Market Research, Deep Learning, and Quantification

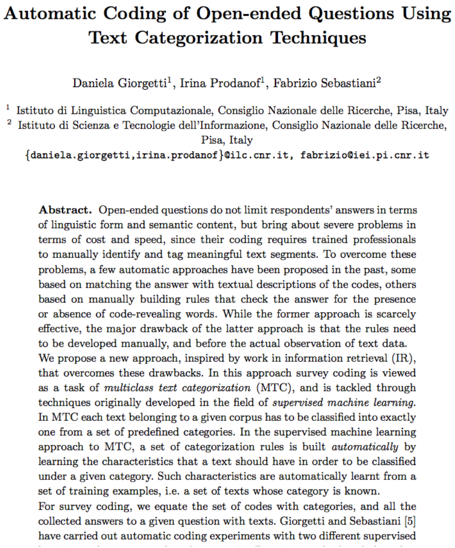
Andrea Esuli, Alejandro Moreo, Fabrizio Sebastiani

Human Language Technologies Group
Istituto di Scienza e Tecnologie dell'Informazione
Consiglio Nazionale delle Ricerche
56124 Pisa, Italy

ASC One-Day Conference
London, UK – November 15, 2018

Download these slides at http://goo.gl/JvWU7A

Machine Learning in Automated Survey Coding



Automatic Coding of Open-ended Questions Using Text Categorization Techniques

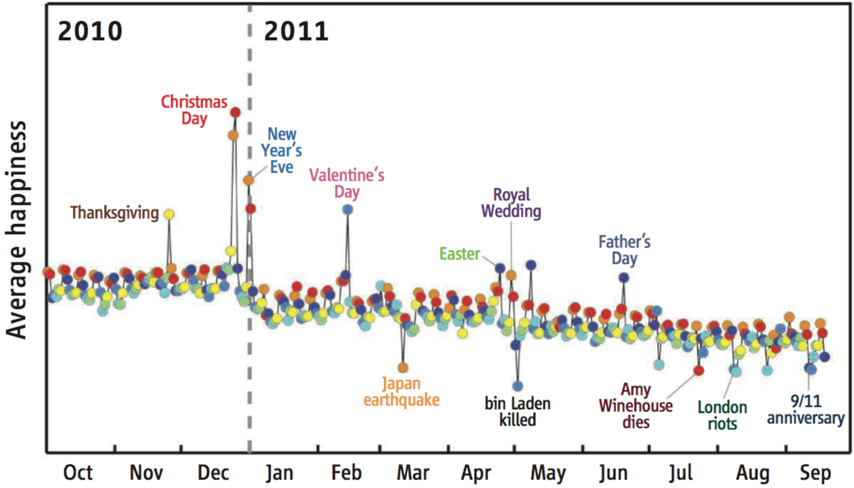
Daniela Giorgetti[1], Irina Prodanof[1], Fabrizio Sebastiani[2]

[1] Istituto di Linguistica Computazionale, Consiglio Nazionale delle Ricerche, Pisa, Italy
[2] Istituto di Scienza e Tecnologie dell'Informazione, Consiglio Nazionale delle Ricerche, Pisa, Italy
{daniela.giorgetti,irina.prodanof}@ilc.cnr.it, fabrizio@iei.pi.cnr.it

Abstract. Open-ended questions do not limit respondents' answers in terms of linguistic form and semantic content, but bring about severe problems in terms of cost and speed, since their coding requires trained professionals to manually identify and tag meaningful text segments. To overcome these problems, a few automatic approaches have been proposed in the past, some based on matching the answer with textual descriptions of the codes, others based on manually building rules that check the answer for the presence or absence of code-revealing words. While the former approach is scarcely effective, the major drawback of the latter approach is that the rules need to be developed manually, and before the actual observation of text data.
We propose a new approach, inspired by work in information retrieval (IR), that overcomes these drawbacks. In this approach survey coding is viewed as a task of *multiclass text categorization* (MTC), and is tackled through techniques originally developed in the field of *supervised machine learning*. In MTC each text belonging to a given corpus has to be classified into exactly one from a set of predefined categories. In the supervised machine learning approach to MTC, a set of categorization rules is built *automatically* by learning the characteristics that a text should have in order to be classified under a given category. Such characteristics are automatically learnt from a set of training examples, i.e. a set of texts whose category is known.
For survey coding, we equate the set of codes with categories, and all the collected answers to a given question with texts. Giorgetti and Sebastiani [5] have carried out automatic coding experiments with two different supervised

