Part III: Weakly-Supervised Natural Language Understanding: Text Classification and Beyond

KDD 2023 Tutorial

Pretrained Language Representations for Text Understanding: A Weakly-Supervised Perspective

Yu Meng, Jiaxin Huang, Yu Zhang, Yunyi Zhang, Jiawei Han

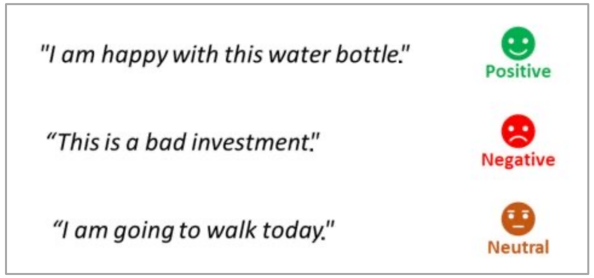
Computer Science, University of Illinois Urbana-Champaign

Aug 9, 2023

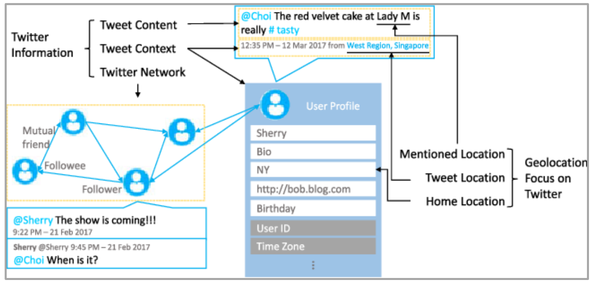
Tutorial Website:

Text Classification

- Given a set of text units (e.g., documents, sentences) and a set of categories, the task is to assign relevant category/categories to each text unit
- Text Classification has a lot of downstream applications

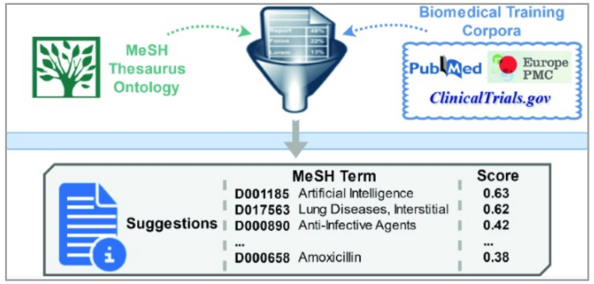


Sentiment Analysis

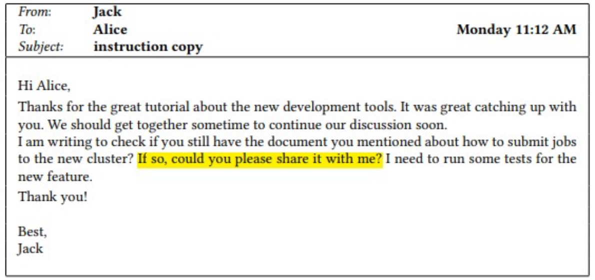


Location Prediction

News Topic Classification



Paper Topic Classification



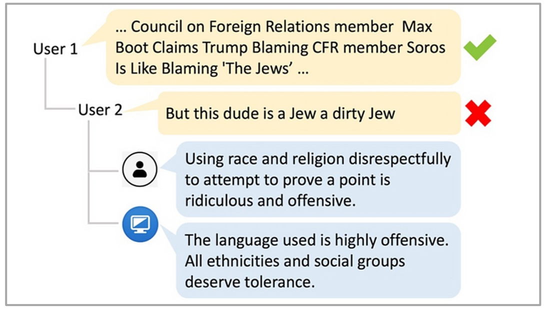
From: Jack
To: Alice
Subject: instruction copy
Monday 11:12 AM

Hi Alice,

Thanks for the great tutorial about the new development tools. It was great catching up with you. We should get together sometime to continue our discussion soon.
I am writing to check if you still have the document you mentioned about how to submit jobs to the new cluster? If so, could you please share it with me? I need to run some tests for the new feature.
Thank you!

Best,
Jack

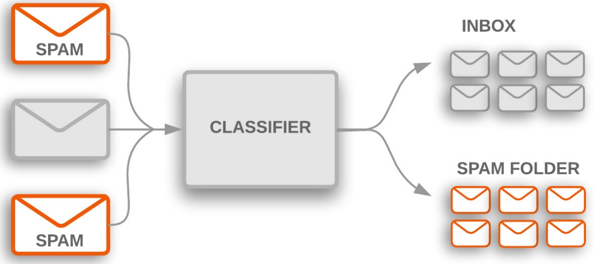
Email Intent Identification



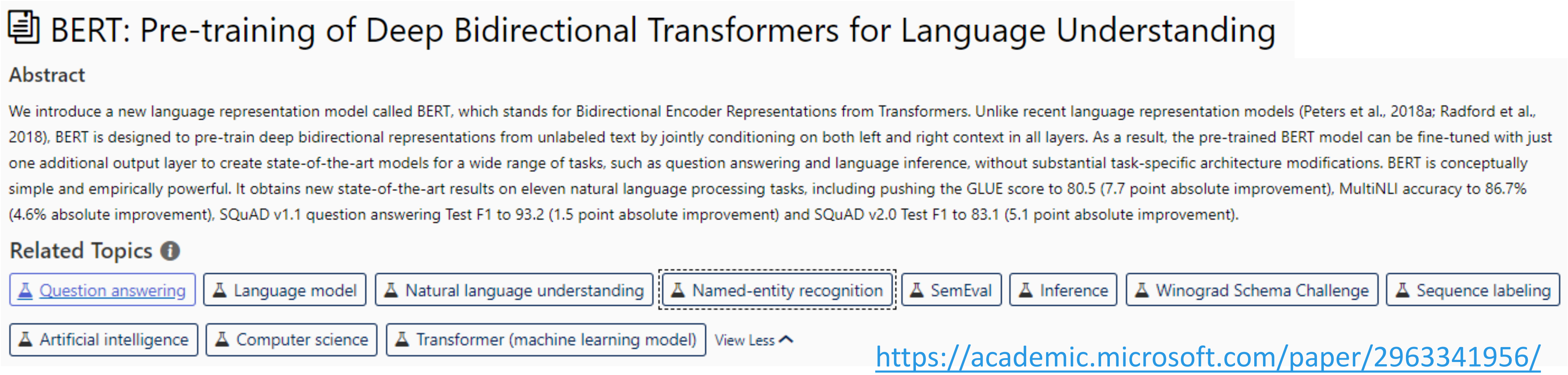
Hate Speech Detection

Different Text Classification Settings: Single-Label vs. Multi-Label

- **Single-label**: Each document belongs to one category.
  - E.g., Spam Detection



- **Multi-label**: Each document has multiple relevant labels.
  - E.g., Paper Topic Classification



BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding

Abstract

We introduce a new language representation model called BERT, which stands for Bidirectional Encoder Representations from Transformers. Unlike recent language representation models (Peters et al., 2018a; Radford et al., 2018), BERT is designed to pre-train deep bidirectional representations from unlabeled text by jointly conditioning on both left and right context in all layers. As a result, the pre-trained BERT model can be fine-tuned with just one additional output layer to create state-of-the-art models for a wide range of tasks, such as question answering and language inference, without substantial task-specific architecture modifications. BERT is conceptually simple and empirically powerful. It obtains new state-of-the-art results on eleven natural language processing tasks, including pushing the GLUE score to 80.5 (7.7 point absolute improvement), MultiNLI accuracy to 86.7% (4.6% absolute improvement), SQuAD v1.1 question answering Test F1 to 93.2 (1.5 point absolute improvement) and SQuAD v2.0 Test F1 to 83.1 (5.1 point absolute improvement).

Related Topics

Question answering | Language model | Natural language understanding | Named-entity recognition | SemEval | Inference | Winograd Schema Challenge | Sequence labeling | Artificial intelligence | Computer science | Transformer (machine learning model) | View Less

https://academic.microsoft.com/paper/2963341956/

Different Text Classification Settings: Flat vs. Hierarchical

- **Flat**: All labels are at the same granularity level
  - E.g., Sentiment Analysis of E-Commerce Reviews (1-5 stars)

It works, it's nice, comfortable, and easy to type on. Not loud (unless you're a key pounder)

This keyboard works. It's comfortable, sensitive enough for touch typers, very quiet by comparison to other mechanicals (unless, of course, you're a 'key pounder'), and the lit keys are excellent for people like me who tend to prefer to work in a cave-like environment.

https://www.amazon.com/gp/product/B089YFHYYS/

- **Hierarchical**: Labels are organized into a hierarchy representing their parent-child relationship
  - E.g., Paper Topic Classification (the arXiv category taxonomy)

BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding

We introduce a new language representation model called BERT, which stands for Bidirectional Encoder Representations from Transformers. Unlike recent language representation models, BERT is designed to pre-train deep bidirectional representations from unlabeled text by jointly conditioning on both left and right context in all layers. As a result, the pre-trained BERT model can be fine-tuned with just one additional output layer to create state-of-the-art models for a wide range of tasks, such as question answering and language inference, without substantial task-specific architecture modifications. BERT is conceptually simple and empirically powerful. It obtains new state-of-the-art results on eleven natural language processing tasks, including pushing the GLUE score to 80.5% (7.7% point absolute improvement), MultiNLI accuracy to 86.7% (4.6% absolute improvement), SQuAD v1.1 question answering Test F1 to 93.2 (1.5 point absolute improvement) and SQuAD v2.0 Test F1 to 83.1 (5.1 point absolute improvement).

Subjects: **Computation and Language (cs.CL)**
Cite as: arXiv:1810.04805 **[cs.CL]**
(or arXiv:1810.04805v2 **[cs.CL]** for this version)

https://arxiv.org/abs/1810.04805

Natural Language Understanding (NLU)

- The widely used General Language Understanding Evaluation (**GLUE**) benchmark has 7 tasks.
  - **MNLI**: Multi-genre Natural Language Inference aims to predict whether a given premise sentence **entails, contradicts or neutral** with respect to a given hypothesis sentence.
  - **QQP**: Quora Question Pairs aims to determine whether a pair of questions asked are **semantically equivalent**.
  - **QNLI**: Question Natural Language Inference aims to predict whether a given sentence **contains the answer** to a given question sentence.
  - **SST-2**: Stanford Sentiment Treebank aims to determine if a movie review has **positive or negative sentiment**.

Task	Label	Prompt
SST-2	positive negative	Rating: 5.0 $\boldsymbol{x}^g$ Rating: 1.0 $\boldsymbol{x}^g$
MNLI	entailment neutral contradiction	$\boldsymbol{x}^s$. In other words, $\boldsymbol{x}^g$ $\boldsymbol{x}^s$. Furthermore, $\boldsymbol{x}^g$ There is a rumor that $\boldsymbol{x}^s$. However, the truth is: $\boldsymbol{x}^g$
QNLI	entailment not entailment	$\boldsymbol{x}^s$? $\boldsymbol{x}^g$ $\boldsymbol{x}^s$? ... $\boldsymbol{x}^g$
RTE	entailment not entailment	$\boldsymbol{x}^s$. In other words, $\boldsymbol{x}^g$ $\boldsymbol{x}^s$. Furthermore, $\boldsymbol{x}^g$
MRPC	equivalent not equivalent	$\boldsymbol{x}^s$. In other words, $\boldsymbol{x}^g$ $\boldsymbol{x}^s$. Furthermore, $\boldsymbol{x}^g$
QQP	equivalent not equivalent	$\boldsymbol{x}^s$? In other words, $\boldsymbol{x}^g$ $\boldsymbol{x}^s$? Furthermore, $\boldsymbol{x}^g$

Natural Language Understanding (NLU)

- The widely used General Language Understanding Evaluation (**GLUE**) benchmark has 7 tasks.
  - **CoLA**: Corpus of Linguistic Acceptability aims to determine whether a given sentence is **linguistically acceptable** or not.
  - **RTE**: Recognizing Textual Entailment aims to predict whether a given premise sentence **entails** a given hypothesis sentence or not.
  - **MRPC**: Microsoft Research Paraphrase Corpus aims to predict whether two sentences are **semantically equivalent** or not.
- Many NLU tasks can be cast as a text classification problem. They classify either one text unit or a pair of text units.

