

ACTIVE LEARNING WITH (L)LMS: STATE OF THE ART AND PRACTICAL CHALLENGES

Christopher Schröder

ScaDS.AI (Center for Scalable Data Analytics and Artificial Intelligence)

Leipzig – Dec 12th, 2023

AGENDA

Part 1: Active Learning for Text Classification

- Introduction to Active Learning and Text Classification
- State of the Art and Recent Trends

Part 2: Small-Text: Active Learning for Text Classification

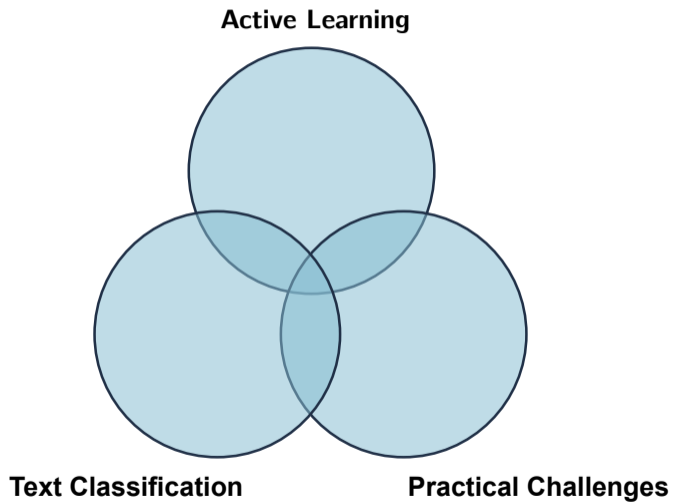
- Introducing the `small-text` library
- Code Example: Active Learning to Build a News Classifier

Part 3: Practical Challenges when using Active Learning with LLMs

- Practical Challenges

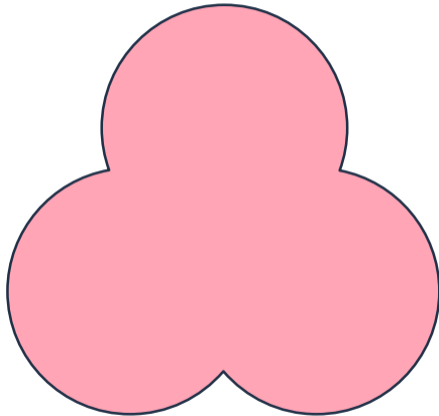
PART 1: ACTIVE LEARNING FOR TEXT CLASSIFICATION

THIS PRESENTATION

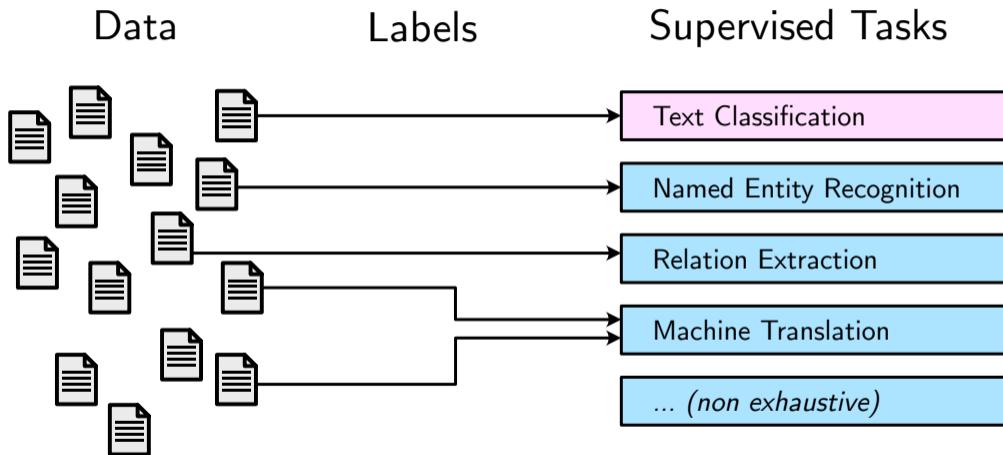


THIS PRESENTATION

Large Language Models



SUPERVISED NATURAL LANGUAGE PROCESSING



SUPERVISED NATURAL LANGUAGE PROCESSING

Data



Reality



Supervised Tasks

Text Classification

Named Entity Recognition

Relation Extraction

Machine Translation

... (*non exhaustive*)

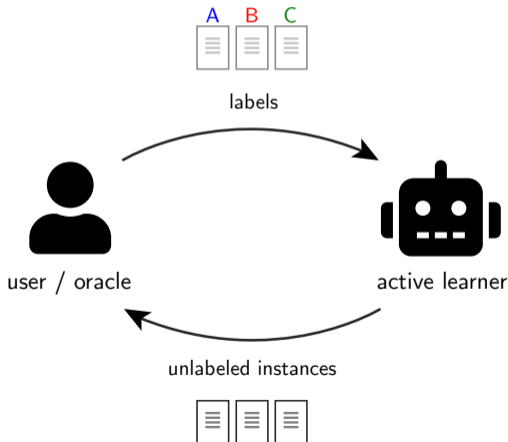
TEXT CLASSIFICATION

Text Classification: For each x_i in a dataset X predict class labels y_i . The number of classes is given by c .

- **binary:** predict $y_i \in \{0, 1\}$
- **multi-class:** predict $y_i \in \{0, 1, \dots, c - 1\}$
- **multi-label:** predict $y_i \subseteq \{0, 1, \dots, c - 1\}$

ACTIVE LEARNING

Active Learning: minimize the labeling costs of training data acquisition while maximizing a model's performance (increase) with each newly labeled problem instance.



ACTIVE LEARNING

Query Strategy: Decides which instances will be labeled next.

Selected Strategies

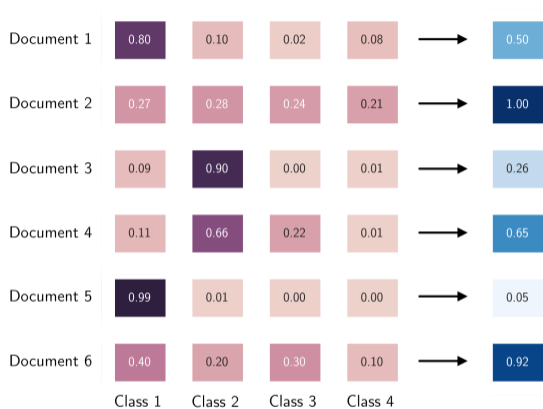
Prediction entropy (Roy and McCallum, 2001; Schohn and Cohn, 2000):

$$\operatorname{argmax}_{x_i} \left[- \sum_{j=1}^c P(y_i = j | x_i) \log P(y_i = j | x_i) \right] \quad (1)$$

Breaking ties / Minimum margin (Scheffer et al., 2001; Luo et al., 2005):

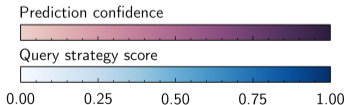
$$\operatorname{argmin}_{x_i} \left[P(y_i = k_1^* | x_i) - P(y_i = k_2^* | x_i) \right] \quad (2)$$