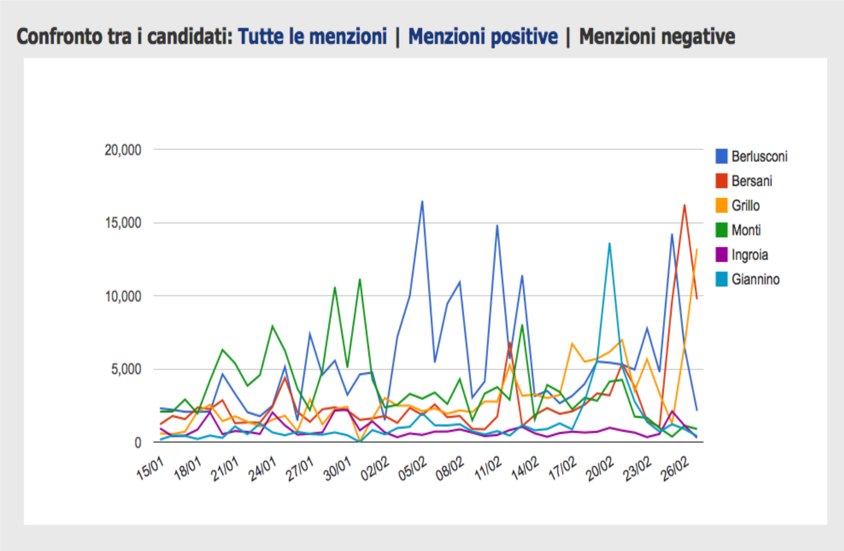
What is quantification?

[1]



[1] Dodds, Peter et al. Temporal Patterns of Happiness and Information in a Global Social Network: Hedonometrics and Twitter. *PLoS ONE*, 6(12), 2011.

What is quantification? (cont'd)



Confronto tra i candidati: Tutte le menzioni | Menzioni positive | Menzioni negative

What is quantification? (cont'd)

- In many applications of classification (a.k.a. coding), coding individual items is only an intermediate step, and the real goal is determining the relative frequency (or: prevalence) of each class in the uncoded ("unlabelled") data.
- In machine learning and data mining this is called quantification, or supervised prevalence estimation

What is quantification? (cont'd)

- E.g.
  - Among the tweets concerning the next presidential elections, what is the percentage of pro-Democrat ones?
  - Among the posts about the Apple Watch 4 on forum X, what is the percentage of "very negative" ones?
  - How have these percentages have evolved over time?
- As in classification, quantification may come in binary / multi-label multi-class / single-label multi-class / ordinal form
- This task has been studied within ML and DM, and has given rise to learning methods specific to it
- We will deal with text quantification

What is quantification? (cont'd)

- Example 1 (CRM):

  > "How satisfied are you with our online bank account?"

  Class of interest: `MayDefectToCompetition`
  Goal: classification (at the individual level)

What is quantification? (cont'd)

- Example 1 (CRM):

  > "How satisfied are you with our online bank account?"

  Class of interest: `MayDefectToCompetition`
  Goal: classification (at the individual level)

- Example 2 (MR):

  > "What do you think about adding onions to cheeseburgers?"