Task	Label	Prompt
SST-2	positive negative	Rating: 5.0 $\boldsymbol{x}^g$ Rating: 1.0 $\boldsymbol{x}^g$
MNLI	entailment neutral contradiction	$\boldsymbol{x}^s$. In other words, $\boldsymbol{x}^g$ $\boldsymbol{x}^s$. Furthermore, $\boldsymbol{x}^g$ There is a rumor that $\boldsymbol{x}^s$. However, the truth is: $\boldsymbol{x}^g$
QNLI	entailment not entailment	$\boldsymbol{x}^s$? $\boldsymbol{x}^g$ $\boldsymbol{x}^s$? ... $\boldsymbol{x}^g$
RTE	entailment not entailment	$\boldsymbol{x}^s$. In other words, $\boldsymbol{x}^g$ $\boldsymbol{x}^s$. Furthermore, $\boldsymbol{x}^g$
MRPC	equivalent not equivalent	$\boldsymbol{x}^s$. In other words, $\boldsymbol{x}^g$ $\boldsymbol{x}^s$. Furthermore, $\boldsymbol{x}^g$
QQP	equivalent not equivalent	$\boldsymbol{x}^s$? In other words, $\boldsymbol{x}^g$ $\boldsymbol{x}^s$? Furthermore, $\boldsymbol{x}^g$

Outline

- Why do we care weakly-supervised text classification/NLU?
- Weakly-supervised text classification
  - ConWea [ACL'20], LOTClass [EMNLP'20], X-Class [NAACL'21], PromptClass [arXiv'23]
- Weakly-supervised structure-enhanced text classification
  - Taxonomy-enhanced: TaxoClass [NAACL'21]
  - Metadata-enhanced: MICoL [WWW'22], MAPLE [WWW'23]
- Weakly-supervised NLU
  - Zero-shot: ZeroGen [EMNLP'22], SuperGen [NeurIPS'22]
  - Few-shot: FewGen [ICML'23]

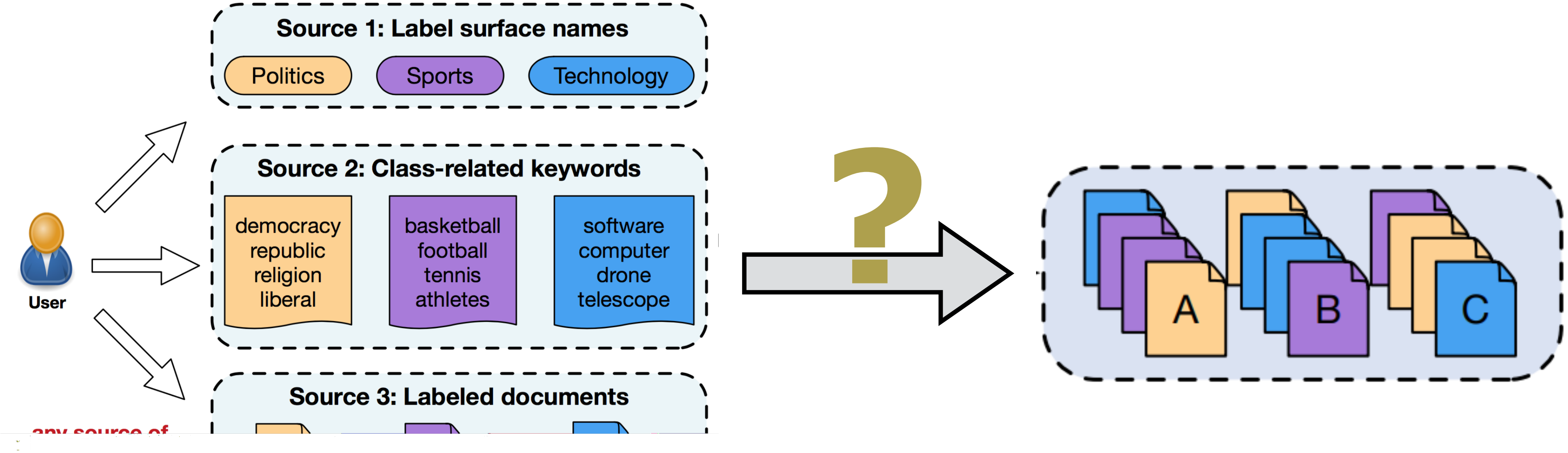
Weakly-Supervised Text Classification: Motivation

- Supervised text classification models (especially recent deep neural models) rely on a significant number of manually labeled training documents to achieve good performance.
- Collecting such training data is usually expensive and time-consuming. In some domains (e.g., scientific papers), annotations must be acquired from domain experts, which incurs additional cost.

- While users cannot afford to label sufficient documents for training a deep neural classifier, they can provide a small amount of seed information:
  - Category names or category-related keywords
  - A small number of labeled documents

Weakly-Supervised Text Classification: Definition

- Text classification without massive human-annotated training data
  - **Keyword-level weak supervision**: category names or a few relevant keywords
  - **Document-level weak supervision**: a small set of labeled docs



General Ideas to Perform Weakly-Supervised Text Classification

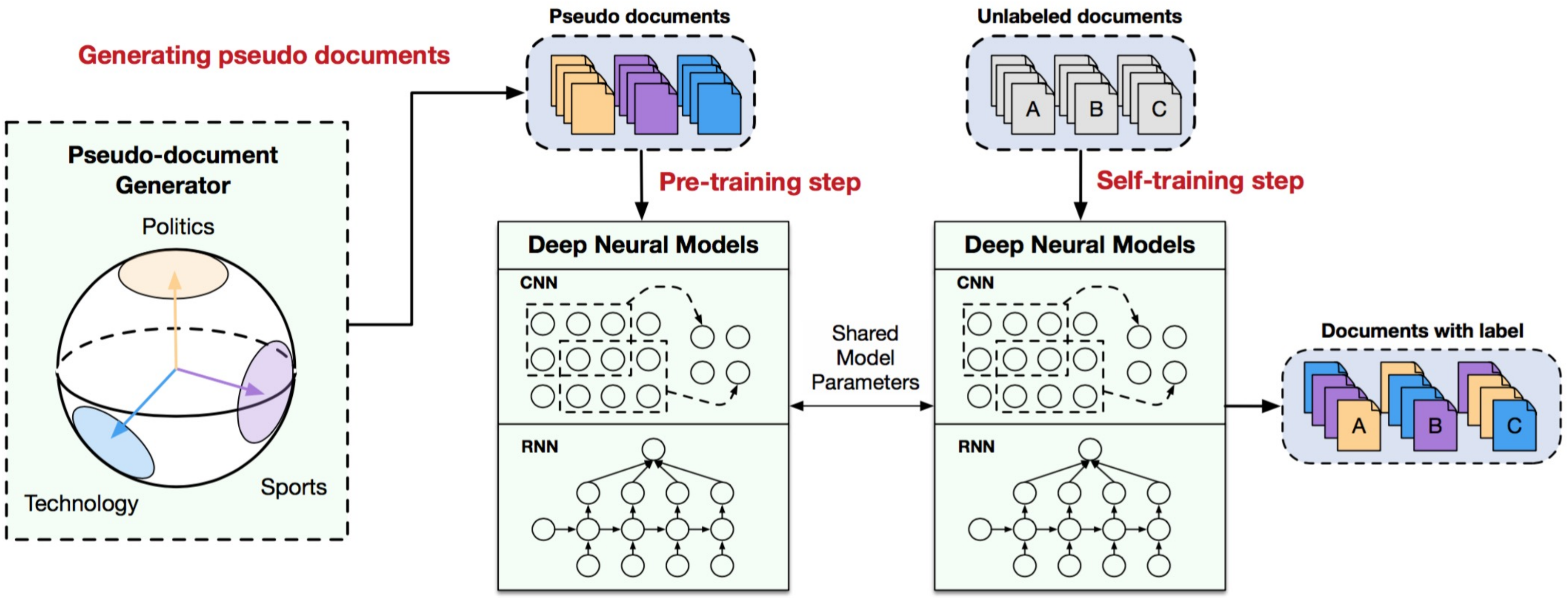
- Joint representation learning
  - Put words, labels, and documents into the same latent space using embedding learning or pre-trained language models

- Pseudo training data generation
  - Retrieve some unlabeled documents or synthesize some artificial documents using text embeddings or contextualized representations
  - Give them pseudo labels to train a text classifier

- Transfer the knowledge of pre-trained language models to classification tasks

An Example – WeSTClass

- Embed all words (including label names and keywords) into the same space
- Pseudo document generation: generate pseudo documents from seeds
- Self-training: train deep neural nets (CNN, RNN) with bootstrapping



Meng, Y., Shen, J., Zhang, C., & Han, J. "Weakly-supervised neural text classification", CIKM'18.

Outline

- Why do we care weakly-supervised text classification/NLU?
- Weakly-supervised text classification
  - ConWea [ACL'20], LOTClass [EMNLP'20], X-Class [NAACL'21], PromptClass [arXiv'23]
- Weakly-supervised structure-enhanced text classification
  - Taxonomy-enhanced: TaxoClass [NAACL'21]
  - Metadata-enhanced: MICoL [WWW'22], MAPLE [WWW'23]
- Weakly-supervised NLU
  - Zero-shot: ZeroGen [EMNLP'22], SuperGen [NeurIPS'22]
  - Few-shot: FewGen [ICML'23]

ConWea: Disambiguating User-Provided Keywords

- User-provided seed words may be ambiguous.
- Example:

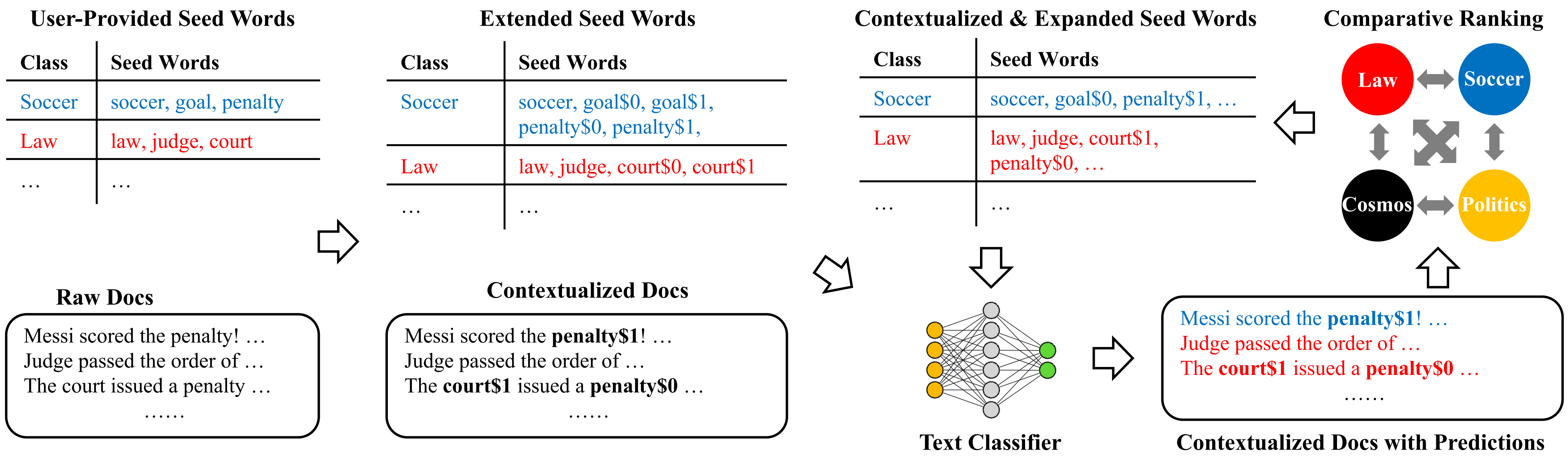
Class	Seed words
Soccer	soccer, goal, penalty
Law	law, judge, court

  - Classify the following sentences:
    - Messi scored the penalty.
    - John was issued a death penalty.
- Disambiguate the "senses" based on contextualized representations

Mekala, D. & Shang, J. "Contextualized Weak Supervision for Text Classification", ACL'20.

ConWea: Clustering for Disambiguation

- For each word, find all its occurrences in the input corpus
  - Run BERT to get their contextualized representations
  - Run a clustering method (e.g., K-Means) to obtain clusters for different "senses"



ConWea: Experiment Results

- Ablations:
  - ConWea-NoCon: Variant of ConWea trained without contextualization.
  - ConWea-NoExpan: Variant of ConWea trained without seed expansion.
  - ConWea-WSD: Variant of ConWea with contextualization replaced by a word sense disambiguation algorithm.

