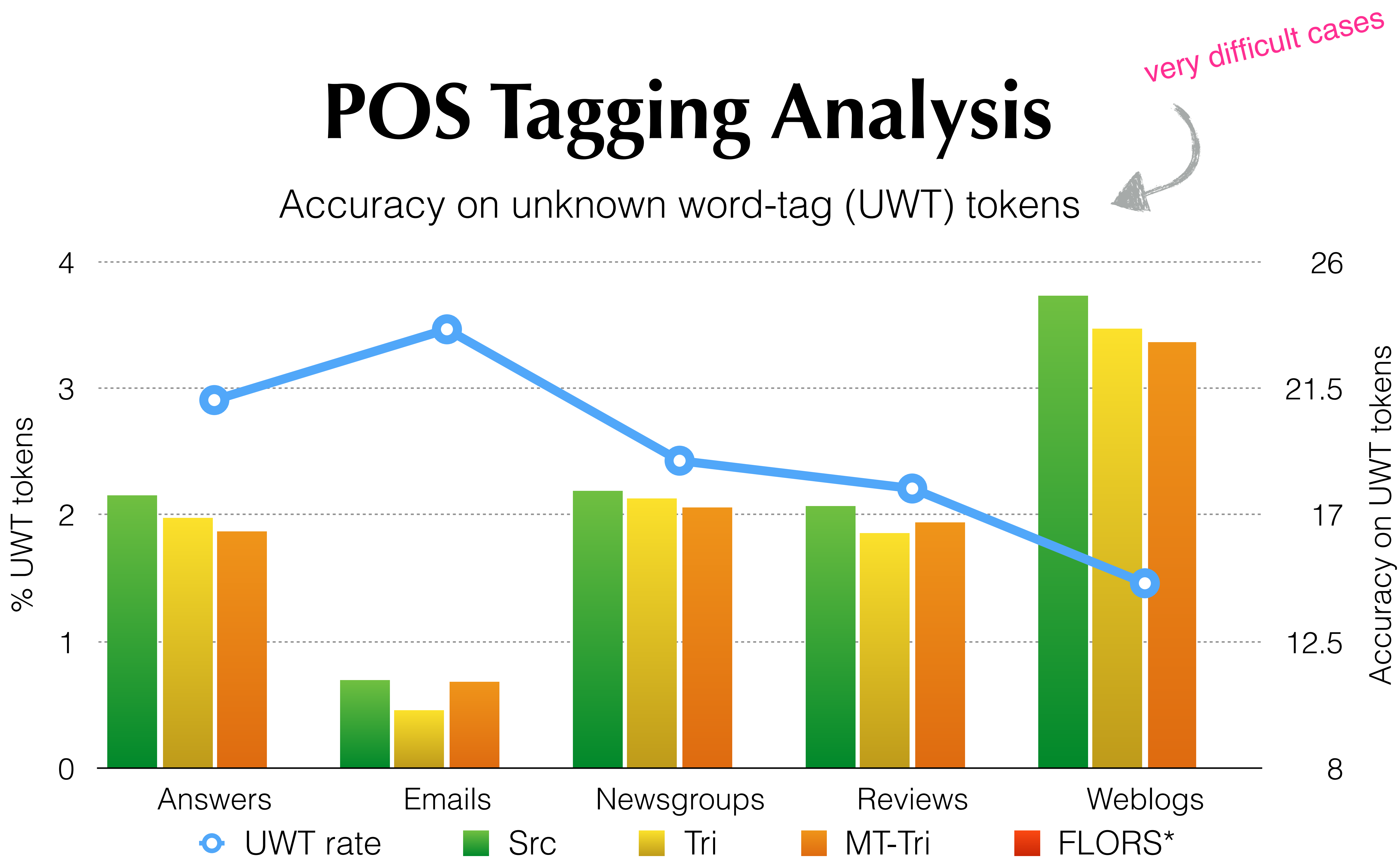
* result from Schnabel & Schütze (2014)

POS Tagging Analysis



* result from Schnabel & Schütze (2014)

