Speeding up the Manifesto Project: Active learning strategies for efficient automated political annotations

Felix Biessmann[1], Philipp Schmidt[2]

[1] felix.biessmann@gmail.com
[2] schmidtiphil@gmail.com

Disclaimers

- (For us) This open source project is a hobby
- It has nothing to do with our job
- Apologies if we missed to cite somebody in this room
- We'd be excited to hear about more related work

Motivation

- Automated political analysis required for
  - Political scientists
  - Journalists
  - Average media consumer
- ML models need in-domain training data [3]
- But annotation budget is often limited:
  - Temporal constraints (before elections) [4]
  - Online news media (too much content)

→ How to choose which texts to annotate?

[3] Bießmann [2016]

[4] Merz [2017], Kühne et al. [2017]

Motivation

- Automated political analysis required for
  - Political scientists
  - Journalists
  - Average media consumer
- ML models need in-domain training data [3]
- But annotation budget is often limited:
  - Temporal constraints (before elections) [4]
  - Online news media (too much content)

→ How to choose which texts to annotate?

[3] Bießmann [2016]

[4] Merz [2017], Kühne et al. [2017]

Motivation

- Automated political analysis required for
  - Political scientists
  - Journalists
  - Average media consumer
- ML models need in-domain training data [3]
- But annotation budget is often limited:
  - Temporal constraints (before elections) [4]
  - Online news media (too much content)

→ How to choose which texts to annotate?

[3] Bießmann [2016]

[4] Merz [2017], Kühne et al. [2017]

Motivation

- Automated political analysis required for
  - Political scientists
  - Journalists
  - Average media consumer
- ML models need in-domain training data [3]
- But annotation budget is often limited:
  - Temporal constraints (before elections) [4]
  - Online news media (too much content)

→ How to choose which texts to annotate?

[3] Bießmann [2016]

[4] Merz [2017], Kühne et al. [2017]

Motivation

- Automated political analysis required for
  - Political scientists
  - Journalists
  - Average media consumer
- ML models need in-domain training data [3]
- But annotation budget is often limited:
  - Temporal constraints (before elections) [4]
  - Online news media (too much content)

→ How to choose which texts to annotate?

[3] Bießmann [2016]

[4] Merz [2017], Kühne et al. [2017]

Motivation

- Automated political analysis required for
  - Political scientists
  - Journalists
  - Average media consumer
- ML models need in-domain training data [3]
- But annotation budget is often limited:
  - Temporal constraints (before elections) [4]
  - Online news media (too much content)

→ How to choose which texts to annotate?

[3] Bießmann [2016]

[4] Merz [2017], Kühne et al. [2017]

Motivation

- Automated political analysis required for
  - Political scientists
  - Journalists
  - Average media consumer
- ML models need in-domain training data [3]
- But annotation budget is often limited:
  - Temporal constraints (before elections) [4]
  - Online news media (too much content)

→ How to choose which texts to annotate?

[3] Bießmann [2016]

[4] Merz [2017], Kühne et al. [2017]

Motivation

- Automated political analysis required for
  - Political scientists
  - Journalists
  - Average media consumer
- ML models need in-domain training data [3]
- But annotation budget is often limited:
  - Temporal constraints (before elections) [4]
  - Online news media (too much content)

→ How to choose which texts to annotate?

[3] Bießmann [2016]

[4] Merz [2017], Kühne et al. [2017]

Motivation

- Automated political analysis required for
  - Political scientists
  - Journalists
  - Average media consumer
- ML models need in-domain training data [3]
- But annotation budget is often limited:
  - Temporal constraints (before elections) [4]
  - Online news media (too much content)

→ How to choose which texts to annotate?

[3] Bießmann [2016]

[4] Merz [2017], Kühne et al. [2017]

Active Learning

- Given limited annotation budget, find the best model
- How?
  - Annotate difficult ones[5] first
- Why?
  - Intuition:
    Model learns most from difficult examples
  - Math:
    Gradient of loss function is larger for difficult examples

[5] For which model is most uncertain.