ACTIVE LEARNING – UNCERTAINTY EXAMPLE

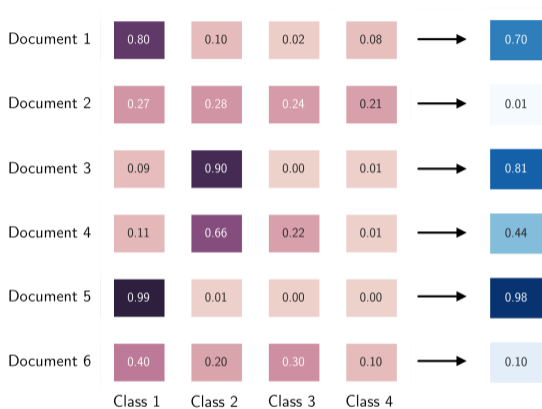


Prediction entropy

$$\operatorname{argmax}_{x_i} \left[-\sum_{j=1}^C P(y_i = j | x_i) \log P(y_i = j | x_i) \right]$$

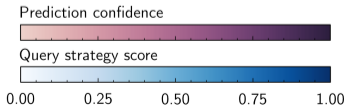


ACTIVE LEARNING – UNCERTAINTY EXAMPLE

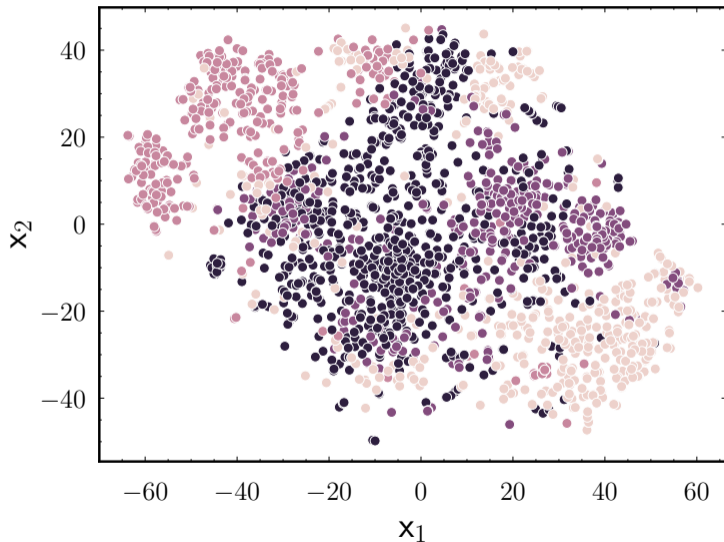


Breaking ties/Minimum margin

$$\operatorname{argmin}_{x_i} \left[P(y_i = k_1^* | x_i) - P(y_i = k_2^* | x_i) \right]$$



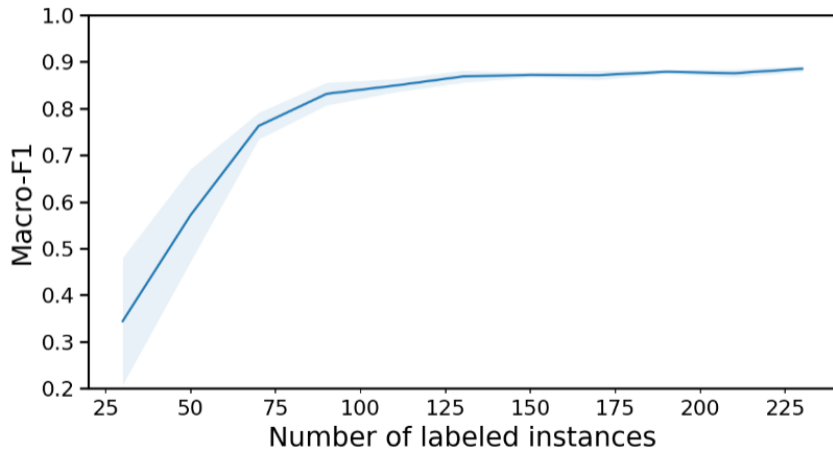
ACTIVE LEARNING – VECTOR SPACE



ACTIVE LEARNING – EVALUATION

Dataset: TREC-6

Iteration: 10



ACTIVE LEARNING IN IN SOCIAL SCIENCES

SO WHAT CAN ACTIVE LEARNING HELP YOU WITH?

Example: Text Classification

- Create a text classification model that generalizes to unseen data (*inductive*).
- Create model that annotates the rest of the corpus (*transductive*).
- Construct labeled corpora.
- Find inconsistencies in labeled corpora.

(Despite the absence of labeled data.)

ACTIVE LEARNING APPLICATIONS FROM THE SOCIAL SCIENCES

Selected Applications

- **Qualitative Content Analysis:** support annotation efforts (Liew et al., 2014; Chen et al., 2018; Wiedemann, 2019) and identify potentially mislabeled instances (Chen et al., 2018).
- **Categorization:** categorize or filter documents (Romberg and Escher, 2022).
- **Sentiment Analysis:** measure and evaluate subjectivity or sentiment (Liu, 2012; DiMaggio, 2015); possibly over time (Kahmann and Heyer, 2019).
- **Literature Research:** identify relevant literature or passages therein (Yu et al., 2018).

STATE OF THE ART AND RECENT RESEARCH

TRANSFORMER ERA AND BEYOND

- Considerable improvements on many text classification tasks have been achieved
- Trends in recent research (excluding LLMs which will be addressed later)
 - **query strategies** (Margatina et al., 2021; Zhang and Plank, 2021; Gonsior et al., 2022; Schröder et al., 2022)
 - **efficiency** (Tsvigun et al., 2022a; Jukić and Snajder, 2023; Yu et al., 2022)
 - **semi-supervised learning** (Tsvigun et al., 2022a; Yu et al., 2022)
 - **tooling** ALToolbox (Tsvigun et al., 2022b), ALANNO (Jukić et al., 2023), small-text (Schröder et al., 2023), ALAMBIC (Nachtegael et al., 2023)

ACTIVE LEARNING: UNCERTAINTY IS A STRONG BASELINE

Revisiting Uncertainty-based Query Strategies for Active Learning with Transformers

(Christopher Schröder, Andreas Niekler, Martin Potthast)

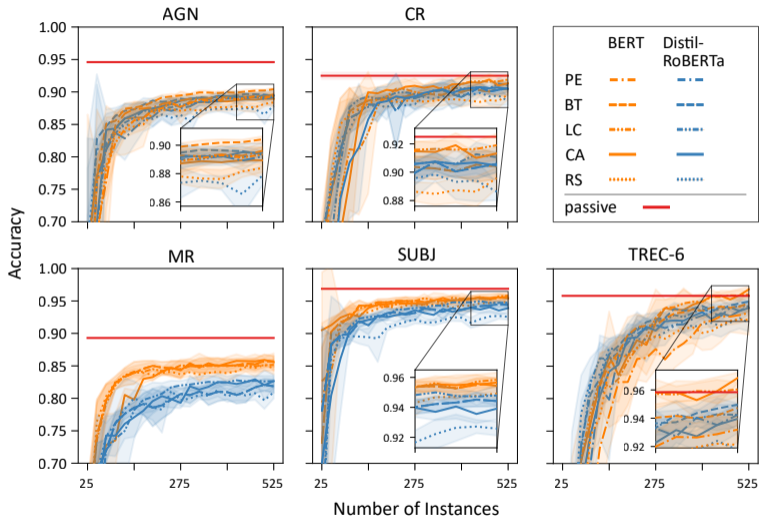
Experiment: Five query strategies were evaluated on BERT, DistilRoBERTa and two baselines (KimCNN and SVM).

