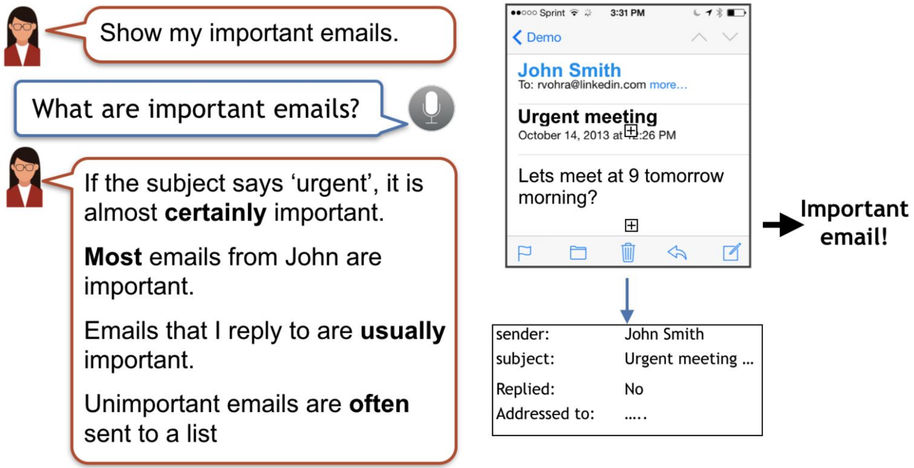
Zero-shot Learning of Classifiers from Natural Language Quantification (ACL 2018)

Shashank Srivastava, Igor Labutov, Tom Mitchell

Presenter: Jeffrey(Young-Min) Cho

February 8, 2021

Motivation



Users teach machine in language:

- Input:
  - Natural language explanations(my guidances)
  - Unlabeled instances(emails)
- Output:
  - A binary classifier(important?)

Able to classify unlabeled data

-> **Zero-Shot Learning!**

Hypothesis

Language describing concepts encodes key properties that can aid statistical learning.

Key properties:

1. Specification of relevant attributes

    (whether an email was replied to)

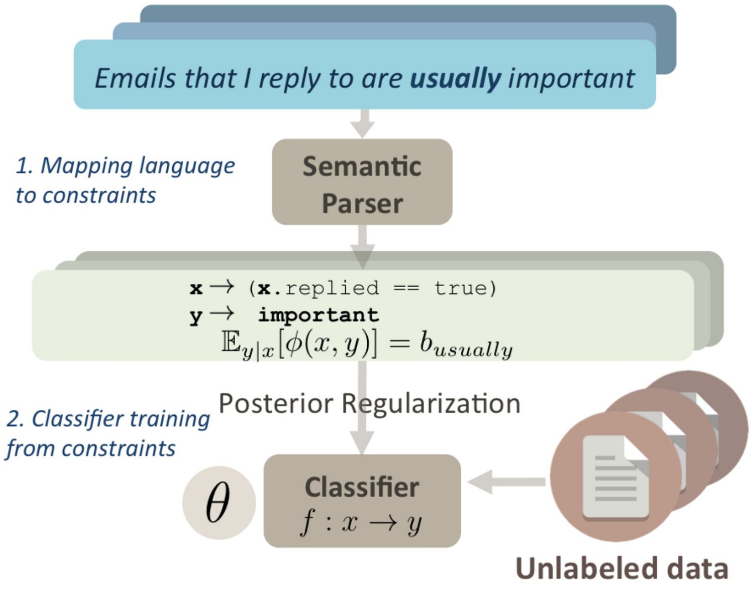
2. Relationships between such attributes and concepts labels

    (if a reply implies the class label of that email is 'important')

3. Strength of these relationships

    (via quantifiers like 'often', 'sometimes', 'rarely')

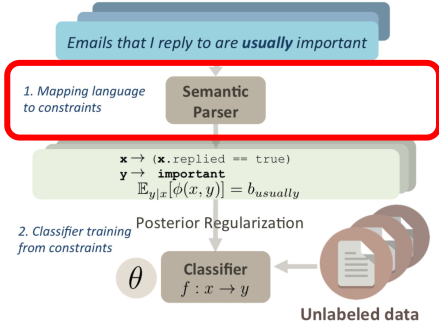
Approach



Approach to Zero-shot learning from Language:

1. Natural language explanations on how to classify concept examples are parsed into formal constraints relating features to concept labels.
2. The constraints are combined with unlabeled data, using posterior regularization to yield a classifier.