		NYT				20 Newsgroup			
		5-Class (Coarse)		**25-Class** (Fine)		**6-Class** (Coarse)		**20-Class** (Fine)	
	Methods	Micro-F_1	Macro-F_1	Micro-F_1	Macro-F_1	Micro-F_1	Macro-F_1	Micro-F_1	Macro-F_1
Baselines	IR-TF-IDF	0.65	0.58	0.56	0.54	0.49	0.48	0.53	0.52
	Dataless	0.71	0.48	0.59	0.37	0.50	0.47	0.61	0.53
	Word2Vec	0.92	0.83	0.69	0.47	0.51	0.45	0.33	0.33
	Doc2Cube	0.71	0.38	0.67	0.34	0.40	0.35	0.23	0.23
	WeSTClass	0.91	0.84	0.50	0.36	0.53	0.43	0.49	0.46
	ConWea	**0.95**	**0.89**	**0.91**	**0.79**	**0.62**	**0.57**	**0.65**	**0.64**
Ablations	ConWea-NoCon	0.91	0.83	0.89	0.74	0.53	0.50	0.58	0.57
	ConWea-NoExpan	0.92	0.85	0.76	0.66	0.58	0.53	0.58	0.57
	ConWea-WSD	0.83	0.78	0.72	0.64	0.52	0.46	0.49	0.47
Upper bound	HAN-Supervised	0.96	0.92	0.94	0.82	0.90	0.88	0.83	0.83

LOTClass: Find Similar Meaning Words with Label Names

- Find topic words based on label names
  - Overcome the low semantic coverage of label names
- Use language models to predict what words can replace the label names
  - Interchangeable words are likely to have similar meanings

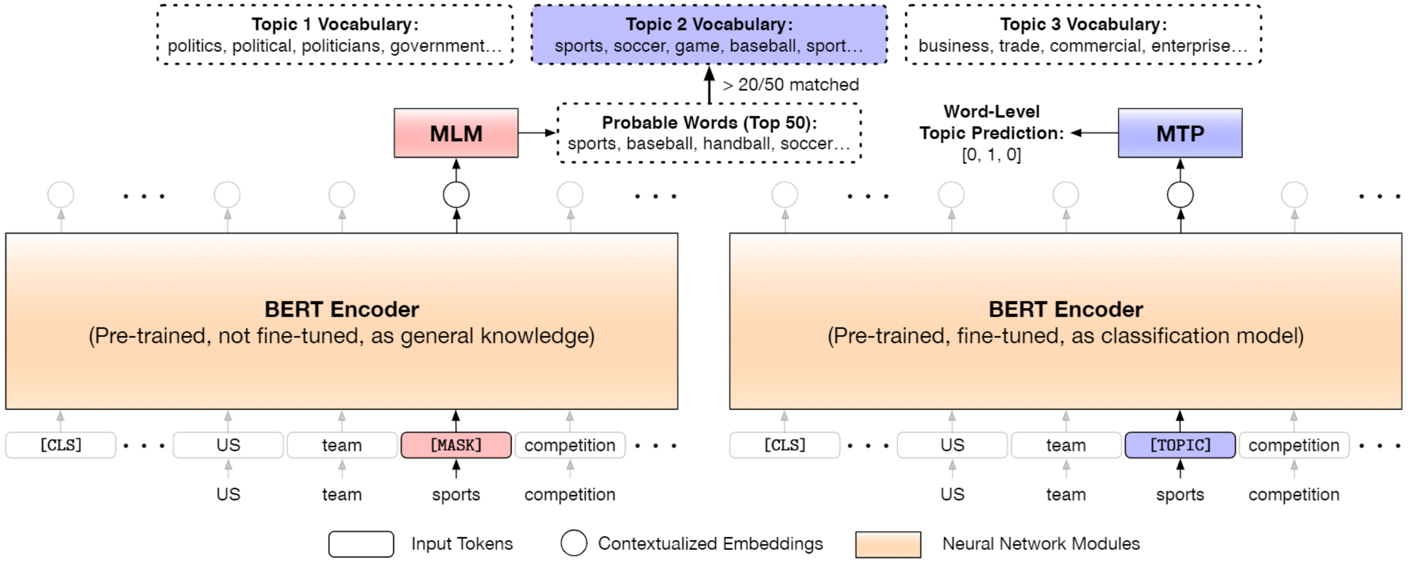
Sentence	Language Model Prediction
The oldest annual US team **sports** competition that includes professionals is not in baseball, or football or basketball or hockey. It's in soccer.	sports, baseball, handball, soccer, basketball, football, tennis, sport, championship, hockey, ...
Samsung's new SPH-V5400 mobile phone **sports** a built-in 1-inch, 1.5-gigabyte hard disk that can store about 15 times more data than conventional handsets, Samsung said.	has, with, features, uses, includes, had, is, contains, featured, have, incorporates, requires, offers, ...

Table 1: BERT language model prediction (sorted by probability) for the word to appear at the position of "sports" under different contexts. The two sentences are from *AG News* corpus.

Meng, Y., Zhang, Y., Huang, J., Xiong, C., Ji, H., Zhang, C., & Han, J. "Text Classification Using Label Names Only: A Language Model Self-Training Approach", EMNLP'20.

LOTClass: Contextualized Word-Level Topic Prediction

- Context-free matching of topic words is inaccurate
  - “Sports” does not always imply the topic “sports”
- Contextualized topic prediction:
  - Predict a word’s implied topic under specific contexts
  - We regard a word as “topic indicative” only when its top replacing words have enough overlap with the topic vocabulary.



LOTClass: Experiment Results

- Achieve around 90% accuracy on four benchmark datasets by only using at most 3 words (1 in most cases) per class as the label name
  - Outperforming previous weakly-supervised approaches significantly
  - Comparable to state-of-the-art semi-supervised models

Supervision Type	Methods	AG News	DBPedia	IMDB	Amazon
Weakly-Sup.	Dataless (Chang et al., 2008)	0.696	0.634	0.505	0.501
	WeSTClass (Meng et al., 2018)	0.823	0.811	0.774	0.753
	BERT w. simple match	0.752	0.722	0.677	0.654
	Ours w/o. self train	0.822	0.850	0.844	0.781
	Ours	**0.864**	**0.889**	**0.894**	**0.906**
Semi-Sup.	UDA (Xie et al., 2019)	0.869	0.986	0.887	0.960
Supervised	char-CNN (Zhang et al., 2015)	0.872	0.983	0.853	0.945
	BERT (Devlin et al., 2019)	0.944	0.993	0.937	0.972

How Powerful Are Vanilla BERT Representations in Category Prediction?

- An average of BERT representations of all tokens in a sentence/document preserves domain information well [1].

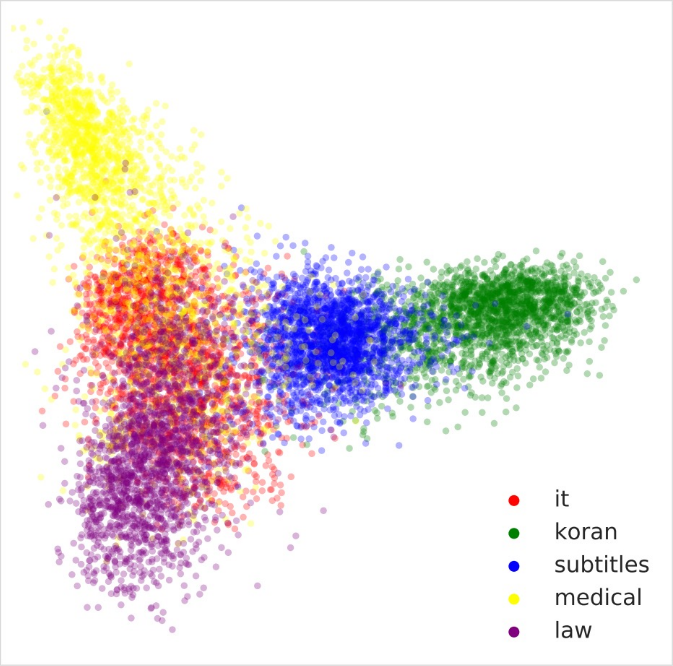


Figure 1: A 2D visualization of average-pooled BERT hidden-state sentence representations using PCA. The colors represent the domain for each sentence.

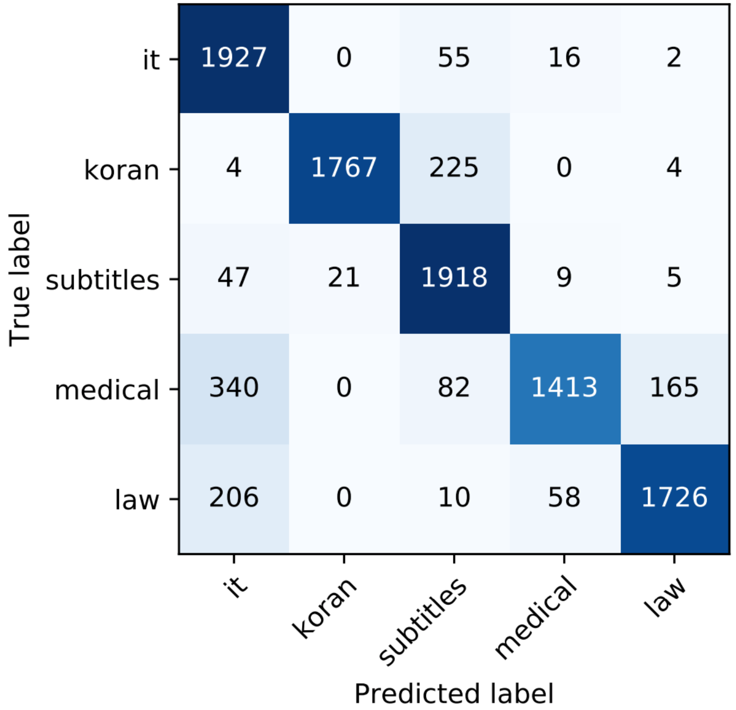
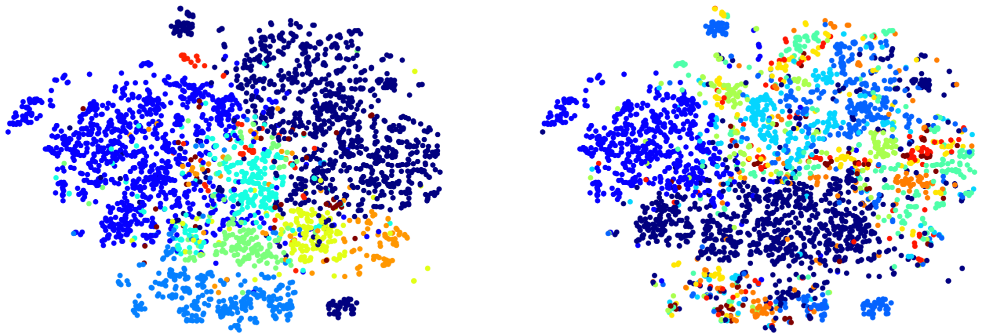


Figure 2: A confusion matrix for clustering with k=5 using BERT-base.

[1] Aharoni, R., & Goldberg, Y. "Unsupervised domain clusters in pretrained language models." ACL'20.

X-Class: Class-Oriented BERT Representations

- A simple idea for text classification
  - Learn representations for documents
  - Set the number of clusters as the number of classes
  - Hope their clustering results are almost the same as the desired classification
- However, the same corpus could be classified differently



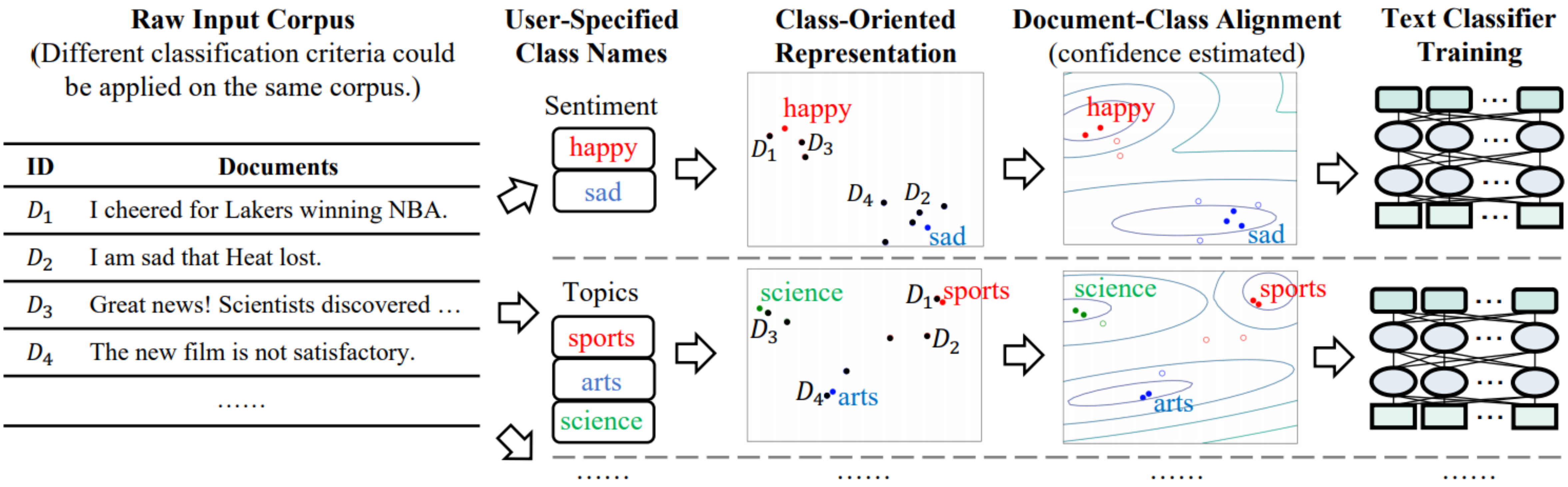
(a) NYT-Topics (b) NYT-Locations

Figure 1: Visualizations of News using Average BERT Representations. Colors denote different classes.