Findings: Uncertainty-based query strategies with transformers are strong on text classification benchmarks.

Table: The “Mean Rank” columns show the mean rank when ordered by mean accuracy (Acc.) after the final iteration and by overall AUC. The “Mean Result” columns show the mean accuracy and AUC. Adopted and adapted from (Schröder et al., 2022).

Model	Strategy	Mean Rank		Mean Result	
		Acc.	AUC	Acc.	AUC
SVM	PE	1.80	2.60	0.764	0.663
	BT	1.60	1.60	0.767	0.697
	LC	3.00	2.60	0.751	0.672
	CA	5.00	5.00	0.667	0.593
	RS	3.00	2.60	0.757	0.686
KimCNN	PE	1.60	2.40	0.818	0.742
	BT	1.60	2.00	0.818	0.750
	LC	3.80	2.80	0.810	0.732
	CA	3.80	4.80	0.793	0.711
	RS	3.60	2.40	0.804	0.749
BERT	PE	2.40	2.40	0.909	0.859
	BT	2.00	1.60	0.914	0.873
	LC	2.20	3.80	0.917	0.866
	CA	2.80	2.60	0.916	0.872
	RS	5.00	4.00	0.899	0.861

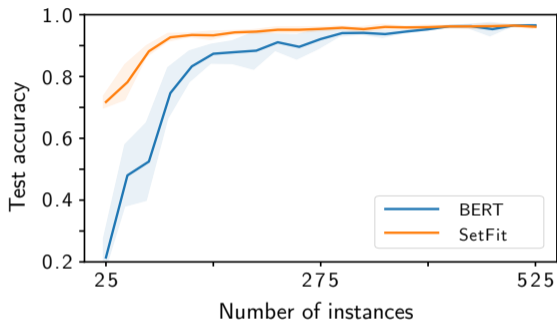
ACTIVE LEARNING: UNCERTAINTY IS A STRONG BASELINE



AGN: AG News (News), CR: Customer Reviews (Sentiment), MR: Movie Reviews (Sentiment),
SUBJ: Subjectivity (Sentiment), TREC-6: TREC-6 (Questions)

ACTIVE LEARNING: CONTRASTIVE REPRESENTATION LEARNING

- SetFit (Tunstall et al., 2022): with contrastive representation fine-tuning, transformer models can be even more sample efficient
- Steeper learning curves with a similar final classification performance (Schröder et al., 2023) (in a comparison that favored vanilla BERT)



An exemplary learning curve showing the difference in test accuracy for breaking ties on the TREC dataset, comparing BERT and SetFit. The tubes represent the standard deviation across five runs. Adopted from (Schröder et al., 2023).

ACTIVE LEARNING AND LARGE LANGUAGE MODELS

- LLMs can do remarkably well without any additional labels:
 - FreeAL (Xiao et al., 2023b) and LLMAAA (Zhang et al., 2023a) use the LLM as an annotator to train smaller LMs
 - In-context learning with samples that are acquired by active learning (Margatina et al., 2023; Mavromatis et al., 2023)
- Zero-shot has been reported to be very effective on simple sentiment classification tasks, so that active learning is not even required (Zhang et al., 2023b)

ACTIVE LEARNING AND LARGE LANGUAGE MODELS

- Problems with LLMs:
 - LLMs require adequate hardware (costs ↑)
 - More parameters increase training and inference time (turnaround times ↑)
 - Current evaluations might be misleading due to *dataset contamination* (Golchin and Surdeanu, 2023; Sainz et al., 2023)
 - LLMs still have problems with some tasks (e.g., finer-grained sentiment) (Zhang et al., 2023b)
 - Also, recent research claims that human labeling is still superior (Lu et al., 2023)

→ *My prediction*: Smaller domain-specific models will likely be preferred.

→ Active learning might utilize LLMs for model distillation or for (semi-)automated labeling.

PART 2: SMALL-TEXT – A LIBRARY FOR
ACTIVE LEARNING FOR TEXT CLASSIFICATION

INTRODUCING SMALL-TEXT



The screenshot shows the homepage for the 'small-text' Python library. At the top, there are several status badges: 'pypi v1.3.2', 'conda-forge v1.3.2', 'codecov 96%', 'docs passing', 'maintained yes', 'contributions welcome', and 'license MIT'. Below these is a DOI badge '10.5281/zenodo.8256087' and a 'Tweet' button. The main heading is 'SMALL-TEXT' in a large, blue, sans-serif font. Underneath, it says 'Active Learning for Text Classification in Python.' A horizontal line separates this from a list of links: 'Installation | Quick Start | Contribution | Changelog | Docs'. The main text describes the library as providing state-of-the-art 'Active Learning' for text classification, with pre-implemented query strategies, initialization strategies, and stopping criteria. A section titled 'What is Active Learning?' includes a link to 'Active Learning' which explains its use for labeling training data in small data scenarios.

small-text (Schröder et al., 2023) is an open source Python library for Active Learning for Text Classification



 Code

github.com/webis-de/small-text

INTRODUCING SMALL-TEXT



The screenshot shows the GitHub repository page for 'small-text'. At the top, there are several status badges: 'pypi v1.3.2', 'conda-forge v1.3.2', 'codecov 96%', 'docs passing', 'maintained yes', 'contributions welcome', and 'license MIT'. Below these is a DOI badge: 'DOI 10.5281/zenodo.8256087' and a 'Tweet' button. The main heading is 'SMALL-TEXT' in a large, blue, sans-serif font. Underneath, it says 'Active Learning for Text Classification in Python.' followed by a horizontal line. Below the line are links for 'Installation', 'Quick Start', 'Contribution', 'Changelog', and 'Docs'. A paragraph of text describes the library: 'Small-Text provides state-of-the-art **Active Learning** for Text Classification. Several pre-implemented Query Strategies, Initialization Strategies, and Stopping Criteria are provided, which can be easily mixed and matched to build active learning experiments or applications.' Below this is a section titled 'What is Active Learning?' with a link to 'Active Learning' that says 'allows you to efficiently label training data in a small data scenario.'

small-text (Schröder et al., 2023) is an open source Python library for Active Learning for Text Classification

Motivation:

- A typical active learning experiment can quickly get very complex



 Code

github.com/webis-de/small-text

INTRODUCING SMALL-TEXT



The screenshot shows the GitHub repository page for 'small-text'. At the top, there are several status badges: 'pypi v1.3.2', 'conda-forge v1.3.2', 'codecov 96%', 'docs passing', 'maintained yes', 'contributions welcome', and 'license MIT'. Below these is the DOI '10.5281/zenodo.8256087' and a 'Tweet' button. The main heading is 'SMALL-TEXT' in large blue letters. Underneath, it says 'Active Learning for Text Classification in Python.' There are links for 'Installation', 'Quick Start', 'Contribution', 'Changelog', and 'Docs'. A paragraph describes the library: 'Small-Text provides state-of-the-art **Active Learning** for Text Classification. Several pre-implemented Query Strategies, Initialization Strategies, and Stopping Criteria are provided, which can be easily mixed and matched to build active learning experiments or applications.' Below that, a section titled 'What is Active Learning?' includes a link to 'Active Learning' which states it allows for efficient labeling of training data in small data scenarios.