Part 1. Mapping Language to Constraints



Key challenge:

How to make this ->

Emails that I reply to are usually important.

to this? ->

$$P(\, important \mid replied : true \,) = 0.7$$

We first need to extract constraints!

Key elements

1. Mapping language to constraints

    1. **Feature** $\boldsymbol{\chi}$: observed attributes[1]

Emails that I reply to are usually important.

$$x \rightarrow \mathit{replied}\!:\!\mathit{true}$$

[1] Shashank Srivastava, Igor Labutov, and Tom Mitchell. 2017. Joint concept learning and semantic parsing from natural language explanations. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, pages 1528–1537. http://aclweb.org/anthology/D17-1161.

Key elements

1. Mapping language to constraints

2. **Concept label** y: specifying the class of instances a statement refers to

Emails that I reply to are usually $\boxed{\text{important.}}$

$$y \rightarrow positive$$

Key elements

1. Mapping language to constraints

    3. **Constraint-type**: relation between feature and concept-label

Emails that I reply to are usually important.

$$type \rightarrow y|x$$

Type	Example description
$P(y \mid x)$	Emails that I reply to are usually important
$P(x \mid y)$	I often reply to important emails
$P(y)$	I rarely get important emails

Key elements

1. Mapping language to constraints

    4. **Strength of the constraint**: specified by a quantifier, point estimate of probability

Emails that I reply to are usually important.

$$quant \rightarrow usually$$

Frequency quantifier	Probability
all, always, certainly, definitely	0.95
usually, normally, generally, likely, typically	0.70
most, majority	0.60
often, half	0.50
many	0.40
sometimes, frequently, some	0.30
few, occasionally	0.20
rarely, seldom	0.10
never	0.05

Key elements(overall)

1. Mapping language to constraints

Statement s :

Emails that I reply to are usually important.

Logical form l : $(\, x \rightarrow replied{:}true \quad y \rightarrow positive \quad type \rightarrow y|x \quad quant \rightarrow usually)$

Mathematical assertion :

$$P(\, important \mid replied : true) \;=\; 0.7$$

How to extract key elements?

1.1. Semantic Parser

Goal: predict l that best represents s -> train $P(l|s)$

Decomposition to three components:

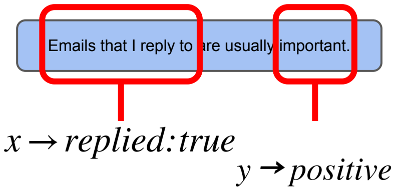
(i) probability of observing a feature and concept labels l_{xy} based on the text of the sentence

(ii) probability of the type of the assertion l_{type} based on the identified feature, concept label and syntactic properties of the sentence s

(iii) identifying the linguistic quantifier, l_{quant}, in the sentence.

$$P(l \mid s) = P(l_{xy}|s)\, P(l_{type}|l_{xy}, s)\, P(l_{quant}|s)$$

How to extract key elements?



1.1. Semantic Parser components

$P(l_{xy} \mid s)$: Identifying features and concept labels

1. Presume a linear score $S(s, l_{xy}) = w^T \Psi(s, l_{xy})$
   a. $\Psi(s, l_{xy}) \in \mathbb{R}^n$: features depend on both the sentence and the partial logical form
   b. $w^T \in \mathbb{R}^n$: parameter weight-vector
2. Assume a loglinear distribution over interpretations of a sentence[1] $P(l_{xy} \mid s) \propto w^T \Psi(s, l_{xy})$
   a. Can be trained via MLE
   b. Used CCG semantic parsing formalism[2]

[1] Percy Liang, Michael I Jordan, and Dan Klein. 2011. Learning dependency-based compositional semantics. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1. Association for Computational Linguistics, pages 590–599.
[2] Luke S Zettlemoyer and Michael Collins. 2007. Online learning of relaxed ccg grammars for parsing to logical form. In EMNLP-CoNLL. pages 678–687.