Wang, Z., Mekala, D., & Shang, J. “X-Class: Text Classification with Extremely Weak Supervision”, NAACL’21.

X-Class: Class-Oriented BERT Representations

- Clustering for classification based on class-oriented representations



X-Class: Experiment Results

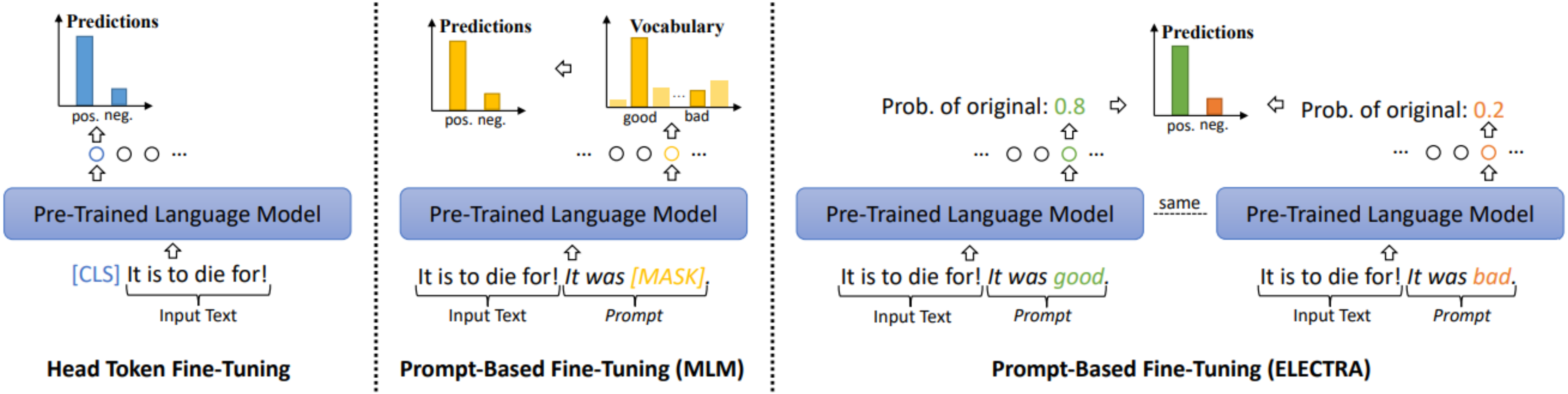
- WeSTClass & ConWea consume at least 3 seed words per class
- LOTClass & X-Class use category names only

	AGNews	20News	NYT-Small	NYT-Topic	NYT-Location	Yelp	DBpedia
Corpus Domain	News	News	News	News	News	Reviews	Wikipedia
Class Criterion	Topics	Topics	Topics	Topics	Locations	Sentiment	Ontology
# of Classes	4	5	5	9	10	2	14
# of Documents	120,000	17,871	13,081	31,997	31,997	38,000	560,000
Imbalance	1.0	2.02	16.65	27.09	15.84	1.0	1.0

Model	AGNews	20News	NYT-Small	NYT-Topic	NYT-Location	Yelp	DBpedia
Supervised	93.99/93.99	96.45/96.42	97.95/95.46	94.29/89.90	95.99/94.99	95.7/95.7	98.96/98.96
WeSTClass	82.3/82.1	71.28/69.90	91.2/83.7	68.26/57.02	63.15/53.22	81.6/81.6	81.1/ N/A
ConWea	74.6/74.2	75.73/73.26	95.23/90.79	**81.67/71.54**	85.31/83.81	71.4/71.2	N/A
LOTClass	**86.89/86.82**	73.78/72.53	78.12/56.05	67.11/43.58	58.49/58.96	87.75/87.68	86.66/85.98
X-Class	84.8/84.65	**81.36/80.6**	**96.67/92.98**	80.6/69.92	**90.5/89.81**	**88.36/88.32**	**91.33/91.14**
X-Class-Rep	77.92/77.03	75.14/73.24	92.13/83.94	77.85/65.38	86.7/87.36	77.87/77.05	74.06/71.75
X-Class-Align	83.1/83.05	79.28/78.62	96.34/92.08	79.64/67.85	88.58/88.02	87.16/87.1	87.37/87.28

PromptClass: Prompt-based Fine-tuning for Text Classification

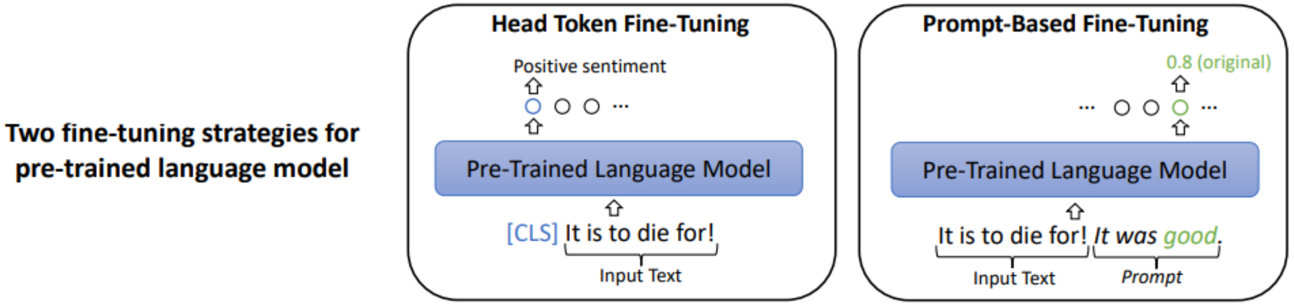
- **Head token fine-tuning** randomly initializes a linear classification head and directly predicts class distribution using the [CLS] token, which needs a substantial amount of training data.
- **Prompt-based fine-tuning for MLM-based PLM** converts the document into the masked token prediction problem by reusing the pre-trained MLM head.
- **Prompt-based fine-tuning for ELECTRA-style PLM** converts documents into the replaced token detection problem by reusing the pre-trained discriminative head.

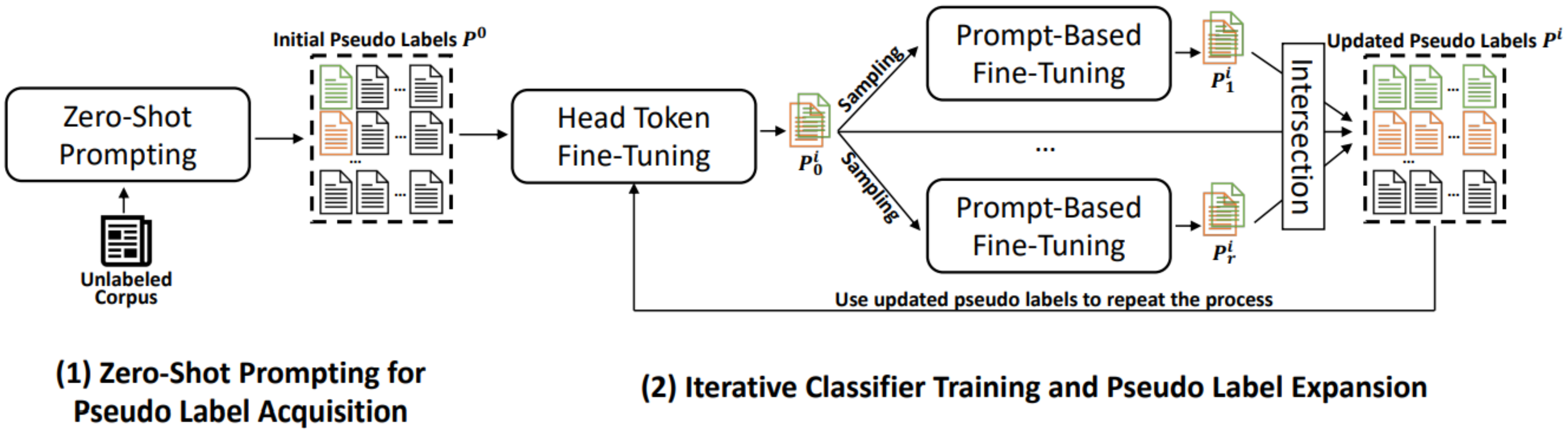


Zhang, Y., Jiang, M., Meng, Y., Zhang, Y., & Han, J. "PromptClass: Weakly-Supervised Text Classification with Prompting Enhanced Noise-Robust Self-Training", arXiv'23.

PromptClass: Integrating Head Token & Prompt-based Fine-tuning

- Why do we need prompts to get pseudo training data?
  - Simple keyword matching may induce errors.
  - E.g., "*die*" is a negative word, but a food review "It is to *die* for!" implies a strong positive sentiment.





PromptClass: Experiment Results

- PromptClass is on par with the fully supervised text classifier on sentiment analysis datasets (i.e., Yelp and IMDB).

Methods	AGNews		20News		Yelp		IMDB	
	Micro-F1	Macro-F1	Micro-F1	Macro-F1	Micro-F1	Macro-F1	Micro-F1	Macro-F1
WeSTClass	0.823	0.821	0.713	0.699	0.816	0.816	0.774	-
ConWea	0.746	0.742	0.757	0.733	0.714	0.712	-	-
LOTClass	0.869	0.868	0.738	0.725	0.878	0.877	0.865	-
XClass	0.857	0.857	0.786	0.778	0.900	0.900	-	-
ClassKG†	0.881	0.881	<u>0.811</u>	**0.820**	0.918	0.918	0.888	0.888
RoBERTa (0-shot)	0.581	0.529	0.507‡	0.445‡	0.812	0.808	0.784	0.780
ELECTRA (0-shot)	0.810	0.806	0.558	0.529	0.820	0.820	0.803	0.802
PromptClass								
ELECTRA+BERT	<u>0.884</u>	<u>0.884</u>	0.789	0.791	0.919	0.919	0.905	0.905
RoBERTa+RoBERTa	**0.895**	**0.895**	0.755‡	0.760‡	<u>0.920</u>	<u>0.920</u>	<u>0.906</u>	<u>0.906</u>
ELECTRA+ELECTRA	<u>0.884</u>	<u>0.884</u>	**0.816**	<u>0.817</u>	**0.957**	**0.957**	**0.931**	**0.931**
Fully Supervised	0.940	0.940	0.965	0.964	0.957	0.957	0.945	-