small-text (Schröder et al., 2023) is an open source Python library for Active Learning for Text Classification

Motivation:

- A typical active learning experiment can quickly get very complex
- AL is inherently very modular, but combining different components is often not



 Code

github.com/webis-de/small-text

INTRODUCING SMALL-TEXT



The screenshot shows the GitHub repository page for 'small-text'. At the top, there are badges for 'python v1.3.2', 'conda-forge v1.3.2', 'codecov 96%', 'docs passing', 'maintained yes', 'contributions welcome', and 'license MIT'. Below these is the DOI '10.5281/zenodo.8266087' and a 'Tweet' button. The main heading is 'SMALL-TEXT' in a large, blue, sans-serif font. Underneath, it says 'Active Learning for Text Classification in Python.' followed by a horizontal line. Below the line are links for 'Installation', 'Quick Start', 'Contribution', 'Changelog', and 'Docs'. A paragraph describes the library: 'Small-Text provides state-of-the-art **Active Learning** for Text Classification. Several pre-implemented Query Strategies, Initialization Strategies, and Stopping Criteria are provided, which can be easily mixed and matched to build active learning experiments or applications.' Below this is a section titled 'What is Active Learning?' with a link to 'Active Learning' that says 'allows you to efficiently label training data in a small data scenario.'



 Code

github.com/webis-de/small-text


small-text (Schröder et al., 2023) is an open source Python library for Active Learning for Text Classification

Motivation:

- A typical active learning experiment can quickly get very complex
- AL is inherently very modular, but combining different components is often not
- Reproducibility and correctness

OVERVIEW: SMALL-TEXT IN 2023

- Github: 502 stars / 550 commits / 7 releases (as of December 5th)
- Paper: Published at EACL23 (Best System Demonstration Award)



The screenshot shows the top section of the SMALL-TEXT project page. At the top, there is a row of status badges: 'pypi v1.3.2', 'conda-forge v1.3.2', 'codecov 96%', 'docs passing', 'maintained yes', 'contributions welcome', and 'license MIT'. Below these is a DOI badge '10.5281/zenodo.8266087' and a 'Tweet' button. The main heading 'SMALL-TEXT' is displayed in a large, blue, sans-serif font. Underneath the heading is a vertical bar followed by the text 'Active Learning for Text Classification in Python.'. A horizontal line separates this from a row of navigation links: 'Installation | Quick Start | Contribution | Changelog | Docs'. Below the links is a paragraph of text: 'Small-Text provides state-of-the-art **Active Learning** for Text Classification. Several pre-implemented Query Strategies, Initialization Strategies, and Stopping Criteria are provided, which can be easily mixed and matched to build active learning experiments or applications.' At the bottom left, the text 'What is Active Learning?' is visible.

pypi v1.3.2 conda-forge v1.3.2 codecov 96% docs passing maintained yes contributions welcome license MIT

DOI 10.5281/zenodo.8266087 X Tweet

SMALL-TEXT

| Active Learning for Text Classification in Python.

[Installation](#) | [Quick Start](#) | [Contribution](#) | [Changelog](#) | [Docs](#)

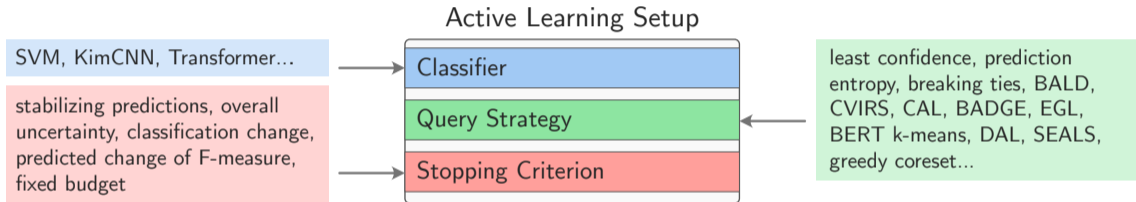
Small-Text provides state-of-the-art **Active Learning** for Text Classification. Several pre-implemented Query Strategies, Initialization Strategies, and Stopping Criteria are provided, which can be easily mixed and matched to build active learning experiments or applications.

What is Active Learning?

SCOPE

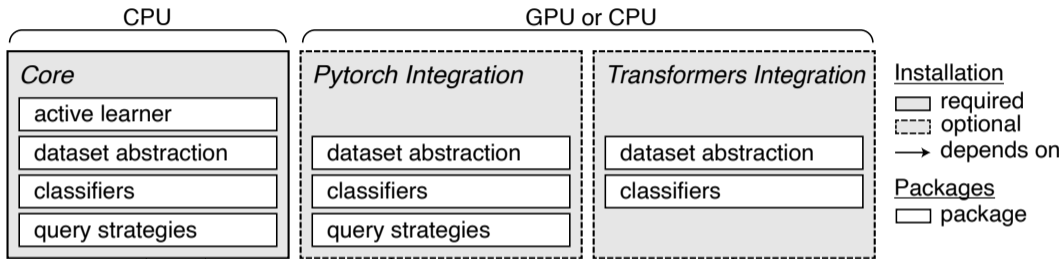
- **Goal: provide state-of-the-art active learning for text classification**
- **Target groups: researchers and practitioners**
- Active Learning: 16 ~~14~~ query strategies, 5 stopping criteria
- Integrates scikit-learn, PyTorch, and transformers
- Available via pip and conda, documentation, examples, unit and integration testing
- Replicability & reproducibility (if applied correctly)

UNIFIED INTERFACES












- Goal: Easily mix and match different components

PACKAGE ARCHITECTURE



- Modular architecture, open for extension
- CPU or GPU installation possible
- Except for the dataset abstractions, most parts of the code operate agnostic of dataset and classifier

RELATED SOFTWARE

Name	Active Learning			Code					
	QS	SC	Text Focus	GPU support	Unit Tests	Language	License	Last Update	Repository
JCLAL ¹	18	2	✗	✗	✗	Java	GPL	2017	
libact ²	19	-	✗	✗	✓	Python	BSD-2-Clause	2021	
modAL ³	21	-	✗	✓	✓	Python	MIT	2022	
ALiPy ⁴	22	4	✗	✗	✓	Python	BSD-3-Clause	2022	
Baal ⁵	9	-	✗	✓	✓	Python	Apache 2.0	2023	
lrtc ⁶	7	-	✓	✓	✗	Python	Apache 2.0	2021	
scikit-activeml ⁷	29	-	✗	✓	✓	Python	BSD-3-Clause	2023	
ALToolbox ⁸	19	-	✓	✓	✓	Python	MIT	2023	
small-text	16-14	5	✓	✓	✓	Python	MIT	2023	