  Class of interest: `LovesOnionsInCheeseburgers`
  Goal: quantification (at the aggregate level)

Applications of quantification

- A number of fields where classification is used are not interested in individual data, but in data aggregated across spatio-temporal contexts and according to other variables (e.g., gender, age group, religion, job type, ...); e.g.,
  - Social sciences
  - Political science
  - Epidemiology
  - Logistics

Applications of quantification

- A number of fields where classification is used are not interested in individual data, but in data aggregated across spatio-temporal contexts and according to other variables (e.g., gender, age group, religion, job type, ...); e.g.,
  - Social sciences
  - Political science
  - Epidemiology
  - Logistics

"We are not interested in finding the needle in the haystack, we are interested in characterising the haystack!"

Applications of quantification

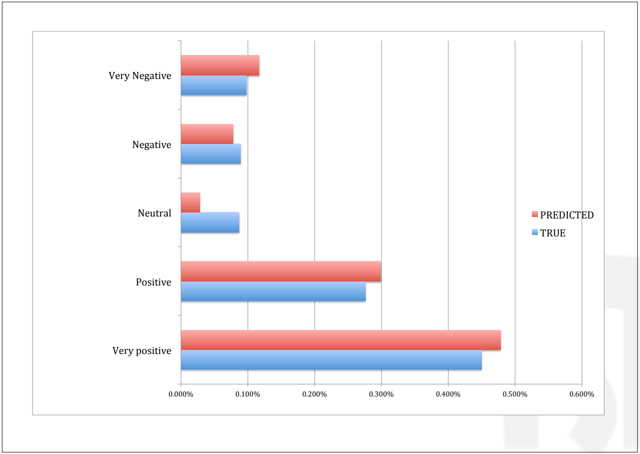
- A number of fields where classification is used are not interested in individual data, but in data aggregated across spatio-temporal contexts and according to other variables (e.g., gender, age group, religion, job type, ...); e.g.,
  - Social sciences
  - Political science
  - Epidemiology
  - Logistics

> "We are not interested in finding the needle in the haystack, we are interested in characterising the haystack!"

- When using supervised ML, monitoring class prevalences across conditions (e.g., time) different from those that held while generating the training data, is of key importance

What is quantification? (cont'd)

- Quantification may be also defined as the task of approximating a true distribution by a predicted distribution



Distribution drift

- The need to perform quantification arises because of distribution drift, i.e., the presence of a discrepancy between the class distribution of *Tr* and that of *Te*.

- Distribution drift may derive when
    1. the environment is not stationary across time and/or space and/or other variables, and the testing conditions are irreproducible at training time
    2. the process of labelling training data is class-dependent (e.g., “stratified” training sets)
    3. the labelling process introduces bias in the training set (e.g., if “active learning” is used)

- Distribution drift clashes with the IID assumption, on which standard ML algorithms are instead based.

The “paradox of quantification”

- Is “classify and count” the optimal quantification strategy? No!
- A perfect classifier is also a perfect “quantifier” (i.e., estimator of class prevalence), but ...
- ... a good classifier is not necessarily a good quantifier (and vice versa) :

	FP	*FN*
Classifier A	5	18
Classifier B	19	21

- Paradoxically, we should choose quantifier B rather than quantifier A, since A is biased
- This means that quantification should be studied as a task in its own right

Why "Classify and Count" does not work (1)

Vladimir N. Vapnik (1936 –)

"If you possess a restricted amount of information for solving some problem, try to solve the problem directly and never solve a more general problem as an intermediate step. It is possible that the available information is sufficient for a direct solution but is insufficient for solving a more general intermediate problem."

- Classification is a more general problem than quantification!