How to extract key elements?

Emails that I reply to are usually important.

$type \rightarrow y|x$

1.1. Semantic Parser components

$P(l_{type} \mid l_{xy}, s)$: Identifying assertion type, by training a Maximum Entropy Classifier.

Features:

1. Boolean value, whether feature x precedes label y
2. Boolean value, if sentence is passive(rather than active) voice
3. Boolean value, whether x is a noun, or a verb
4. Features indicating the occurrence of conditional tokens('if', 'then', and 'that') preceding or following feature x and y
5. Features indicating presence of a linguistic quantifier in a *det* or an *advmod* relation with x or y

Trained this classifier based on a manually annotated set of 80 sentences describing classes in the small UCI Zoo dataset[1]

[1] M. Lichman. 2013. UCI machine learning repository. http://archive.ics.uci.edu/ml.

How to extract key elements?

Emails that I reply to are usually important.

$quant \rightarrow usually$

1.1. Semantic Parser components

$P(\, l_{quant} \mid s)$: Identifying quantifiers

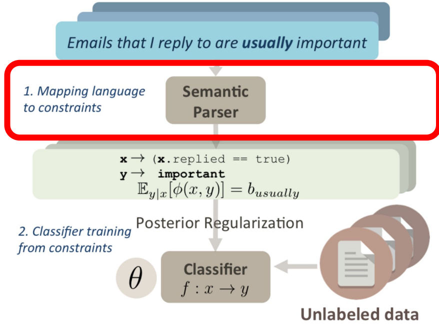
1. Only look for the first occurrence of a linguistic quantifier in a sentence
2. Ignore statements which lack an explicit quantifier in training

    eg) 'Emails from my boss are important'

3. Decouple quantification from logical representation(e.g lambda calculus)

    Irrespective at the cost of linguistic coarseness

Result of Part 1. Semantic Parser



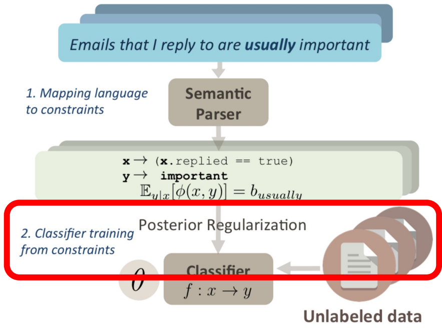
1.1. Semantic Parser

Statement S : Emails that I reply to are usually important.

Logical form l : $(\, x \rightarrow replied{:}true \quad y \rightarrow positive \quad type \rightarrow y|x \quad quant \rightarrow usually)$

Mathematical assertion : $P(\, important \mid replied : true\,) = 0.7$

Part 2. Classifier training from constraints



Target:

Predict **unobserved concept labels** $\left(Y=\{y_1 \ldots y_n\}\right)$, from **unlabeled examples** $\left(X=\{x_1 \ldots x_n\}\right)$ agree with **human-provided advice**

Solution:

Training classifier using **Posterior Regularization** framework

Posterior Regularization

2. Classifier training from constraints

Training classifier using **Posterior Regularization** framework

$$J_Q(\theta) = \boxed{L(\theta)} - \boxed{min_{q \in Q} KL\Big(q \mid p_\theta(Y|X)\Big)}$$

(1) (2)

(1) **Likelihood Term**: how well does a model θ explain the data
(2) **KL-divergence Term**: how far it is from the set Q (human advice)

Optimizing the objective reflects a tension between choosing models that **increase data likelihood**, and **emulating language advice**. (EM)

Posterior Regularization

2. Classifier training from constraints

$$J_Q(\theta) = L(\theta) - min_{q \in \boxed{Q}} KL\left(q \mid p_\theta(Y|X) \right)$$

Q : set of preferred posterior distributions over latent variables Y

$$Q := \left\{ q_X(Y) : \mathbb{E}_q\left[\phi(X, Y) \right] \leq b \right\}$$

Each parsed statement defines a probabilistic constraint,

The conjunction of all constraints defines Q

(representing models that exactly agree with human-provided advice)

Posterior Regularization

2. Classifier training from constraints

How to convert constraints adaptable to PR?