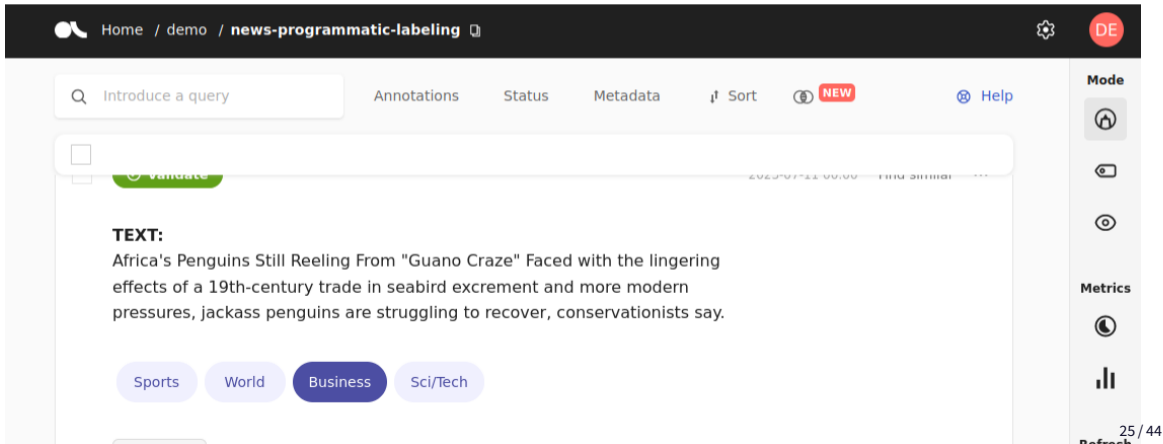
Comparison between small-text and relevant previous active learning libraries. We abbreviated the number of query strategies by “QS”, number of stopping criteria by “SC”, and the low-resource-text-classification framework by lrtc.

Publications: ¹Reyes et al., 2018, ²Yang et al., 2017, ³Danka and Horvath, 2018, ⁴Tang et al., 2019, ⁵Atighehchian et al., 2020,

⁶Ein-Dor et al., 2020, ⁷Kottke et al., 2021, ⁸Tsvigun et al., 2022a.

LIBRARY VERSUS ANNOTATION TOOL

Of course, you can also “simply use” `smalldata-text` for annotation. For example, argilla, a platform for data-centric NLP and LLMs, provides a tutorial for integrating `smalldata-text`:



The screenshot displays the Argilla web interface. At the top, a navigation bar shows the breadcrumb "Home / demo / news-programmatic-labeling" and a user profile icon labeled "DE". Below this is a search bar with the placeholder "Introduce a query" and a list of tabs: "Annotations", "Status", "Metadata", "Sort", and "Help". A "NEW" badge is visible next to the "Sort" tab. The main content area shows a news article snippet with the text: "Africa's Penguins Still Reeling From 'Guano Craze' Faced with the lingering effects of a 19th-century trade in seabird excrement and more modern pressures, jackass penguins are struggling to recover, conservationists say." Below the text are four category buttons: "Sports", "World", "Business" (which is selected and highlighted in dark blue), and "Sci/Tech". On the right side, there is a vertical sidebar with icons for "Mode", "Metrics", and a bar chart icon. At the bottom right corner, the text "25 / 44" and a "Refresh" button are visible.

CODE EXAMPLE: WORDS OF THE DAY CORPUS

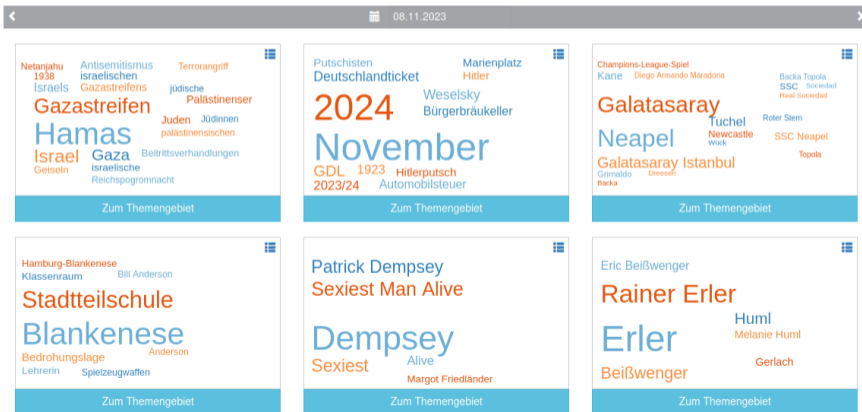
WORDS OF THE DAY



WÖRTER DES TAGES
UNIVERSITÄT LEIPZIG

2023
08.11.

Die »Wörter des Tages« zeigen, welche Begriffe heute besonders aktuell sind. Dazu werden verschiedene Tageszeitungen und Newsdienste täglich ausgewertet. Die »Wörter des Tages« stehen morgens ab etwa 7 Uhr zur Verfügung. Die Aktualität eines Begriffs ergibt sich aus seiner Häufigkeit heute, verglichen mit seiner durchschnittlichen Häufigkeit über längere Zeit hinweg.



ACTIVE LEARNING APPLICATION: WORDS OF THE DAY

How to use `small-text` to perform a (simulated) `active learning experiment`?

Data

- Words of the Day corpus (“eng_news_2023”)
- Labels similar to AG News¹: SPORTS, WORLD, BUSINESS, and SCITECH
- Gold labels: use article URL as a heuristic
e.g., `https://newspaper.com/sports/`
- Sample 200K sentences from articles with similar URL patterns

¹http://groups.di.unipi.it/~gulli/AG_corpus_of_news_articles.html

ACTIVE LEARNING APPLICATION: DATA EXPLORATION

Examples

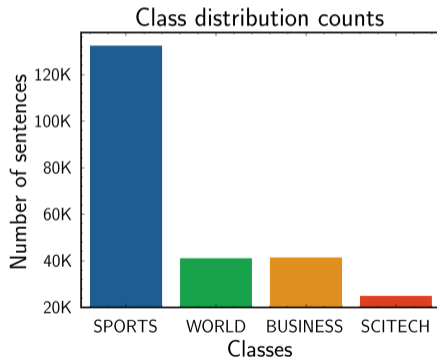
SPORTS Get the surgery and return midseason, or play through pain?

WORLD Firefighters extinguish a car burned during night clashes in the Alma district of Roubaix, in northern France, on Friday.

BUSINESS It offloaded its environmental consulting division last year in a move to reduce its debt by more than one billion US dollars (£805 million).

SCITECH For example, one of its two main instruments, the craft's Vis imager, was mostly built in the UK.