POS Tagging Analysis

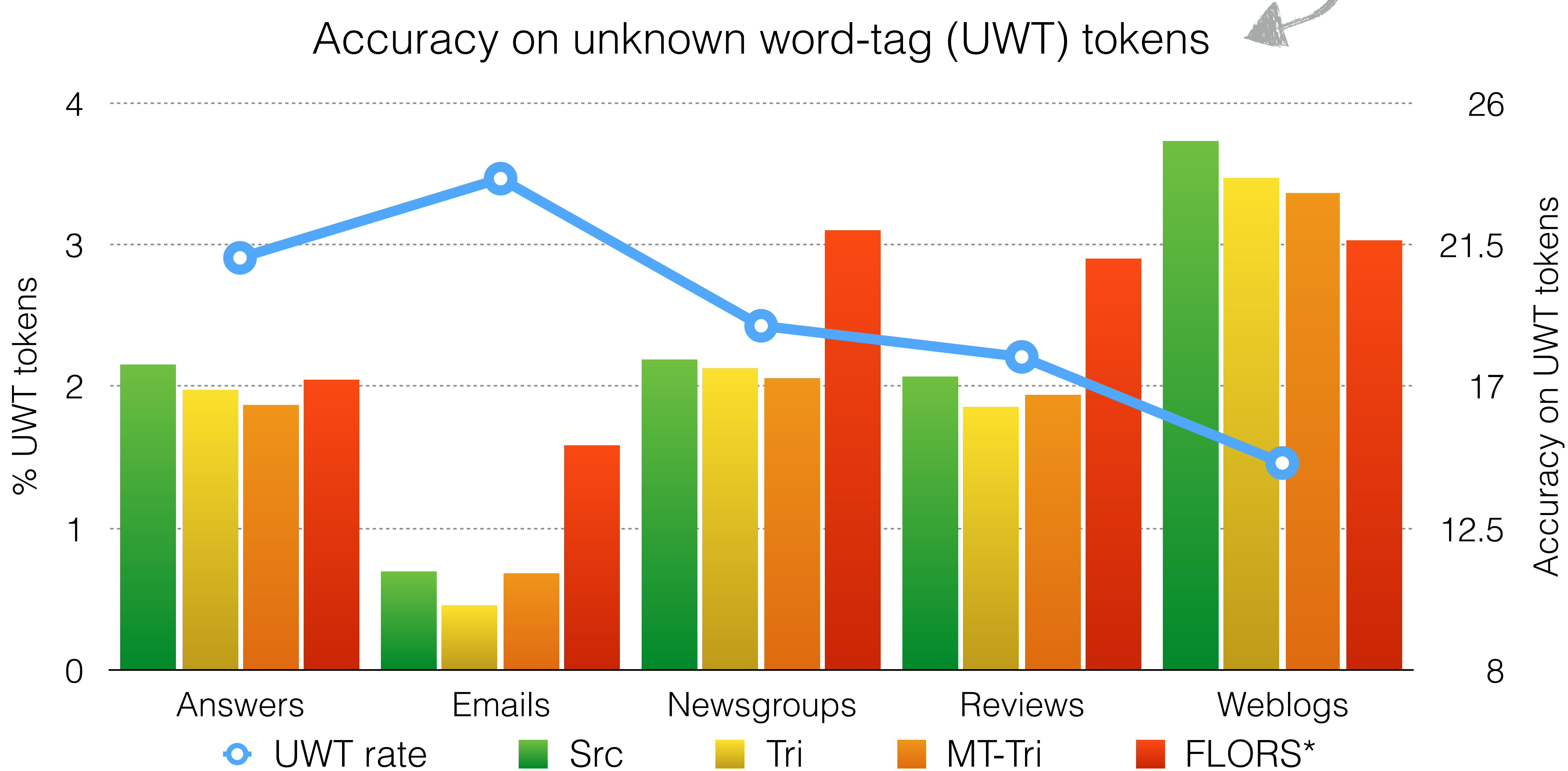


- No bootstrapping method works well on unknown word-tag combinations.