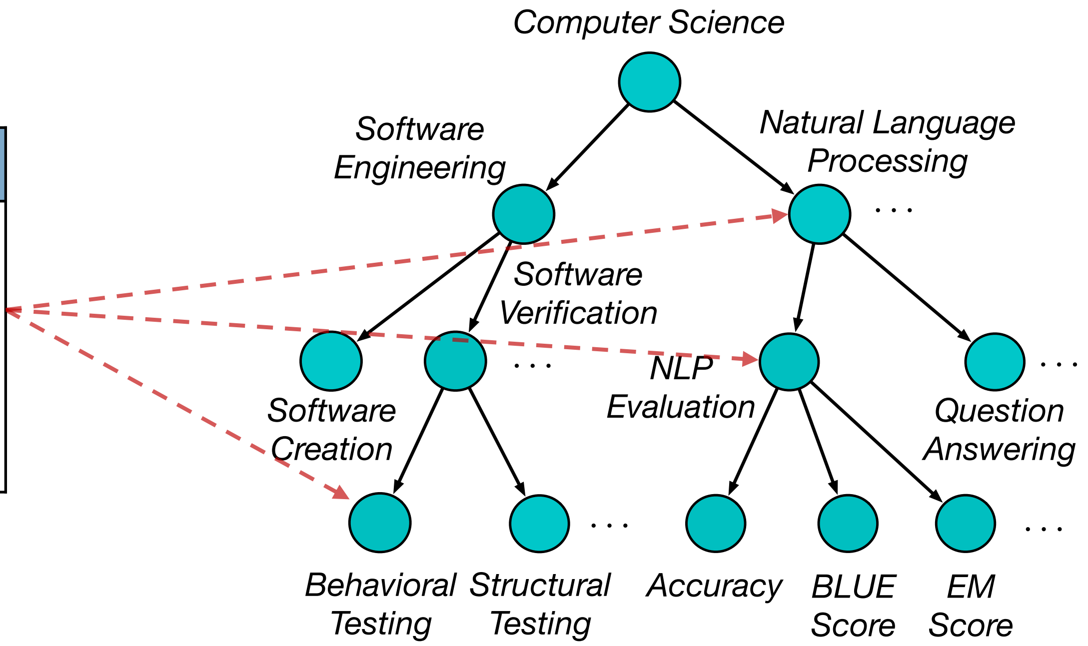
Outline

- Why do we care weakly-supervised text classification/NLU?
- Weakly-supervised text classification
  - ConWea [ACL'20], LOTClass [EMNLP'20], X-Class [NAACL'21], PromptClass [arXiv'23]
- Weakly-supervised structure-enhanced text classification
  - Taxonomy-enhanced: TaxoClass [NAACL'21]
  - Metadata-enhanced: MICoL [WWW'22], MAPLE [WWW'23]
- Weakly-supervised NLU
  - Zero-shot: ZeroGen [EMNLP'22], SuperGen [NeurIPS'22]
  - Few-shot: FewGen [ICML'23]

TaxoClass: Weakly-supervised Hierarchical Multi-Label Text Classification

- The taxonomy is a directed acyclic graph (DAG)
- Each paper can have multiple categories distributed on different paths
- Category names can be phrases and may not appear in the corpus

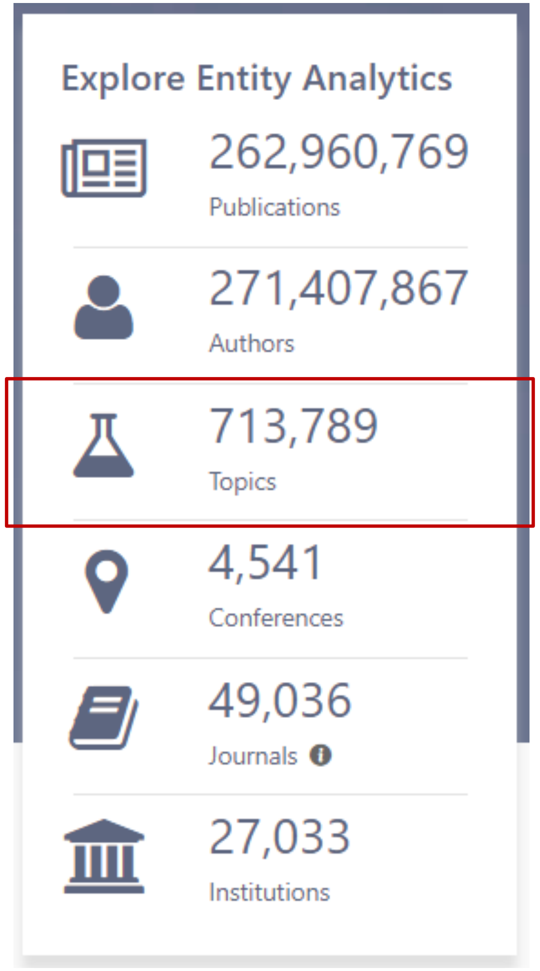
Document
Measuring held-out accuracy often overestimates the performance of ***NLP*** models… Inspired by principles of ***behavioral testing*** in software engineering, we introduce CheckList, a task-agnostic methodology for ***testing NLP models***…



Shen, J., Qiu, W., Meng, Y., Shang, J., Ren, X., & Han, J., "TaxoClass: Hierarchical Multi-Label Text Classification Using Only Class Names", NAACL'21.

TaxoClass: Why Category Names Only?

- Taxonomies for multi-label text classification are often big.
  - Amazon Product Catalog: $\times 10^4$ categories
  - MeSH Taxonomy (for medical papers): $\times 10^4$ categories
  - Microsoft Academic Taxonomy: $\times 10^5$ labels
- Impossible for users to provide even a small set of (e.g., 3) keywords/labeled documents for each category

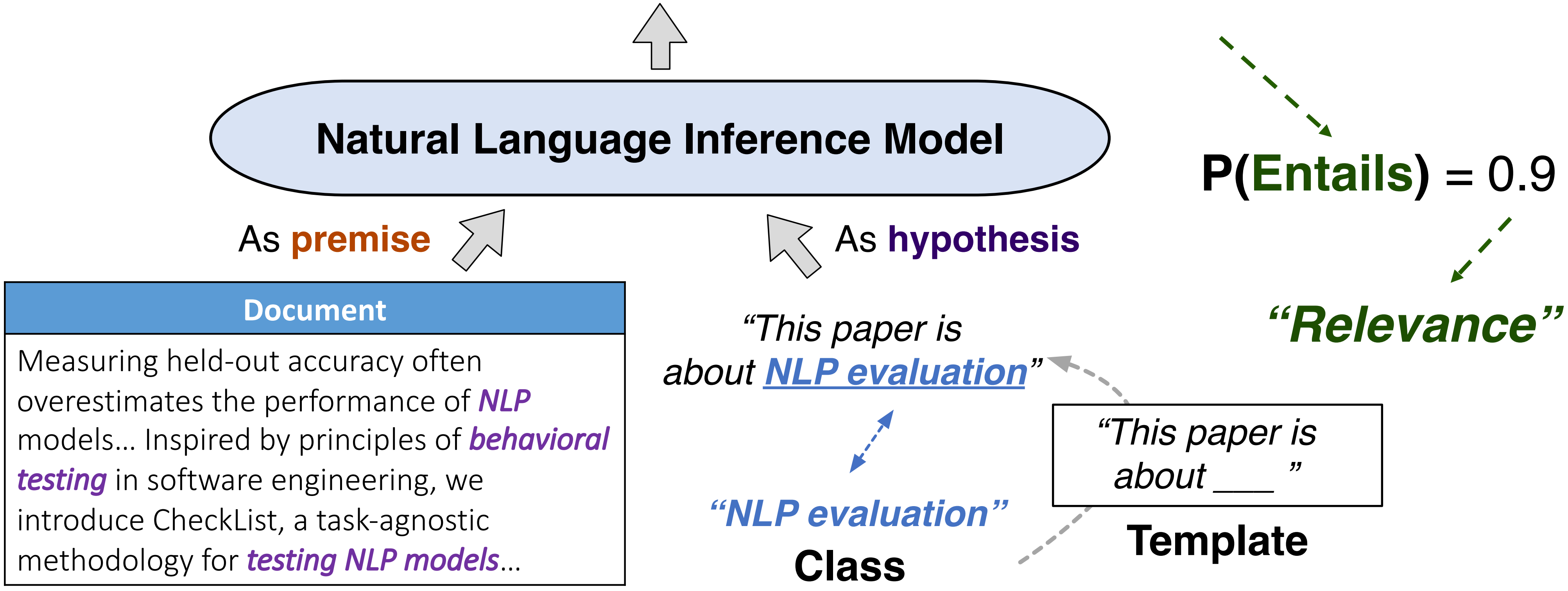


https://academic.microsoft.com/home

TaxoClass: Document-Class Relevance Calculation

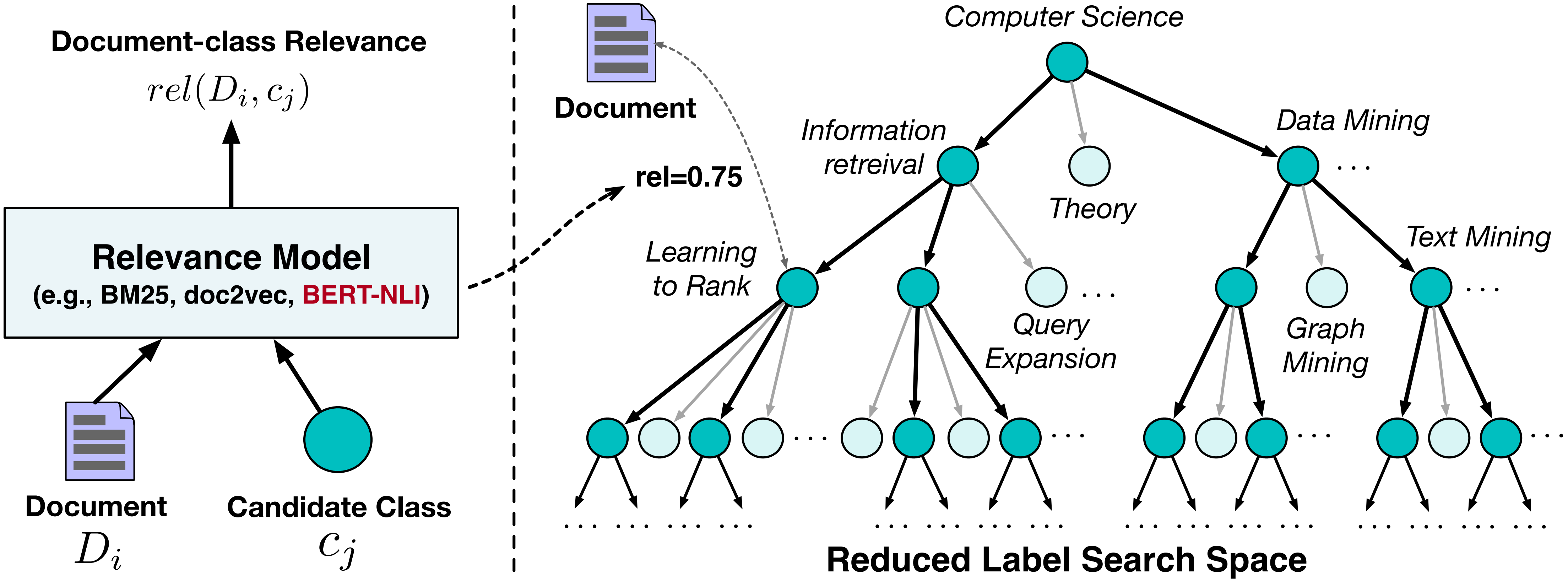
- How to use the knowledge from pre-trained LMs?
- Relevance model: BERT/RoBERTa fine-tuned on the NLI task
  - https://huggingface.co/roberta-large-mnli

After reading the **premise**, can you infer the **hypothesis**?



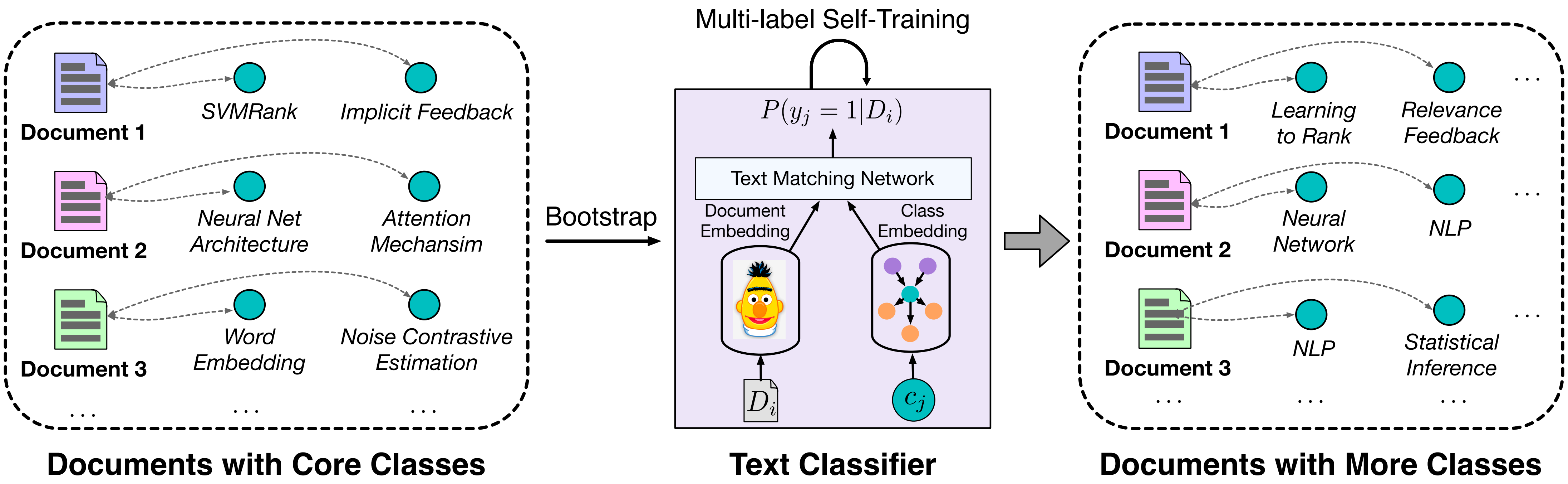
TaxoClass: Top-Down Exploration

- How to use the taxonomy?
- Shrink the label search space with top-down exploration
  - Use a relevance model to filter out completely irrelevant classes