Class distribution



SMALL-TEXT: SETTING UP DATASET AND TOKENIZER

```
import pandas as pd
from transformers import AutoTokenizer

num_classes = 4
transformer_model = 'bert-base-uncased'

tokenizer = AutoTokenizer.from_pretrained(transformer_model)

df = pd.read_parquet('eng_news_2023_200K.parquet')
train_set_size = 180_000
```


SMALL-TEXT: TRAIN AND TEST SPLIT (EXPERIMENT SCENARIO)

```
import numpy as np
from small_text import TransformersDataset, TransformerModelArguments

train = TransformersDataset.from_arrays(
    df['text'][:train_set_size], df['label'][:train_set_size].values, tokenizer,
    target_labels=np.arange(num_classes),
    max_length=128
)

test = TransformersDataset.from_arrays(
    df['text'][train_set_size:], df['label'][train_set_size:].values, tokenizer,
    target_labels=np.arange(num_classes),
    max_length=128
)
```

SMALL-TEXT: SETTING UP CLASSIFIER AND QUERY STRATEGY

```
from small_text import BreakingTies, TransformerBasedClassificationFactory
```

```
model_args = TransformerModelArguments(transformer_model)
```

```
clf_kwargs = {'device': 'cuda', 'mini_batch_size': 64}
```

```
clf_factory = TransformerBasedClassificationFactory(  
    model_args,  
    num_classes,  
    kwargs=clf_kwargs)
```

```
query_strategy = BreakingTies()
```

SMALL-TEXT: POOL-BASED ACTIVE LEARNING

```
from small_text import PoolBasedActiveLearner, random_initialization_balanced

active_learner = PoolBasedActiveLearner(clf_factory, query_strategy, train)

# Randomly select initial samples.
indices_initial = random_initialization_balanced(train.y, n_samples=10)

active_learner.initialize_data(
    indices_initial,
    train.y[indices_initial]
)
```

SMALL-TEXT: POOL-BASED ACTIVE LEARNING

```
from sklearn.metrics import accuracy_score, f1_score

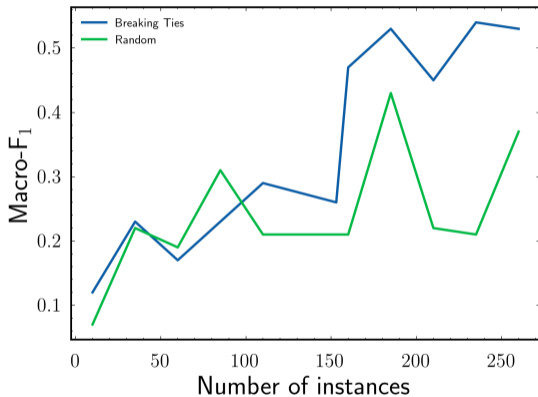
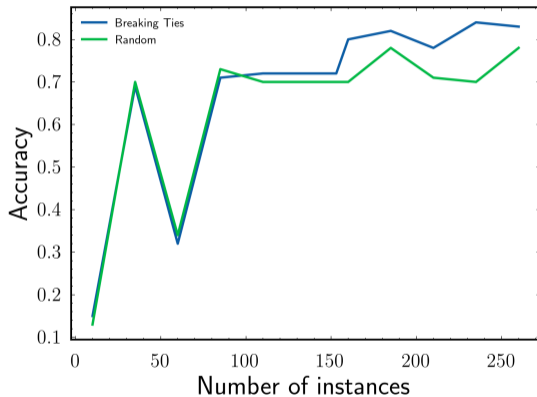
num_queries = 10
for i in range(num_queries):
    # Query 25 samples per iteration.
    indices_queried = active_learner.query(num_samples=25)

    # Simulate user interaction here. Replace this for real-world usage.
    y = train.y[indices_queried]
    active_learner.update(y)

y_pred_test = active_learner.classifier.predict(test)

accuracy = accuracy_score(y_pred_test, test.y)
macro_f1 = f1_score(y_pred_test, test.y, average='macro')
print(f'Iteration {i + 1}: Test accuracy: {accuracy:.2f} / {macro_f1:.2f}')
```

RESULT: LEARNING CURVES



→ A news classifier using **260 labeled instances** built with **less than 100 lines of code**.

SUMMARY AND FUTURE WORK

SUMMARY AND FUTURE WORK

`small-text` provides easy-to-use modular active learning for text classification in Python.

Interested? Try it yourself.

- Use `small-text` to build new datasets or models
- Use `small-text` for your research
- Contribute to the Github repository

FUTURE PLANS

Current version: 1.3.2 (August 19th)

v2.0.0+

- More "active learning" (query strategies)
- Additional classification functionality
- Convenience and usability
- Documentation and examples

v3.0.0+

- Active learning for token classification

PART 3: PRACTICAL CHALLENGES WHEN USING ACTIVE LEARNING WITH LLMs

LANGUAGE MODEL INSTABILITY

Model Instability: When trained on only few instances of data, model performance exhibits a large variance (Mosbach et al., 2021).

In the context of Active Learning:

- Especially prone to this due to the inherent low-data scenario.
- Workaround: Reinitialize the model before every iteration.

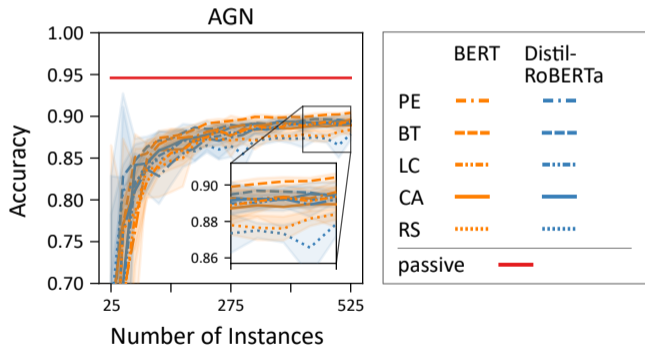


Figure adopted and adapted from (Schröder et al., 2022).

LIMITED CONTEXT LENGTH

Context Length: Transformer models operate on tokens.

- Attention mechanism: number of maximum input tokens is fixed.
(e.g., 512 for a BERT model (Devlin et al., 2019))
- This maximum token size is responsible for quadratic scaling and memory in the original attention mechanism (Vaswani et al., 2017).

Several solutions have been proposed to extend this window:

- Among others: CogLTX (Ding et al., 2020), Longformer (Beltagy et al., 2020), Attention sinks (Xiao et al., 2023a).

HARDWARE RESOURCES

GPUs are required for most recent methods.

Runtime

- Even for “smaller” BERT-era models, experiments quickly get infeasible.
- Aggravated for active learning experiments: multiple configurations, multiple repetitions per configuration.
- Turnaround time is a factor for active learning.

VRAM

- Often the limiting factor for the choice of model.

RUNTIME OPTIMIZATIONS

Reduce training and/or inference time

- Smaller models (possibly distilled).
- Code compilation (GPU optimizations).
- Minimize data transfer between CPU and GPU.

Case Example (by Sebastian Raschka)

- Reduction in training time from 21.33 min to 8.25 min by switching to mixed precision training reduced training time for a particular configuration.

Source: <https://sebastianraschka.com/blog/2023/pytorch-faster.html>

MEMORY OPTIMIZATIONS

Reduce required VRAM

- Parameter-efficient fine-tuning
- Mixed or lower precision training.
- Use different optimization algorithm (e.g., instead of Adam).

SUMMARY

- Practical obstacles shape experiments and applications (e.g., the choice of model).
- Knowledge about hardware and lower-level optimizations is beneficial.
- Multitude of different techniques exists that reduce required resources.
- Efficiency
 - Larger models increase runtime. This is often undesirable.
 - Take power usage into account: Green AI (Schwartz et al., 2020).
 - Appropriateness: Is a 1T model needed for my dataset of 10k instances?