Why "Classify and Count" does not work (2)

- Explicit Loss Minimisation: Modern learning algorithms "are aware of" the accuracy measure used to evaluate the results

- Classification and quantification have different accuracy measures; e.g.,

$$\begin{aligned} \text{Classification:} \quad F_1 &= \frac{2 \cdot TP}{2 \cdot TP + FN + FP} \\ \text{Quantification:} \quad AE &= \frac{1}{n} \sum_{i=1}^{n} |p(c) - \hat{p}(c)| \end{aligned}$$

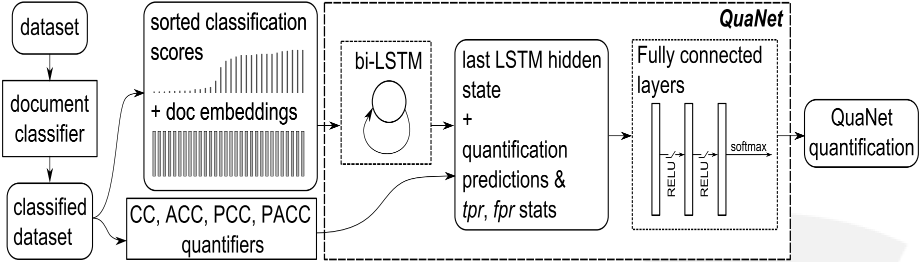
- A classifier trained via traditional learning methods "goes for" a classification accuracy measure, not for a quantification accuracy one!

Experiments

- Several quantification methods have been proposed in the last 10 years; we here only discuss one of them
- We run binary experiments where we test the ability of a system to correctly guess the value of Pr(Positive)
- We compare two systems, i.e.,
  1. a CC method based on a state-of-the-art classifier; for this we choose a deep learning method based on ("LSTM") recurrent neural networks
  2. a state-of-the-art quantification method; for this we choose QuaNet, a deep learning quantification method also based on ("LSTM") recurrent neural networks[2]

[2] A. Esuli, A. Moreo, F. Sebastiani. A Recurrent Neural Network for Sentiment Quantification. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, 2018. https://bit.ly/2Tee0qW

QuaNet: A State-of-the-Art Quantification Method



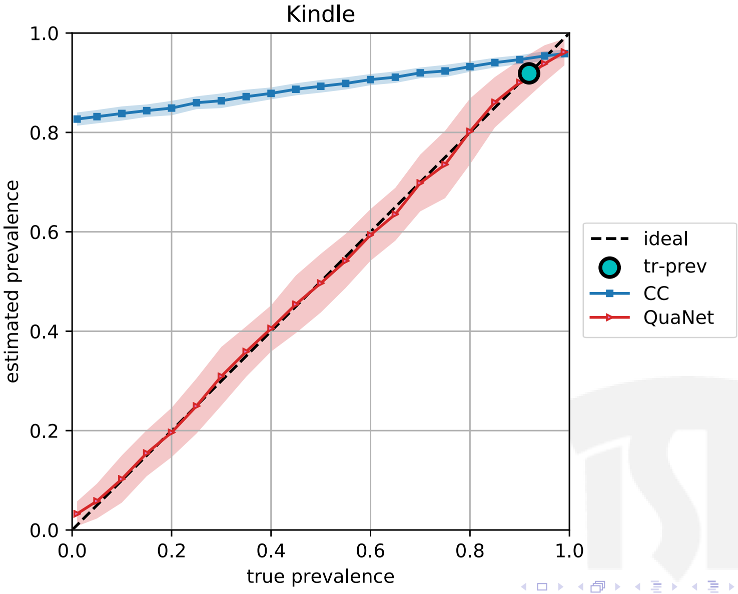
Experiments (cont'd)

- Three datasets of product reviews (Positive vs. Negative)