Type	Example description	Conversion to Expectation Constraint
$P(y \mid x)$	Emails that I reply to are usually important	$\mathbb{E}[\mathbb{I}_{y=important,reply(x):true}] - p_{usually} \times \mathbb{E}[\mathbb{I}_{reply(x):true}] = 0$
$P(x \mid y)$	I often reply to important emails	$\mathbb{E}[\mathbb{I}_{y=important,reply(x):true}] - p_{often} \times \mathbb{E}[\mathbb{I}_{y=important}] = 0$
$P(y)$	I rarely get important emails	Same as $P(y\|x_0)$, where x_0 is a constant feature

Each constraint type can be converted in an equivalent form $\mathbb{E}_q[\phi(X,Y)] = b$

e.g)

$$P(y = important \mid replied : true) = p_{usually}$$

$$\frac{\sum_i \mathbb{E}[\mathbb{I}_{y_i = important,\ replied:true}]}{\sum_i \mathbb{E}[\mathbb{I}_{y_i = replied:true}]} = p_{usually}$$

$$\sum_i \mathbb{E}[\mathbb{I}_{y_i = important,\ replied:true}] = p_{usually} \times \sum_i \mathbb{E}[\mathbb{I}_{y_i = replied:true}]$$

Posterior Regularization

2. Classifier training from constraints

$$J_Q(\theta) = L(\theta) - \min_{q \in Q} KL\left(q \,|\, \boxed{p_\theta(Y|X)}\right)$$

$p_\theta(Y|X)$: loglinear parametrization for the concept classifier

$$p_\theta\left(y_i|x_i\right) \propto exp(y\theta^T x)$$

Training Classifier

2. Classifier training from constraints

Solve a relaxed version of the optimization using EM algorithm, that allows slack variables, and modifies the PR objective with a L2 regularizer[1]:

$$J'(\theta, q) = L(\theta) - KL\Big(q \mid p_\theta(Y|X)\Big) - \lambda \left|\left|\mathbb{E}_q\big[\phi(X, Y)\big] - b\right|\right|^2$$

This allows solutions even when the problem is over-constrained, and the set Q is empty(due to contradictory advice)

[1] Kuzman Ganchev, Jennifer Gillenwater, Ben Taskar, et al. 2010. Posterior regularization for structured latent variable models. Journal of Machine Learning Research 11(Jul):2001–2049.

Training Classifier

2. Classifier training from constraints

The key step in the training is the computation of the posterior regularizer in the E-step:

$$\underset{q}{argmin}\ KL\Big(q \mid p_\theta(Y|X)\Big) + \lambda\Big|\Big|\mathbb{E}_q\big[\phi(X, Y)\big] - b\Big|\Big|^2$$

This objective is strictly convex, and all constraints are linear in q.

The minimization problem in the E-step can be efficiently solved through gradient steps in the dual space[1].

[1] Kedar Bellare, Gregory Druck, and Andrew McCallum. 2009. Alternating projections for learning with expectation constraints. In Proceedings of the TwentyFifth Conference on Uncertainty in Artificial Intelligence. AUAI Press, pages 43–50.

Training Classifier

2. Classifier training from constraints

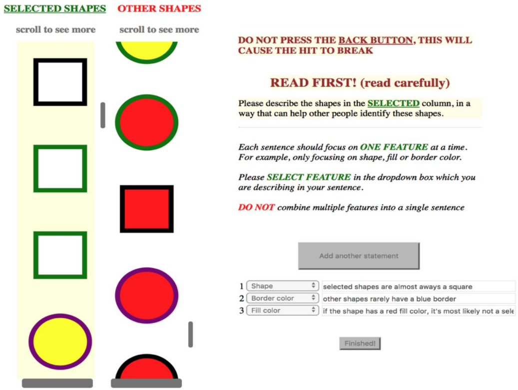
In M-step, update the model parameters for the classifier based on label distributions q estimated in the E-step.

This simply reduces to estimating the parameters θ for the logistic regression classifier, when class label probabilities are known.