Active Learning

- Given limited annotation budget, find the best model
- How?
  - Annotate difficult ones[5] first
- Why?
  - Intuition:
    Model learns most from difficult examples
  - Math:
    Gradient of loss function is larger for difficult examples

[5] For which model is most uncertain.

Active Learning

- Given limited annotation budget, find the best model
- How?
  - Annotate difficult ones[5] first
- Why?
  - Intuition:
    Model learns most from difficult examples
  - Math:
    Gradient of loss function is larger for difficult examples

[5] For which model is most uncertain.

Active Learning

- Given limited annotation budget, find the best model
- How?
  - Annotate difficult ones[5] first
- Why?
  - Intuition:
    Model learns most from difficult examples
  - Math:
    Gradient of loss function is larger for difficult examples

[5] For which model is most uncertain.

Active Learning

- Given limited annotation budget, find the best model
- How?
  - Annotate difficult ones[5] first
- Why?
  - Intuition:
    Model learns most from difficult examples
  - Math:
    Gradient of loss function is larger for difficult examples

[5] For which model is most uncertain.

Active Learning

- Given limited annotation budget, find the best model
- How?
  - Annotate difficult ones[5] first
- Why?
  - Intuition:
    Model learns most from difficult examples
  - Math:
    Gradient of loss function is larger for difficult examples

[5] For which model is most uncertain.

Data

- All annotated German texts from: `https://manifestoproject.wzb.eu/`
- Custom python tooling for manifesto API: `https://github.com/felixbiessmann/active-manifesto`
- Only texts with more than 1000 observed labels

Model

- Preprocessing
  - Unigram Bag-of-Words features
  - Hashing Vectorizer
- Classification Model: Multinomial Logistic Regression

$$p(y = k|\mathbf{x}) = \frac{e^{z_k}}{\sum_{j=1}^{K} e^{z_j}} \text{ with } z_k = \mathbf{w}_k^\top \mathbf{x}. \tag{1}$$

With

- Labels $y \in \{1, 2, \ldots, K\}$ (manifesto code)
- $\mathbf{w}_1, \ldots, \mathbf{w}_K \in \mathbb{R}^d$ weight vectors of kth manifesto code

Offline Experiments

- Train model on 1%, 10%, 20%, . . . , 100% of training data
- Vary sampling strategies to select from unlabelled texts
- Compute accuracy on hold-out data

Active Learning Strategies

- Random Baseline: Uniform random sampling
- Uncertainty Sampling: Only top-prediction counts

$$\mathbf{x}_i = \underset{i,k}{\operatorname{argmax}} \left(1 - p(y = k|\mathbf{x}_i, \mathbf{W})\right) \tag{2}$$

- Entropy Sampling: All predictions count

$$\mathbf{x}_i = \underset{i}{\operatorname{argmax}} \sum_k p(y = k|\mathbf{x}_i, \mathbf{W}) \log(p(y = k|\mathbf{x}_i, \mathbf{W})) \tag{3}$$

- Margin Sampling: Top 2 predictions count

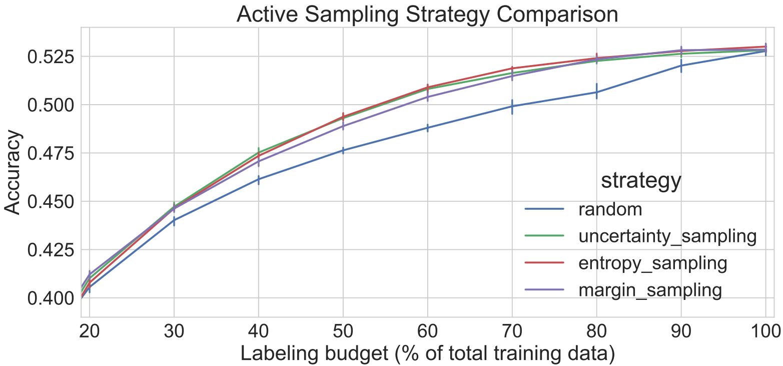
$$\mathbf{x}_i = \underset{i}{\operatorname{argmin}} \left(p(y = k_1|\mathbf{x}_i, \mathbf{W}) - p(y = k_2|\mathbf{x}_i, \mathbf{W})\right) \tag{4}$$