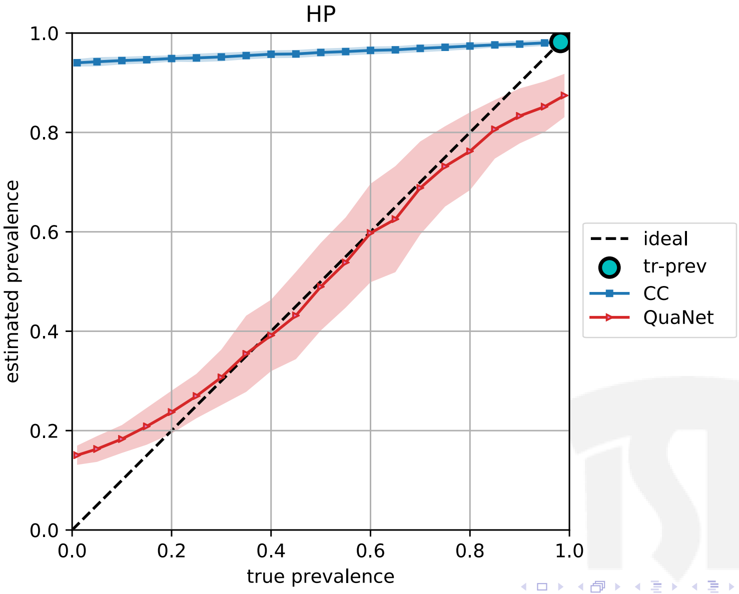
	Type	# Training	# Test	$\Pr_{Tr}$(Positive)
IMDB	Movie	25,000	25,000	0.500
Kindle (Amazon)	CE	3,821 (Aug 2010 / Oct 2010)	21,592 (Nov 2010 / Jul 2011)	0.917
HP (Amazon)	Book	9,533 (1998 / 2000)	18,401 (2001 / 2011)	0.982

- From each test set we extract 2,100 samples of 500 docs each
  - 21 different values of Pr(Positive), i.e., all values in $\{0.00, 0.05, ..., 0.95, 1.00\}$
  - 100 random samples for each such value
- We thus test the ability of a system to correctly guess the value of Pr(Positive) on samples that exhibit widely different test prevalences