TaxoClass: Identify Core Classes and More Classes

- Identify document core classes in reduced label search space
- Generalize from core classes with bootstrapping and self-training



TaxoClass: Experiment Results

Methods	Amazon		DBPedia	
	Example-F1	P@1	Example-F1	P@1
WeSHClass (Meng et al., AAAI'19)	0.246	0.577	0.305	0.536
SS-PCEM (Xiao et al., WebConf'19)	0.292	0.537	0.385	0.742
Semi-BERT (Devlin et al., NAACL'19)	0.339	0.592	0.428	0.761
Hier-0Shot-TC (Yin et al., EMNLP'19)	0.474	0.714	0.677	0.787
TaxoClass (ours)	**0.593**	**0.812**	**0.816**	**0.894**

Weakly-supervised multi-class classification method

Semi-supervised methods using 30% of training set

Zero-shot method

- **vs. WeSHClass**: better model document-class relevance
- **vs. SS-PCEM, Semi-BERT**: better leverage supervision signals from taxonomy
- **vs. Hier-0Shot-TC**: better capture domain-specific information from core classes

Example-F1 $= \frac{1}{N}\sum_{i=1}^{N}\frac{2|true_i \cap pred_i|}{|true_i|+|pred_i|}$, **P@1** $= \frac{\#docs\ with\ top\!-\!1\ pred\ dorrect}{\#total\ docs}$

Outline

- Why do we care weakly-supervised text classification/NLU?
- Weakly-supervised text classification
  - ConWea [ACL'20], LOTClass [EMNLP'20], X-Class [NAACL'21], PromptClass [arXiv'23]
- Weakly-supervised structure-enhanced text classification
  - Taxonomy-enhanced: TaxoClass [NAACL'21]
  - Metadata-enhanced: MICoL [WWW'22], MAPLE [WWW'23]
- Weakly-supervised NLU
  - Zero-shot: ZeroGen [EMNLP'22], SuperGen [NeurIPS'22]
  - Few-shot: FewGen [ICML'23]

Metadata

- Metadata is prevalent in many text sources
  - **GitHub repositories**: User, Tag
  - **Tweets**: User, Hashtag
  - **Amazon reviews:** User, Product
  - **Scientific papers:** Author, Venue, Reference

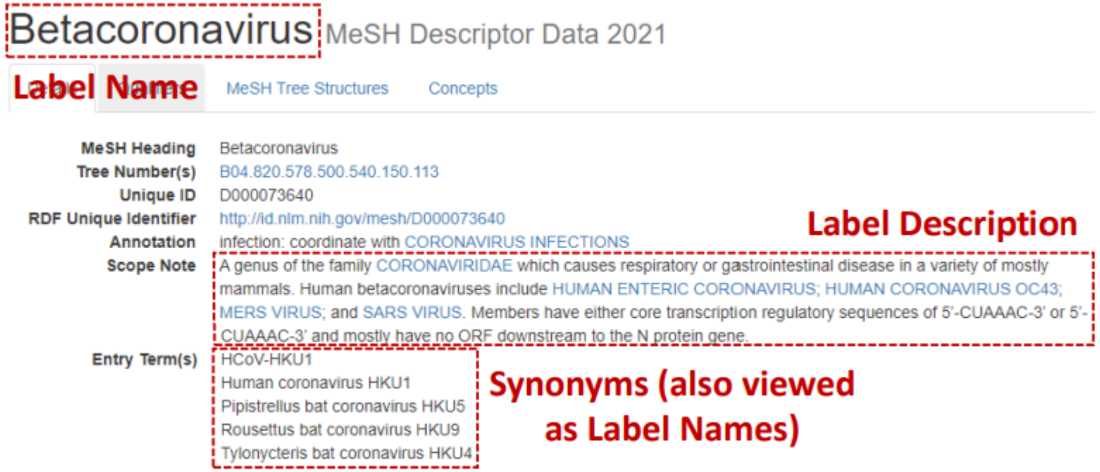
- How to leverage these heterogenous signals in the categorization process?

(a) GitHub Repository (b) Tweet (c) Amazon Review

MICoL: Metadata-Induced Contrastive Learning for Zero-Shot Multi-Label Text Classification

- Input
  - A set of labels. Each label has its name and description.
  - A large set of unlabeled documents associated with metadata (e.g., authors, venue, references) that can connect the documents together.
- Output
  - A multi-label text classifier. Given some new documents, the classifier can predict relevant labels for each document.

(a) Label "Webgraph" from Microsoft Academic (https://academic.microsoft.com/topic/2777569578/).

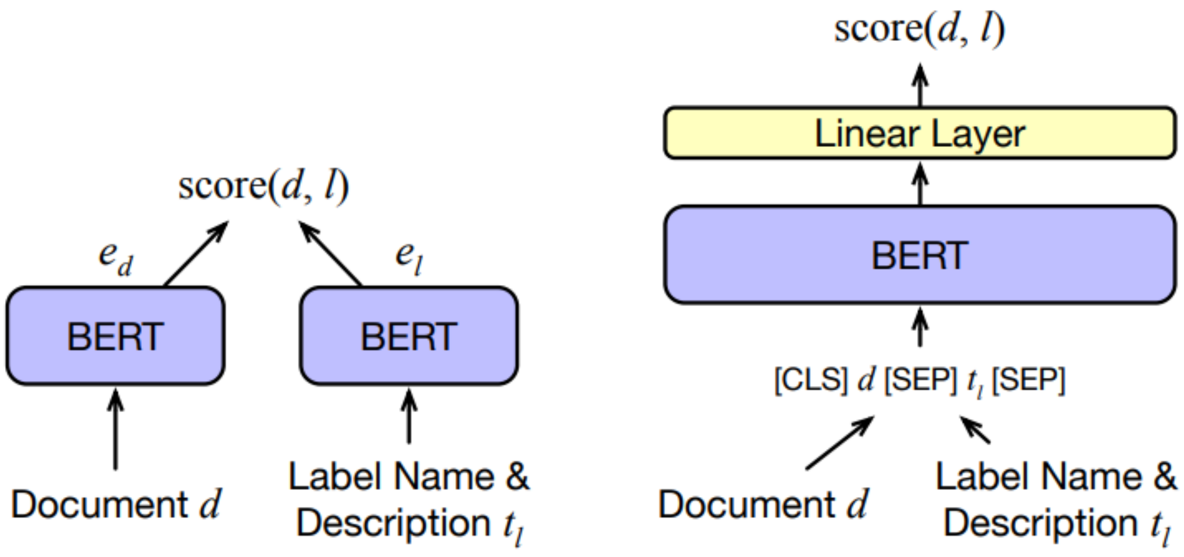


(b) Label "Betacoronavirus" from PubMed (https://meshb.nlm.nih.gov/record/ui?ui=D000073640).

Zhang, Y., Shen, Z., Wu, C., Xie, B., Wang, Y., Wang, K., & Han, J. "Metadata-Induced Contrastive Learning for Zero-Shot Multi-Label Text Classification", WWW'22.

Pretrained Language Models for Multi-Label Text Classification

- If we could have some labeled documents, …
  - We can use relevant (document, label) pairs to fine-tune the pre-trained LM.
  - Both Bi-Encoder and Cross-Encoder are applicable.

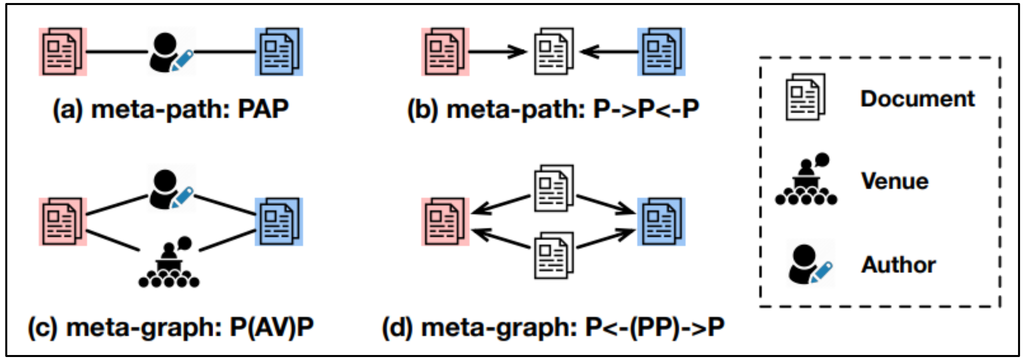


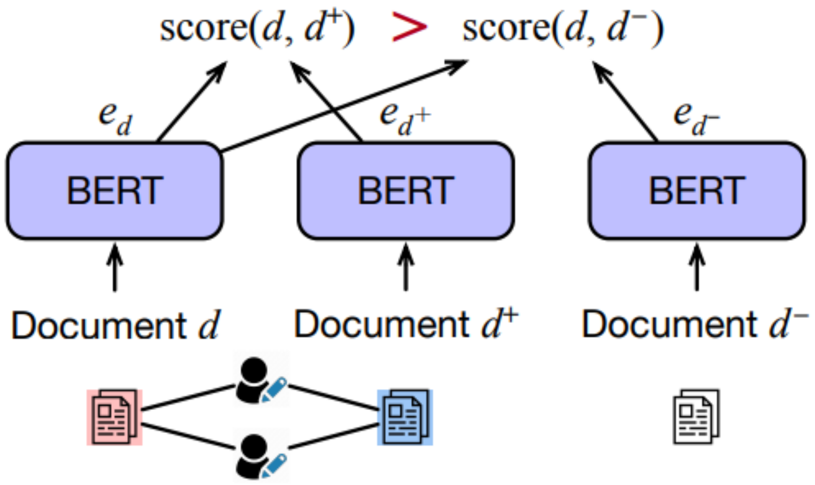
(a) Bi-Encoder (b) Cross-Encoder

- However, we do not have any labeled documents!!!

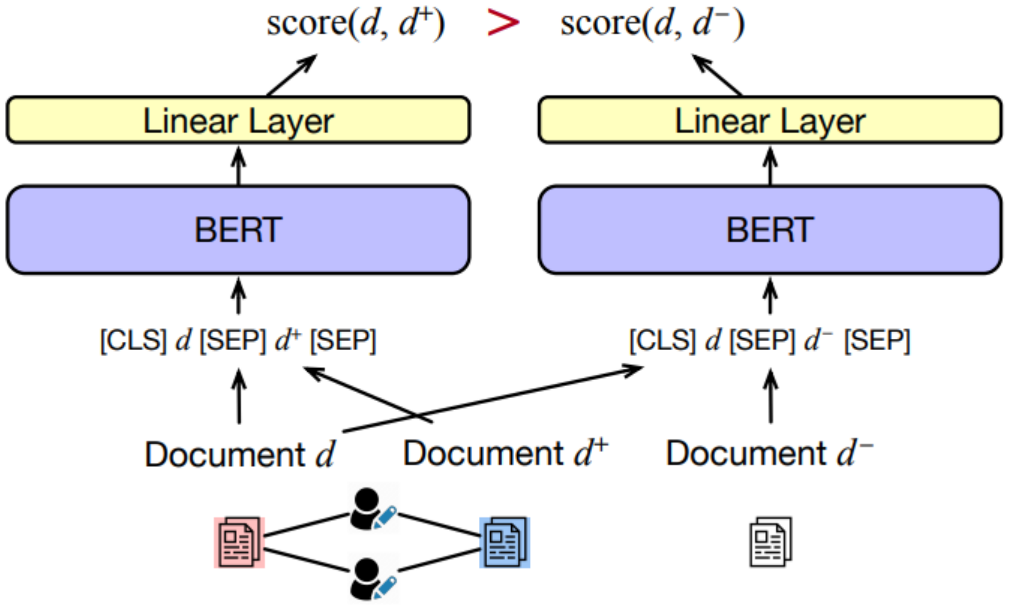
Metadata-Induced Contrastive Learning

- Contrastive learning [1]: Instead of training the model to know "what is what" (e.g., relevant (document, label) pairs), train it to know "what is similar with what" (e.g., similar (document, document) pairs).
- Using metadata to define similar (document, document) pairs.