OUTRO: TAKEAWAYS

TAKEAWAYS

Part 1

- What is active learning? What are typical use cases?
- Recognize problems where active learning can be applied





Part 2

- Motivation and features of the small-text library
- Follow a minimal code example for an active learning experiment



Part 3

- Learn about practical challenges when using LLMs

BIBLIOGRAPHY I

-  Atighehchian, Parmida, Frédéric Branchaud-Charron, and Alexandre Lacoste (2020). “Bayesian active learning for production, a systematic study and a reusable library”. In: *arXiv preprint arXiv:2006.09916*.
-  Beltagy, Iz, Matthew E. Peters, and Arman Cohan (2020). “Longformer: The Long-Document Transformer”. In: *arXiv preprint arXiv:2004.05150*.
-  Chen, Nan-Chen, Margaret Drouhard, Rafal Kocielnik, Jina Suh, and Cecilia R. Aragon (2018). “Using Machine Learning to Support Qualitative Coding in Social Science: Shifting the Focus to Ambiguity”. In: *ACM Trans. Interact. Intell. Syst.* 8.2. ISSN: 2160-6455.
-  Danka, Tivadar and Peter Horvath (2018). “modAL: A modular active learning framework for Python”. In: *arXiv preprint arXiv:1805.00979*.



BIBLIOGRAPHY II

-  Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova (2019). “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding”. In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*. Association for Computational Linguistics, pp. 4171–4186.
-  DiMaggio, Paul (2015). “Adapting computational text analysis to social science (and vice versa)”. In: *Big Data & Society* 2.2, p. 2053951715602908. eprint: <https://doi.org/10.1177/2053951715602908>.



BIBLIOGRAPHY III

-  Ding, Ming, Chang Zhou, Hongxia Yang, and Jie Tang (2020). “CogLTX: Applying BERT to Long Texts”. In: *Advances in Neural Information Processing Systems*. Ed. by H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin. Vol. 33. Curran Associates, Inc., pp. 12792–12804.
-  Ein-Dor, Liat, Alon Halfon, Ariel Gera, Eyal Shnarch, Lena Dankin, Leshem Choshen, Marina Danilevsky, Ranit Aharonov, Yoav Katz, and Noam Slonim (2020). “Active Learning for BERT: An Empirical Study”. In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Online: Association for Computational Linguistics, pp. 7949–7962.
-  Golchin, Shahriar and Mihai Surdeanu (2023). “Time Travel in LLMs: Tracing Data Contamination in Large Language Models”. In: *arXiv preprint arXiv:2308.08493*.




BIBLIOGRAPHY IV

-  Gonsior, Julius, Christian Falkenberg, Silvio Magino, Anja Reusch, Maik Thiele, and Wolfgang Lehner (2022). “To Softmax, or not to Softmax: that is the question when applying Active Learning for Transformer Models”. In: *arXiv preprint arXiv:2210.03005*.
-  Jukić, Josip, Fran Jelenić, Miroslav Bićanić, and Jan Snajder (2023). “ALANNO: An Active Learning Annotation System for Mortals”. In: *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*. Ed. by Danilo Croce and Luca Soldaini. Dubrovnik, Croatia: Association for Computational Linguistics, pp. 228–235.




BIBLIOGRAPHY V

-  Jukić, Josip and Jan Snajder (2023). “Parameter-Efficient Language Model Tuning with Active Learning in Low-Resource Settings”. In: *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*. Ed. by Houda Bouamor, Juan Pino, and Kalika Bali. Singapore: Association for Computational Linguistics, pp. 5061–5074.
-  Kahmann, Christian and Gerhard Heyer (2019). “Measuring Context Change to Detect Statements Violating the Overton Window”. In: *Proceedings of the 11th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management, IC3K 2019, Volume 1: KDIR, Vienna, Austria, September 17-19, 2019*. Ed. by Ana L. N. Fred and Joaquim Filipe. ScitePress, pp. 392–396.




BIBLIOGRAPHY VI

-  Kottke, Daniel, Marek Herde, Tuan Pham Minh, Alexander Benz, Pascal Mergard, Atal Roghman, Christoph Sandrock, and Bernhard Sick (2021). “scikitactiveml: A Library and Toolbox for Active Learning Algorithms”. In: *Preprints.org*.
-  Liew, Jasy Suet Yan, Nancy McCracken, Shichun Zhou, and Kevin Crowston (2014). “Optimizing Features in Active Machine Learning for Complex Qualitative Content Analysis”. In: *Proceedings of the ACL 2014 Workshop on Language Technologies and Computational Social Science*. Ed. by Cristian Danescu-Niculescu-Mizil, Jacob Eisenstein, Kathleen McKeown, and Noah A. Smith. Baltimore, MD, USA: Association for Computational Linguistics, pp. 44–48.
-  Liu, Bing (2012). *Sentiment Analysis and Opinion Mining*. Morgan & Claypool Publishers. ISBN: 1608458849.



BIBLIOGRAPHY VII

-  Lu, Yuxuan, Bingsheng Yao, Shao Zhang, Yun Wang, Peng Zhang, Tun Lu, Toby Jia-Jun Li, and Dakuo Wang (2023). “Human Still Wins over LLM: An Empirical Study of Active Learning on Domain-Specific Annotation Tasks”. In: *arXiv preprint arXiv:2311.09825*.
-  Luo, Tong, Kurt Kramer, Dmitry B. Goldgof, Lawrence O. Hall, Scott Samson, Andrew Remsen, and Thomas Hopkins (2005). “Active Learning to Recognize Multiple Types of Plankton”. In: *Journal of Machine Learning Research (JMLR)* 6, pp. 589–613.
-  Margatina, Katerina, Timo Schick, Nikolaos Aletras, and Jane Dwivedi-Yu (2023). “Active Learning Principles for In-Context Learning with Large Language Models”. In: *arXiv preprint arXiv:2305.14264*.



BIBLIOGRAPHY VIII

-  Margatina, Katerina, Giorgos Vernikos, Loïc Barrault, and Nikolaos Aletras (2021). “Active Learning by Acquiring Contrastive Examples”. In: *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Online and Punta Cana, Dominican Republic: Association for Computational Linguistics, pp. 650–663.
-  Mavromatis, Costas, Balasubramaniam Srinivasan, Zhengyuan Shen, Jiani Zhang, Huzefa Rangwala, Christos Faloutsos, and George Karypis (2023). “Which Examples to Annotate for In-Context Learning? Towards Effective and Efficient Selection”. In: *arXiv preprint arXiv:2310.20046*.
-  Mosbach, Marius, Maksym Andriushchenko, and Dietrich Klakow (2021). “On the Stability of Fine-tuning BERT: Misconceptions, Explanations, and Strong Baselines”. In: *Proceedings of the 9th International Conference on Learning Representations (ICLR 2021)*. OpenReview.net.