* result from Schnabel & Schütze (2014)

POS Tagging Analysis

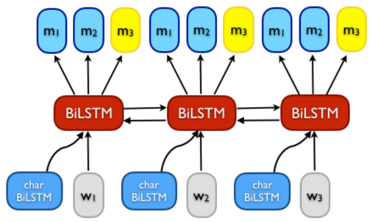
very difficult cases

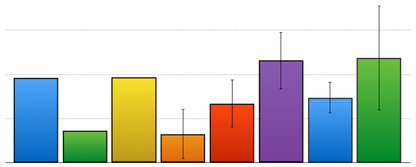


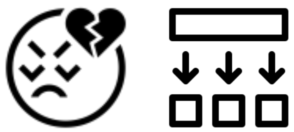
- No bootstrapping method works well on unknown word-tag combinations.
- Less lexicalized FLORS approach is superior.

* result from Schnabel & Schütze (2014)

Takeaways

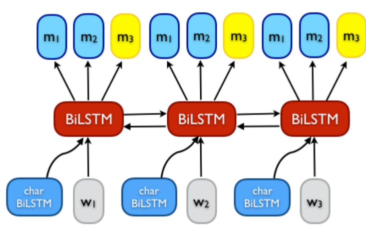


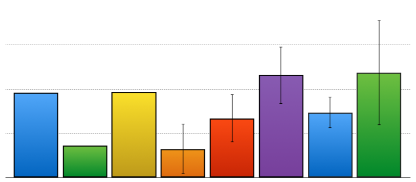


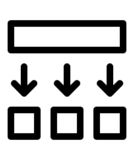


Takeaways

- **Classic tri-training works best:** outperforms recent state-of-the-art methods for sentiment analysis.

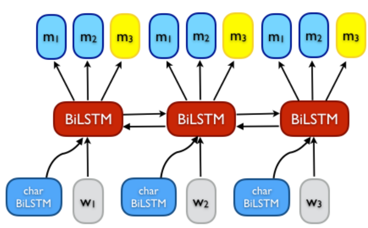




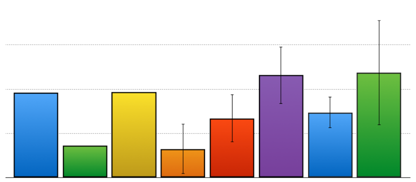


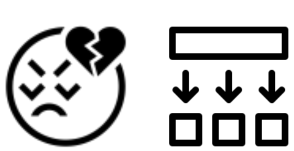
Takeaways

- **Classic tri-training works best:** outperforms recent state-of-the-art methods for sentiment analysis.



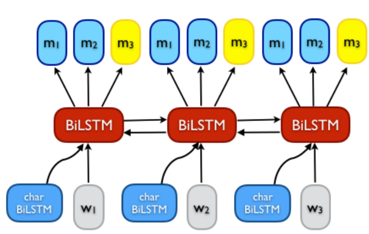
- We address the drawback of tri-training (space & time complexity) via the proposed **MT-Tri** model
  - MT-Tri works best on sentiment, but not for POS.



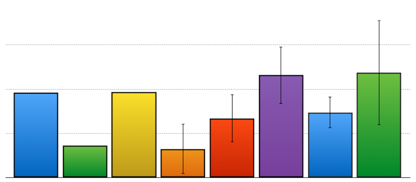


Takeaways

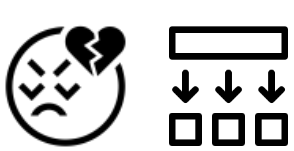
- **Classic tri-training works best:** outperforms recent state-of-the-art methods for sentiment analysis.



- We address the drawback of tri-training (space & time complexity) via the proposed **MT-Tri** model
  - MT-Tri works best on sentiment, but not for POS.
- **Importance** of:



  - **Comparing** neural methods to **classics** (**strong baselines**)



  - **Evaluation** on **multiple tasks** & **domains**