(a) Bi-Encoder fine-tuning



(b) Cross-Encoder fine-tuning

[1] Chen, T., Kornblith, S., Norouzi, M., & Hinton, G. A simple framework for contrastive learning of visual representations. ICML'20.

MICoL: Experiment Results

- MICoL significantly outperforms text-based contrastive learning baselines.
- MICoL is competitive with the supervised SOTA trained on 10K–50K labeled documents.

	Algorithm	MAG-CS [49]					PubMed [24]				
		P@1	P@3	P@5	NDCG@3	NDCG@5	P@1	P@3	P@5	NDCG@3	NDCG@5
Zero-shot	Doc2Vec [31]	0.5697**	0.4613**	0.3814**	0.5043**	0.4719**	0.3888**	0.3283**	0.2859**	0.3463**	0.3252**
	SciBERT [2]	0.6440**	0.5030**	0.4011**	0.5545**	0.5061**	0.4427**	0.3572**	0.3031**	0.3809**	0.3510**
	ZeroShot-Entail [61]	0.6649**	0.5003**	0.3959**	0.5570**	0.5057**	0.5275**	0.4021	**0.3299**	0.4352	**0.3913**
	SPECTER [8]	0.7107**	0.5381**	0.4184**	0.5979**	0.5365**	0.5286**	0.3923**	0.3181**	0.4273**	0.3815**
	EDA [53]	0.6442**	0.4939**	0.3948**	0.5471**	0.5000**	0.4919	0.3754*	0.3101*	0.4058*	0.3667*
	UDA [57]	0.6291**	0.4848**	0.3897**	0.5362**	0.4918**	0.4795**	0.3696**	0.3067**	0.3986**	0.3614**
	MICoL (Bi-Encoder, $P \rightarrow P \leftarrow P$)	0.7062*	0.5369*	0.4184*	0.5960*	0.5355*	0.5124**	0.3869*	0.3172*	0.4196*	0.3774*
	MICoL (Bi-Encoder, $P \leftarrow (PP) \rightarrow P$)	0.7050*	0.5344*	0.4161*	0.5937*	0.5331*	0.5198**	0.3876*	0.3172*	0.4215*	0.3786*
	MICoL (Cross-Encoder, $P \rightarrow P \leftarrow P$)	**0.7177**	**0.5444**	**0.4219**	**0.6048**	**0.5415**	**0.5412**	**0.4036**	0.3257	**0.4391**	0.3906
	MICoL (Cross-Encoder, $P \leftarrow (PP) \rightarrow P$)	0.7061	0.5376	0.4187	0.5964	0.5357	0.5218	0.3911	0.3172*	0.4249	0.3794
Supervised	MATCH [68] (10K Training)	0.4423**	0.2851**	0.2152**	0.3375**	0.3003**	0.6915	0.3869*	0.2785**	0.4649	0.3896
	MATCH [68] (50K Training)	0.6215**	0.4280**	0.3269**	0.4987**	0.4489**	0.7701	0.4716	0.3585	0.5497	0.4750
	MATCH [68] (100K Training)	0.8321	0.6520	0.5142	0.7342	0.6761	0.8286	0.5680	0.4410	0.6405	0.5626
	MATCH [68] (Full, 560K+ Training)	0.9114	0.7634	0.6312	0.8486	0.8076	0.9151	0.7425	0.6104	0.8001	0.7310

MICoL: Effect of Different Types of Metadata

- All meta-paths and meta-graphs used in MICoL, except Paper-Venue-Paper, can improve the classification performance upon unfine-tuned SciBERT.

Algorithm	MAG-CS [49]					PubMed [24]				
	P@1	P@3	P@5	NDCG@3	NDCG@5	P@1	P@3	P@5	NDCG@3	NDCG@5
Unfine-tuned SciBERT	0.6599**	0.5117**	0.4056**	0.5651**	0.5136**	0.4371**	0.3544**	0.3014**	0.3775**	0.3485**
MICoL (Bi-Encoder, PAP)	0.6877**	0.5285**	0.4143**	0.5852**	0.5280**	0.4974**	0.3818**	0.3154*	0.4122**	0.3727**
MICoL (Bi-Encoder, PVP)	0.6589**	0.5123**	0.4063**	0.5656**	0.5145**	0.4440**	0.3507**	0.2966**	0.3761**	0.3458**
MICoL (Bi-Encoder, $P \rightarrow P$)	0.7094	0.5391	0.4190	0.5982	0.5367	0.5200*	0.3903*	0.3195	0.4240*	0.3808*
MICoL (Bi-Encoder, $P \leftarrow P$)	0.7095*	0.5374*	0.4178*	0.5970*	0.5356*	0.5195**	0.3905*	0.3192	0.4240*	0.3806*
MICoL (Bi-Encoder, $P \rightarrow P \leftarrow P$)	0.7062*	0.5369*	0.4184*	0.5960*	0.5355*	0.5124**	0.3869*	0.3172*	0.4196*	0.3774*
MICoL (Bi-Encoder, $P \leftarrow P \rightarrow P$)	0.7039*	0.5379*	0.4187*	0.5963*	0.5356*	0.5174**	0.3886*	0.3187*	0.4220*	0.3795*
MICoL (Bi-Encoder, $P(AA)P$)	0.6873**	0.5272**	0.4130**	0.5840**	0.5269**	0.4963**	0.3794**	0.3139**	0.4101**	0.3711**
MICoL (Bi-Encoder, $P(AV)P$)	0.6832**	0.5263**	0.4135**	0.5823**	0.5263**	0.4894**	0.3743**	0.3099**	0.4045**	0.3664**
MICoL (Bi-Encoder, $P \rightarrow (PP) \leftarrow P$)	0.7015**	0.5334**	0.4160**	0.5920**	0.5322**	0.5163**	0.3879*	0.3172*	0.4211*	0.3781*
MICoL (Bi-Encoder, $P \leftarrow (PP) \rightarrow P$)	0.7050*	0.5344*	0.4161*	0.5937*	0.5331*	0.5198**	0.3876*	0.3172*	0.4215*	0.3786*
MICoL (Cross-Encoder, PAP)	0.7034*	0.5355	0.4168	0.5943	0.5337	0.5212**	0.3921*	0.3207	0.4255*	0.3818*
MICoL (Cross-Encoder, PVP)	0.6720*	0.5203*	0.4103*	0.5750*	0.5210*	0.4668**	0.3633**	0.3051**	0.3908**	0.3574**
MICoL (Cross-Encoder, $P \rightarrow P$)	0.7033*	0.5391	0.4201	0.5971*	0.5365*	0.5266	0.3946	0.3207	0.4286	0.3830
MICoL (Cross-Encoder, $P \leftarrow P$)	0.7169	0.5430	0.4214	0.6033	0.5406	0.5265	0.3924	0.3186	0.4268	0.3811
MICoL (Cross-Encoder, $P \rightarrow P \leftarrow P$)	**0.7177**	**0.5444**	**0.4219**	**0.6048**	**0.5415**	**0.5412**	**0.4036**	**0.3257**	**0.4391**	**0.3906**
MICoL (Cross-Encoder, $P \leftarrow P \rightarrow P$)	0.7045	0.5356*	0.4168*	0.5944*	0.5336*	0.5243*	0.3932*	0.3190*	0.4271*	0.3814*
MICoL (Cross-Encoder, $P(AA)P$)	0.7028	0.5351	0.4171	0.5939	0.5338	0.5290*	0.3937	0.3201	0.4285*	0.3830
MICoL (Cross-Encoder, $P(AV)P$)	0.7024*	0.5354*	0.4177	0.5940*	0.5343*	0.5164**	0.3897*	0.3195*	0.4225*	0.3797*
MICoL (Cross-Encoder, $P \rightarrow (PP) \leftarrow P$)	0.7076*	0.5379*	0.4188	0.5971*	0.5363*	0.5186	0.3924*	0.3184*	0.4254*	0.3800*
MICoL (Cross-Encoder, $P \leftarrow (PP) \rightarrow P$)	0.7061	0.5376	0.4187	0.5964	0.5357	0.5218	0.3911	0.3172*	0.4249	0.3794

MAPLE: A Cross-Field Cross-Model Study

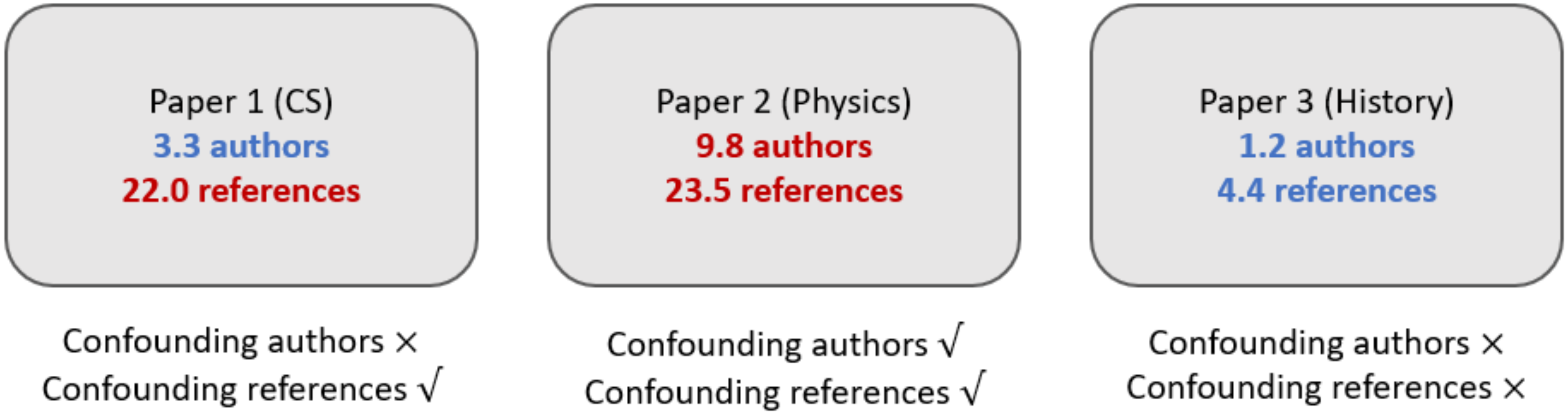
- **Q1: Are metadata always helpful across all scientific fields?**
  - The focus of previous studies is restricted to one or two scientific fields only (e.g., **computer science** and **biomedicine**).
  - The effect of metadata in other fields (e.g., **art, economics, mathematics, physics**) has not been systematically examined.

Paper 1 (CS)	Paper 2 (Chemistry)	Paper 3 (History)
Venue: **KDD**	Venue: **Nature**	Venue: **Journal of Roman Archaeology**
Topic: **Data Mining**, **Graph Mining**, …	Topic: **Organic Chemistry**, **Suzuki Coupling**, …	Topic: **Archaeology**, **Roman Archaeology**, …
Coarse-grained √	Coarse-grained ×	Coarse-grained √
Fine-grained ×	Fine-grained ×	Fine-grained √

Zhang, Y., Jin, B., Zhu, Q., Meng, Y., & Han, J. "The Effect of Metadata on Scientific Literature Tagging: A Cross-Field Cross-Model Study", WWW'23.

MAPLE: A Cross-Field Cross-Model Study

- **Q1: Are metadata always helpful across all scientific fields?**
  - The focus of previous studies is restricted to one or two scientific fields only (e.g., **computer science** and **biomedicine**).
  - The effect of metadata in other fields (e.g., **art, economics, mathematics, physics**) has not been systematically examined.



MAPLE: Constructing a Cross-Field Benchmark

- We construct a large-scale scientific literature tagging benchmark, **MAPLE**, from the Microsoft Academic Graph.
- MAPLE covers **19 scientific fields** and consists of more than **11.9 million papers**.
- The number of candidate tags in each field ranges between **~700** and **~64,000**.

- https://doi.org/10.5281/zenodo.7611544

Table 1: Statistics of the 20 datasets in MAPLE across 19 fields. There are 2 datasets in the Computer Science field, one of which is collected from top conferences and the other from top journals.