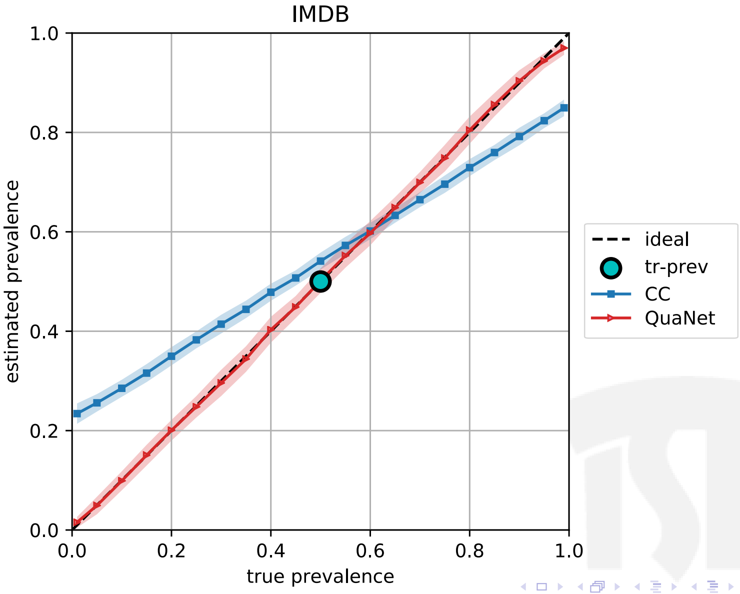
Experiments



Experiments



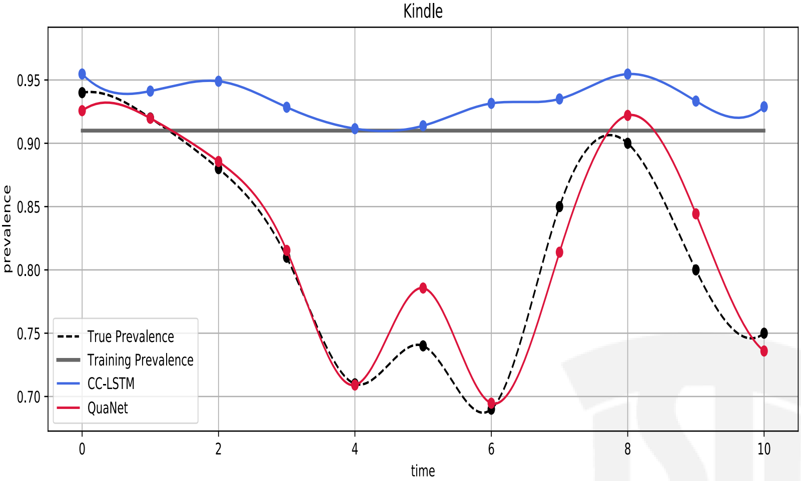
Experiments



Experiments: Overall results

		AE		*RAE*	
IMDB	CC(LSTM)	0.096	(+421%)	1.193	(+1008%)
	QuaNet(LSTM)	0.018		0.108	
Kindle	CC(LSTM)	0.417	(+585%)	5.805	(+1083%)
	QuaNet(LSTM)	0.061		0.491	
HP	CC(LSTM)	0.476	(+379%)	6.487	(+526%)
	QuaNet(LSTM)	0.099		1.036	

Monitoring class prevalences through time



Why “Classify and Count” does not work (3)

Rev. Thomas Bayes
(1701–61)

- The “Naive Bayesian Classifier”:

$$\Pr(c|\mathbf{x}) = \Pr(c) \prod_{i=1}^{n} \frac{\Pr(x_i|c)}{\Pr(x_i)}$$

- The probability $\Pr(c|\mathbf{x})$ that an uncoded item $\mathbf{x}$ is assigned to class c grows with the frequency $\Pr(c)$ of that class in the training set

Why "Classify and Count" does not work (3)

Rev. Thomas Bayes
(1701–61)

- The "Naive Bayesian Classifier":

$$\Pr(c|\mathbf{x}) = \Pr(c) \prod_{i=1}^{n} \frac{\Pr(x_i|c)}{\Pr(x_i)}$$

- The probability $\Pr(c|\mathbf{x})$ that an uncoded item **x** is assigned to class c grows with the frequency $\Pr(c)$ of that class in the training set
- A classifier thus tends to replicate, in the data it codes, the class frequencies it has been trained on

The Takeaway Message

"When I need to automatically code data,
what do I really care about?"

The Takeaway Message

“When I need to automatically code data,
what do I really care about?”

- A: The codes assigned to the individual unlabelled data

  ⇒ Use a (standard) classification method!

The Takeaway Message

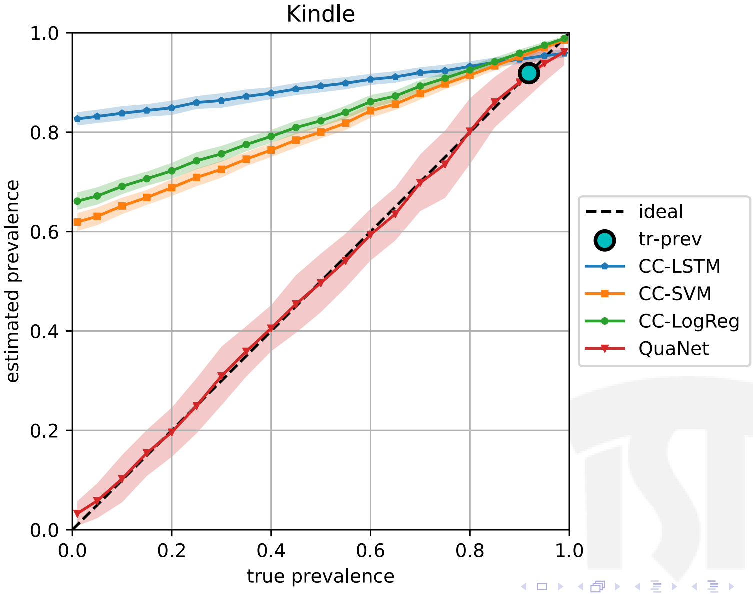
"When I need to automatically code data,
what do I really care about?"

- A: The codes assigned to the individual unlabelled data
  ⇒ Use a (standard) classification method!
- A: The prevalences of the codes in the unlabelled data
  ⇒ Use a real quantification method!