Results: 'Perfect' Reference Model

manifesto code	precision	recall	f1-score	support
107	0.60	0.48	0.53	774
201	0.51	0.55	0.53	1194
202	0.63	0.57	0.60	983
305	0.46	0.59	0.52	783
403	0.52	0.48	0.50	1281
411	0.39	0.60	0.47	1535
501	0.61	0.55	0.58	1380
502	0.65	0.41	0.50	587
503	0.46	0.52	0.49	2083
506	0.63	0.48	0.54	1026
605	0.56	0.44	0.49	576
701	0.59	0.39	0.47	1123
avg / total	0.50	0.48	0.48	17559

Table: Precision, recall, F1 score and number of instances per class.

Active Learning Results



Median accuracy and the 5th/95th percentile across 100 repetitions

Conclusion

- Automated political analysis requires annotations
- **Limited budged** for annotations of political texts
- Active Learning
  - Helps to select which texts to annotate
  - Perfect model with 80% of data
  - Almost perfect (over 90%) with 50% of data

→ Active learning can speed up political annotations.

- Code:
  `https://github.com/felixbiessmann/active-manifesto`

Limitations

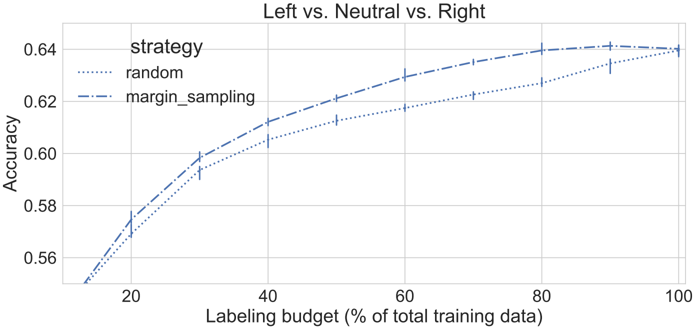
- We used a simple model
- Random sampling is an unrealistic baseline
- We only performed offline experiments

→ More convincing: online experiments

Demo http://rightornot.info

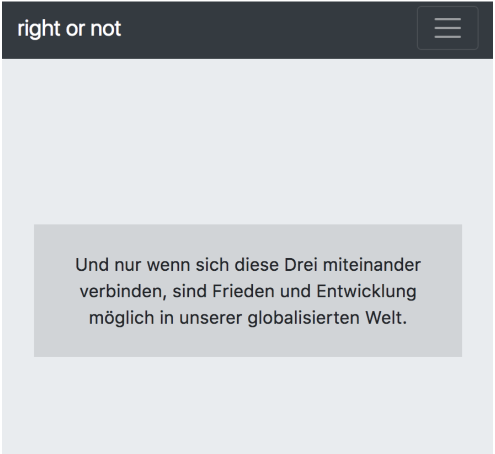
- Goal: Collect Annotations with Active Learning
  1. For political analysis of non-manifesto texts
  2. For comparing manifesto annotations with laymen judgements
- Incentive for users:
  1. Estimate your political bias
  2. Escape your political filter bubble

Demo `http://rightornot.info`



Labels: Left, Neutral, Right

Demo http://rightornot.info



Und nur wenn sich diese Drei miteinander verbinden, sind Frieden und Entwicklung möglich in unserer globalisierten Welt.

References

F. Bießmann. Automating political bias prediction. *CoRR*, abs/1608.02195, 2016. URL http://arxiv.org/abs/1608.02195.

S. Kühne, O. Schnuck, and R. Schöffel. Der computer sagt: Jamaika. *https://web.br.de/interaktiv/wahlprogramm-analyse-bundestagswahl/*, 2017.

N. Merz. Alle wahlprogramme lesen? dauert nur 17 stunden. *http://www.zeit.de/politik/deutschland/2017-08/bundestagswahl-wahlprogramme-parteien-computeranalyse*, 2017.