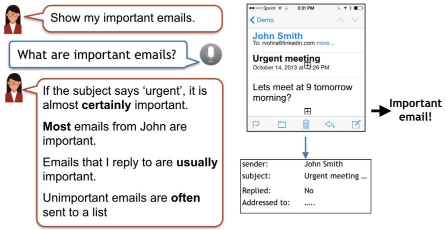
The paper run EM for 20 iterations, $\lambda = 0.1$

Datasets

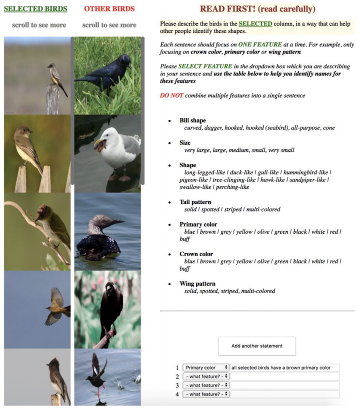
Shapes



Emails



Birds[1]



Example of explanation:

- If a shape doesn't have a blue border, it is probably not a selected shape.
- Selected shapes occasionally have a yellow fill.

- Emails that mention the word 'meet' in the subject are usually meeting requests
- Personal reminders almost always have the same recipient and sender

- A specimen that has a striped crown is likely to be a selected bird.
- Birds in the other category rarely ever have dagger-shaped beaks

Labels:

- selected/not selected, like/don't like, ...

- important/not important, meeting/not meeting, reminders/not reminders, ...

- selected/not selected, category/not category, ...

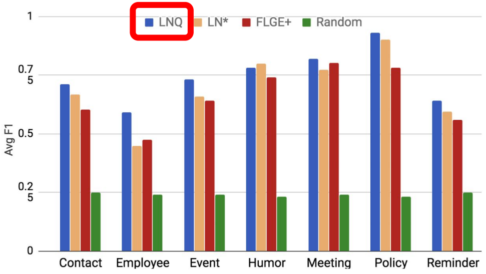
[1] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. 2011. The Caltech-UCSD Birds-200-2011 Dataset. Technical report.

Result

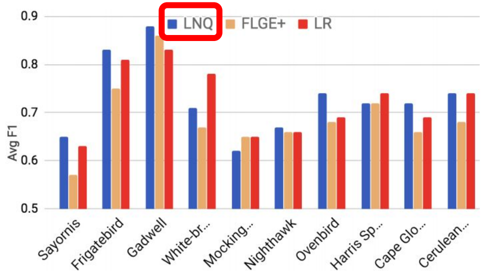
Shapes

Approach	Avg Accuracy	Labels	Descriptions
LNQ	0.751	no	yes
Bayes Optimal	0.831	–	–
FLGE+	0.659	no	yes
FLGE	0.598	no	yes
LR	0.737	yes	no
Random	0.524	–	–
Ablation:			
LNQ (coarse quant)	0.679	no	yes
LNQ (no quant)	0.545	no	yes
Human:			
Human teacher	0.802	yes	writes
Human learner	0.734	no	yes

Emails



Birds



Baseline:

FLGE: Feature Labeling through Generalized Expectation criterion[1][2]
LN*: LNQ without quantification[3]
LR : Logistic Regression trained on n=8-10 random labeled instances

[1] Gregory Druck, Gideon Mann, and Andrew McCallum. 2008. Learning from labeled features using generalized expectation criteria. In Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval. ACM, pages 595–602.
[2] Gideon S Mann and Andrew McCallum. 2010. Generalized expectation criteria for semi-supervised learning with weakly labeled data. Journal of machine learning research 11(Feb):955–984.
[3] Shashank Srivastava, Igor Labutov, and Tom Mitchell. 2017. Joint concept learning and semantic parsing from natural language explanations. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, pages 1528–1537. http://aclweb.org/anthology/D17-1161.

Conclusion

Main achievement: Zero-Shot Learning classifier from free language!

Discussion(potential improvements):

1. Modifiers('very likely'), nested quantification
2. Context based quantifier semantics
    a. Distribution, not point estimation
3. Task specific(not universal)
4. Rare language
5. Binary classification

Thank you!