BIBLIOGRAPHY IX

-  Nachtegaele, Charlotte, Jacopo De Stefani, and Tom Lenaerts (2023). “ALAMBIC : Active Learning Automation Methods to Battle Inefficient Curation”. In: *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*. Ed. by Danilo Croce and Luca Soldaini. Dubrovnik, Croatia: Association for Computational Linguistics, pp. 117–127.
-  Reyes, Oscar, Carlos Morell, and Sebastián Ventura (2018). “Effective active learning strategy for multi-label learning”. In: *Neurocomputing* 273, pp. 494–508.




BIBLIOGRAPHY X

-  Romberg, Julia and Tobias Escher (2022). “Automated Topic Categorisation of Citizens’ Contributions: Reducing Manual Labelling Efforts Through Active Learning”. In: *Electronic Government*. Ed. by Marijn Janssen, Csaba Csáki, Ida Lindgren, Euripidis Loukis, Ulf Melin, Gabriela Viale Pereira, Manuel Pedro Rodríguez Bolívar, and Efthimios Tambouris. Cham: Springer International Publishing, pp. 369–385. ISBN: 978-3-031-15086-9.
-  Roy, Nicholas and Andrew McCallum (2001). “Toward Optimal Active Learning through Sampling Estimation of Error Reduction”. In: *Proceedings of the Eighteenth International Conference on Machine Learning (ICML)*. ICML’01. Morgan Kaufmann Publishers Inc., pp. 441–448.

BIBLIOGRAPHY XI

-  Sainz, Oscar, Jon Ander Campos, Iker García-Ferrero, Julen Etxaniz, Oier Lopez de Lacalle, and Eneko Agirre (2023). “NLP Evaluation in trouble: On the Need to Measure LLM Data Contamination for each Benchmark”. In: *arXiv preprint arXiv:2310.18018*.
-  Scheffer, Tobias, Christian Decomain, and Stefan Wrobel (2001). “Active Hidden Markov Models for Information Extraction”. In: *Proceedings of the 4th International Conference on Advances in Intelligent Data Analysis (IDA)*. IDA '01. Berlin, Heidelberg: Springer-Verlag, pp. 309–318. ISBN: 3540425810.
-  Schohn, Greg and David Cohn (2000). “Less is More: Active Learning with Support Vector Machines”. In: *Proceedings of the Seventeenth International Conference on Machine Learning (ICML)*. Ed. by Pat Langley. Morgan Kaufmann, pp. 839–846.


BIBLIOGRAPHY XII

-  Schröder, Christopher, Lydia Müller, Andreas Niekler, and Martin Potthast (2023). “Small-Text: Active Learning for Text Classification in Python”. In: *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*. Ed. by Danilo Croce and Luca Soldaini. Dubrovnik, Croatia: Association for Computational Linguistics, pp. 84–95.
-  Schröder, Christopher, Andreas Niekler, and Martin Potthast (2022). “Revisiting Uncertainty-based Query Strategies for Active Learning with Transformers”. In: *Findings of the Association for Computational Linguistics: ACL 2022 (Findings of ACL 2022)*, pp. 2194–2203.
-  Schwartz, Roy, Jesse Dodge, Noah A. Smith, and Oren Etzioni (2020). “Green AI”. In: *Commun. ACM* 63.12, pp. 54–63. ISSN: 0001-0782.





BIBLIOGRAPHY XIII

-  Tang, Ying-Peng, Guo-Xiang Li, and Sheng-Jun Huang (2019). “ALiPy: Active Learning in Python”. In: *arXiv preprint arXiv:1901.03802*.
-  Tsvigun, Akim, Artem Shelmanov, Gleb Kuzmin, Leonid Sanochkin, Daniil Larionov, Gleb Gusev, Manvel Avetisian, and Leonid Zhukov (2022a). “Towards Computationally Feasible Deep Active Learning”. In: *Findings of the Association for Computational Linguistics: NAACL 2022*. Ed. by Marine Carpuat, Marie-Catherine de Marneffe, and Ivan Vladimir Meza Ruiz. Seattle, United States: Association for Computational Linguistics, pp. 1198–1218.

BIBLIOGRAPHY XIV

-  Tsvigun, Akim et al. (2022b). “ALToolbox: A Set of Tools for Active Learning Annotation of Natural Language Texts”. In: *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Ed. by Wanxiang Che and Ekaterina Shutova. Abu Dhabi, UAE: Association for Computational Linguistics, pp. 406–434.
-  Tunstall, Lewis, Nils Reimers, Unso Eun Seo Jo, Luke Bates, Daniel Korat, Moshe Wasserblat, and Oren Pereg (2022). “Efficient Few-Shot Learning Without Prompts”. In: *arXiv preprint arXiv:2209.11055*.
-  Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin (2017). “Attention is All you Need”. In: *Proceedings of the Advances in Neural Information Processing Systems 30 (NeurIPS)*, pp. 5998–6008.



BIBLIOGRAPHY XV

-  Wiedemann, Gregor (2019). “Proportional Classification Revisited: Automatic Content Analysis of Political Manifestos Using Active Learning”. In: *Social Science Computer Review* 37.2, pp. 135–159. eprint: <https://doi.org/10.1177/0894439318758389>.
-  Xiao, Guangxuan, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis (2023a). “Efficient Streaming Language Models with Attention Sinks”. In: *arXiv preprint arXiv:2309.17453*.
-  Xiao, Ruixuan, Yiwen Dong, Junbo Zhao, Runze Wu, Minmin Lin, Gang Chen, and Haobo Wang (2023b). “FreeAL: Towards Human-Free Active Learning in the Era of Large Language Models”. In: *arXiv preprint arXiv:2311.15614*.
-  Yang, Yao-Yuan, Shao-Chuan Lee, Yu-An Chung, Tung-En Wu, Si-An Chen, and Hsuan-Tien Lin (2017). “libact: Pool-based Active Learning in Python”. In: *arXiv preprint arXiv:1710.00379*.

BIBLIOGRAPHY XVI

-  Yu, Yue, Lingkai Kong, Jieyu Zhang, Rongzhi Zhang, and Chao Zhang (2022). “AcTune: Uncertainty-Based Active Self-Training for Active Fine-Tuning of Pretrained Language Models”. In: *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Ed. by Marine Carpuat, Marie-Catherine de Marneffe, and Ivan Vladimir Meza Ruiz. Seattle, United States: Association for Computational Linguistics, pp. 1422–1436.
-  Yu, Zhe, Nicholas A. Kraft, and Tim Menzies (2018). “Finding better active learners for faster literature reviews”. In: *Empir. Softw. Eng.* 23.6, pp. 3161–3186.
-  Zhang, Mike and Barbara Plank (2021). “Cartography Active Learning”. In: *Findings of the Association for Computational Linguistics (EMNLP Findings)*. Punta Cana, Dominican Republic: Association for Computational Linguistics, pp. 395–406.

BIBLIOGRAPHY XVII

-  Zhang, Ruoyu, Yanzeng Li, Yongliang Ma, Ming Zhou, and Lei Zou (2023a). “LLMaAA: Making Large Language Models as Active Annotators”. In: *arXiv preprint 2310.19596*.
-  Zhang, Wenxuan, Yue Deng, Bing Liu, Sinno Jialin Pan, and Lidong Bing (2023b). “Sentiment Analysis in the Era of Large Language Models: A Reality Check”. In: *arXiv preprint arXiv:2305.15005*.