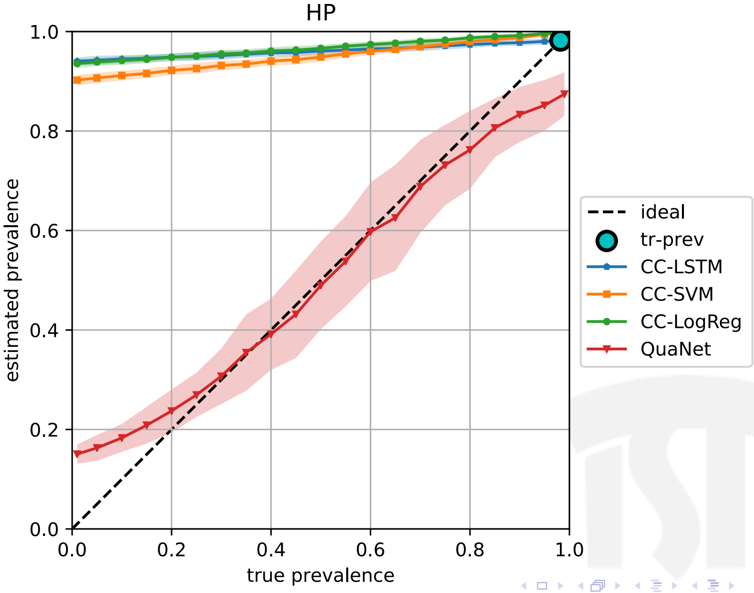
Thank you!

For any question, Skype me at `fabseb60`

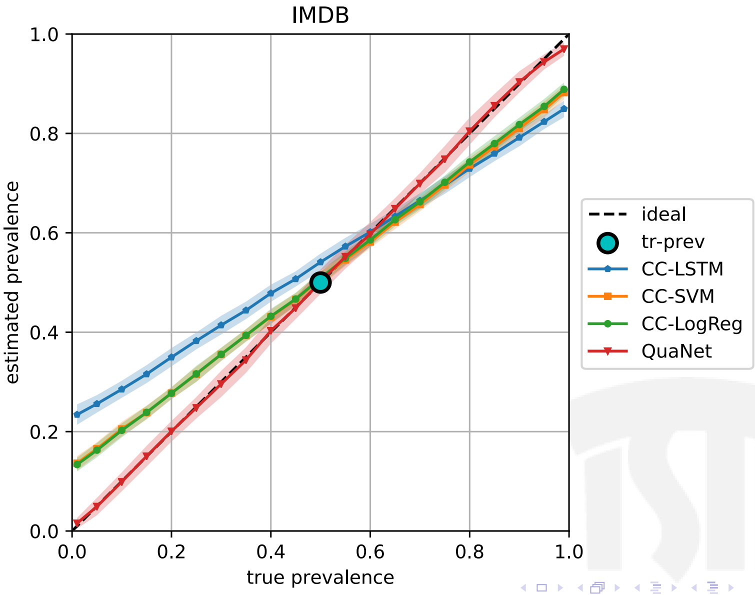
Experiments: Additional CC methods



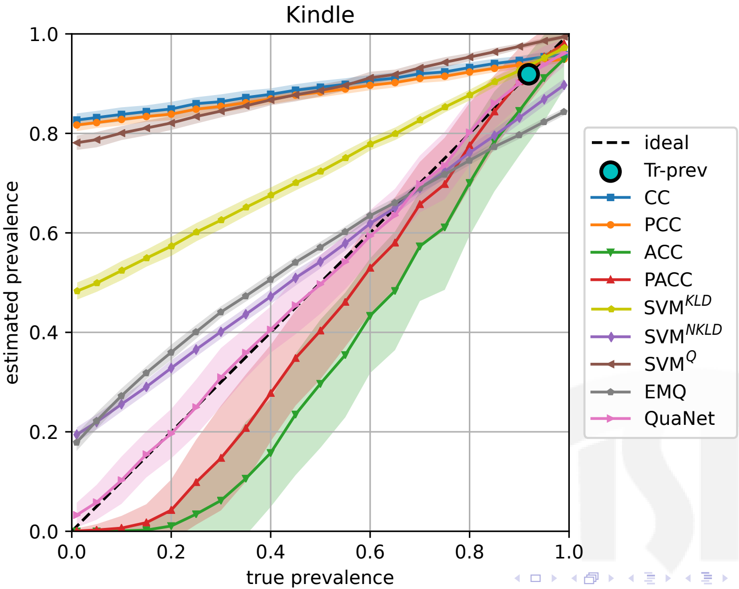
Experiments: Additional CC methods



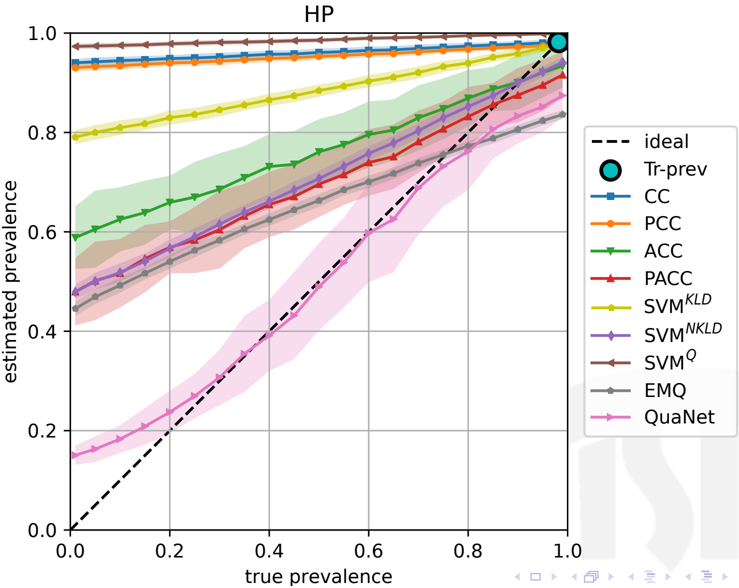
Experiments: Additional CC methods



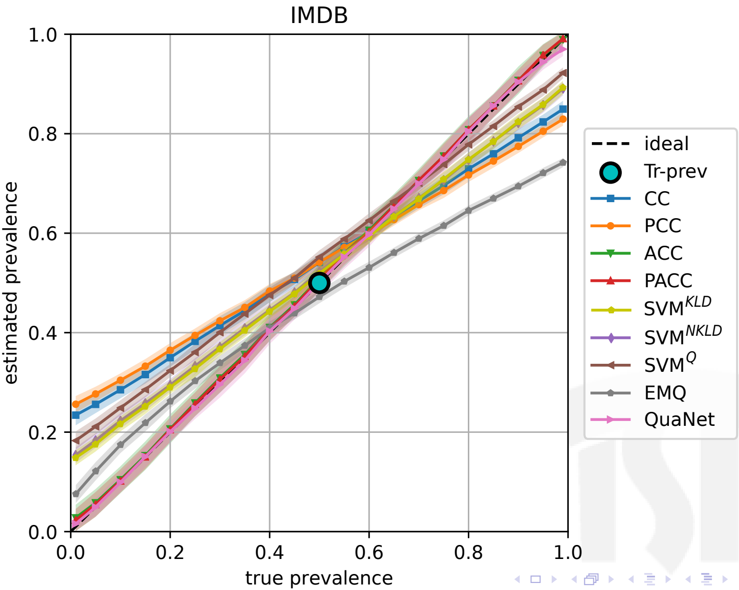
Experiments: Additional quantification methods



Experiments: Additional quantification methods



Experiments: Additional quantification methods


IMDB
estimated prevalence
true prevalence
ideal
Tr-prev
CC
PCC
ACC
PACC
SVM^{KLD}
SVM^{NKLD}
SVM^{Q}
EMQ
QuaNet