Field	Paper Source	#Papers	#Labels	#Venues	#Authors	#References
Art	Journal	58,373	1,990	98	54,802	115,343
Philosophy	Journal	59,296	3,758	98	36,619	198,010
Geography	Journal	73,883	3,285	98	157,423	884,632
Business	Journal	84,858	2,392	97	100,525	685,034
Sociology	Journal	90,208	1,935	98	85,793	842,561
History	Journal	113,147	2,689	99	84,529	284,739
Political Science	Journal	115,291	4,990	98	93,393	480,136
Environmental Science	Journal	123,945	694	100	265,728	1,217,268
Economics	Journal	178,670	5,205	97	135,247	1,042,253
Engineering	Journal	270,006	10,683	100	430,046	1,867,276
Psychology	Journal	372,954	7,641	100	460,123	2,313,701
Computer Science	Conference	263,393	13,613	75	331,582	1,084,440
	Journal	410,603	15,540	96	634,506	2,751,996
Geology	Journal	431,834	7,883	100	471,216	1,753,762
Mathematics	Journal	490,551	14,271	98	404,066	2,150,584
Materials Science	Journal	1,337,731	6,802	99	1,904,549	5,457,773
Physics	Journal	1,369,983	16,664	91	1,392,070	3,641,761
Biology	Journal	1,588,778	64,267	100	2,730,547	7,086,131
Chemistry	Journal	1,849,956	35,538	100	2,721,253	8,637,438
Medicine	Journal	2,646,105	36,619	100	4,345,385	7,405,779

MAPLE: A Cross-Field Cross-Model Study

- **Q2: Are metadata always helpful across all classifiers?**
  - **Bag-of-words**: Parabel [1]
  - **Sequenced-based**: Transformer [2]
  - **Pretrained language model**: OAG-BERT [3]
- In the **19 fields**, using the **3 classifiers**, we empirically study if adding metadata (i.e., **venues, authors, and references**) can be helpful.
- Key observations:
  - Venues are consistently beneficial in almost all 19×3 cases; authors in fewer cases; references in even fewer.
  - In some fields (not CS), venues can even benefit the prediction of fine-grained labels.
  - The effect of metadata varies remarkably across different fields and models.

[1] Prabhu et al. “Parabel: Partitioned label trees for extreme classification with application to dynamic search advertising”, WWW’18.
[2] Xun et al. “Correlation networks for extreme multi-label text classification”, KDD’20.
[3] Liu et al. “OAG-BERT: Towards a Unified Backbone Language Model for Academic Knowledge Services”, KDD’22.

MAPLE: A Cross-Field Cross-Model Study

- **Q3: Shall I use a certain type of metadata in a field for a classifier?**
- The effect of metadata tends to be similar in two fields that **belong to the same high-level scientific area** [1]. For example, **Biology** and **Medicine** are both **life sciences**, and the effects of venues, authors, and references are largely aligned in the two fields.
- The experience of using metadata in one field can be extrapolated to a similar field!

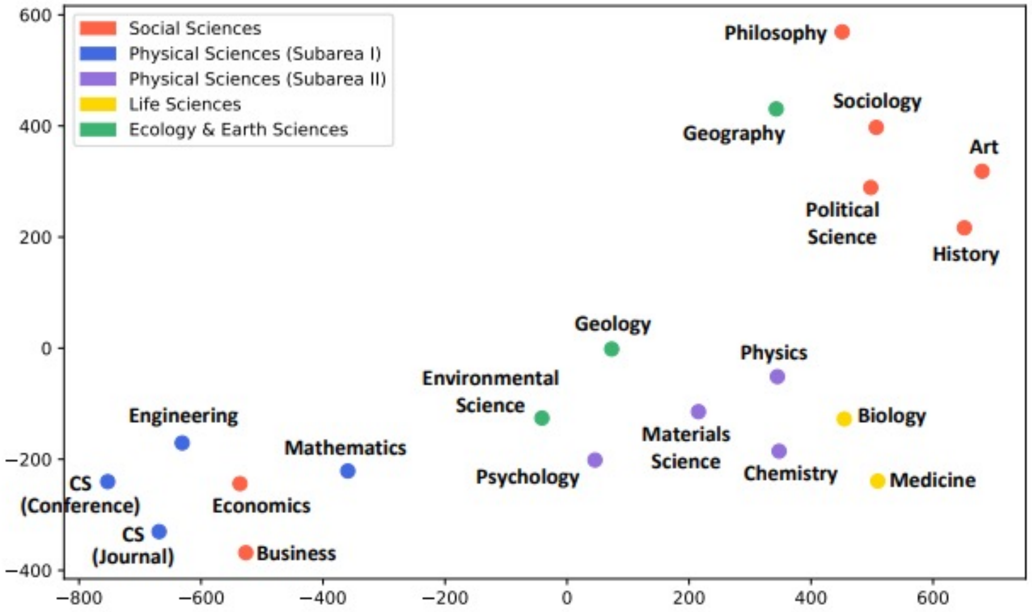


Figure 3: We represent each field with a 24-dimensional vector based on the effect of venue, author, and reference information on the three classifiers. Then, we apply t-SNE [27] to visualize these fields in a 2-dimensional space. The color scheme highlights several high-level scientific areas, following the major clusters of science detected by [35, 51] and suggesting similar effects of metadata within each area.

[1] Yin et al., "Public use and public funding of science". Nature Human Behaviour 2022.

Outline

- Why do we care weakly-supervised text classification/NLU?
- Weakly-supervised text classification
  - ConWea [ACL'20], LOTClass [EMNLP'20], X-Class [NAACL'21], PromptClass [arXiv'23]
- Weakly-supervised structure-enhanced text classification
  - Taxonomy-enhanced: TaxoClass [NAACL'21]
  - Metadata-enhanced: MICoL [WWW'22], MAPLE [WWW'23]
- Weakly-supervised NLU
  - Zero-shot: SuperGen [NeurIPS'22], ZeroGen [EMNLP'22]
  - Few-shot: FewGen [ICML'23]

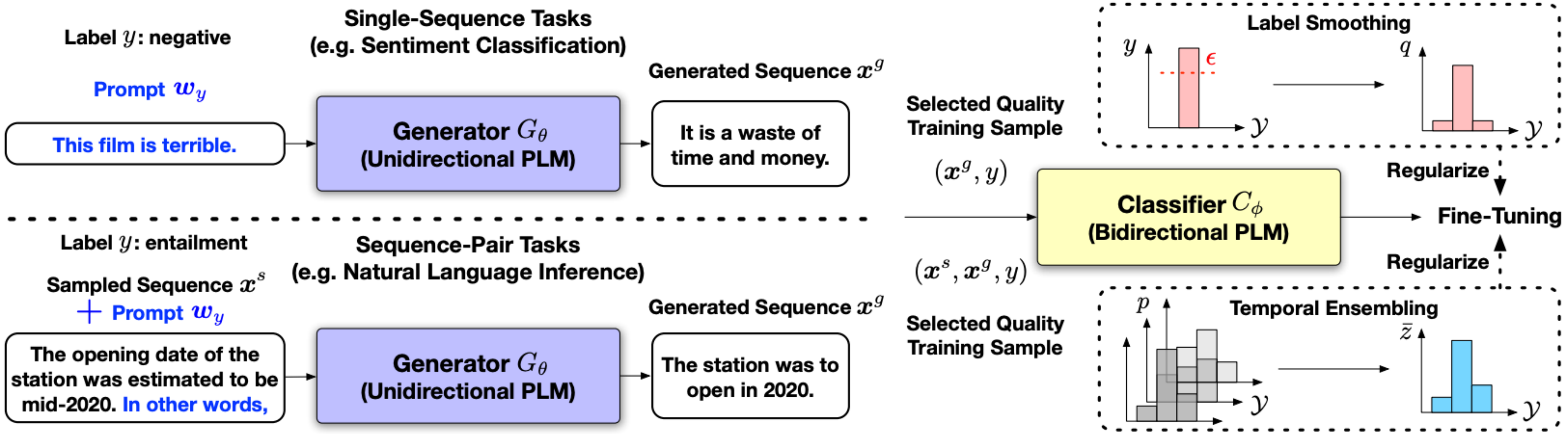
Zero-Shot Fine-Tuning of PLMs for NLU

- How can PLMs perform zero-shot NLU?
  - (Text Input, Prompt) -> Label
- Without any task-specific samples, it is challenging for PLMs to interpret the prompts that come in different formats and are unseen in the pretraining data.
- When there are no training data, we can create them from scratch using PLMs!
  - (Prompt, Label) -> Text Input
  - Generate pseudo training data pertaining to a specific label upon given a label-descriptive prompt (e.g., “write a negative review:”)

Task	Label	Prompt
SST-2	positive negative	Rating: 5.0 $\boldsymbol{x}^g$ Rating: 1.0 $\boldsymbol{x}^g$
MNLI	entailment neutral contradiction	$\boldsymbol{x}^s$. In other words, $\boldsymbol{x}^g$ $\boldsymbol{x}^s$. Furthermore, $\boldsymbol{x}^g$ There is a rumor that $\boldsymbol{x}^s$. However, the truth is: $\boldsymbol{x}^g$
QNLI	entailment not entailment	$\boldsymbol{x}^s$? $\boldsymbol{x}^g$ $\boldsymbol{x}^s$? ... $\boldsymbol{x}^g$
RTE	entailment not entailment	$\boldsymbol{x}^s$. In other words, $\boldsymbol{x}^g$ $\boldsymbol{x}^s$. Furthermore, $\boldsymbol{x}^g$
MRPC	equivalent not equivalent	$\boldsymbol{x}^s$. In other words, $\boldsymbol{x}^g$ $\boldsymbol{x}^s$. Furthermore, $\boldsymbol{x}^g$
QQP	equivalent not equivalent	$\boldsymbol{x}^s$? In other words, $\boldsymbol{x}^g$ $\boldsymbol{x}^s$? Furthermore, $\boldsymbol{x}^g$

SuperGen: Prompt-Based Zero-Shot Training Data Generation

- SuperGen: A **Super**vision **Gen**eration approach
- Use a unidirectional PLM (e.g., CTRL) to generate class-conditioned texts guided by prompts
- Fine-tune a bidirectional PLM (e.g., COCO-LM) on the generated data for the corresponding task



Meng, Y., Huang, J., Zhang, Y., & Han, J. "Generating Training Data with Language Models: Towards Zero-Shot Language Understanding", NeurIPS'22.

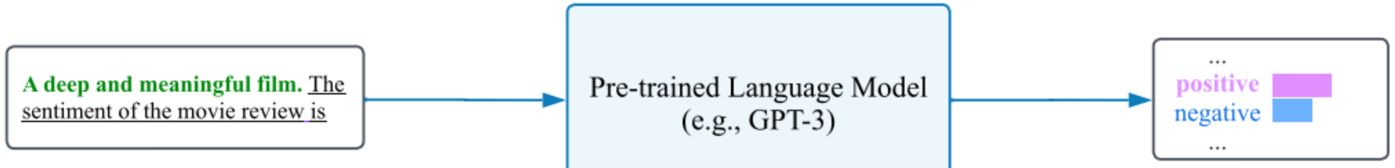
SuperGen: Experiment Results

- Using the same prompt-based fine-tuning method, zero-shot SuperGen (fine-tuned on generated training data) is comparable or even better than strong few-shot methods (fine-tuned on 32 manually annotated training samples per class)

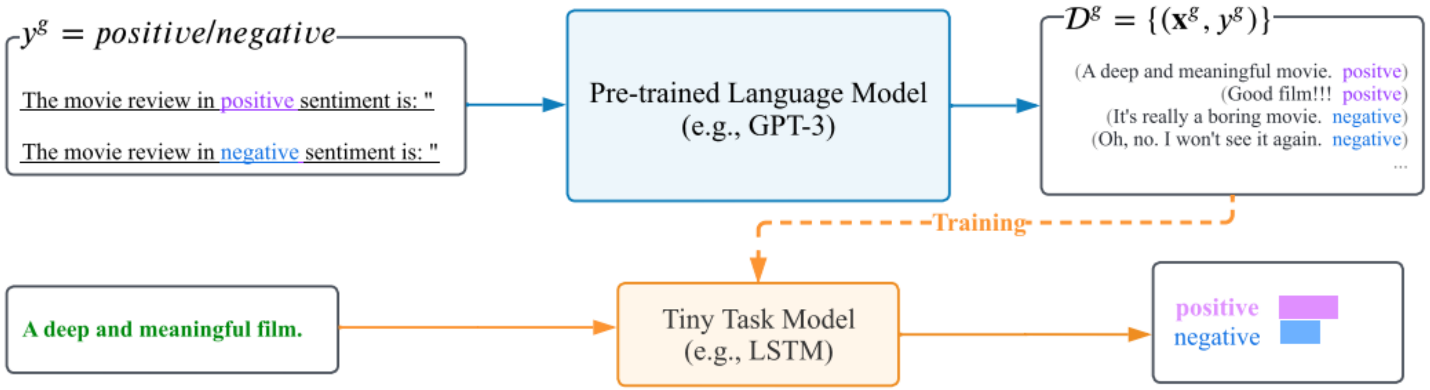
Method	**MNLI-(m/mm)** (Acc.)	**QQP** (F1)	**QNLI** (Acc.)	**SST-2** (Acc.)	**CoLA** (Matt.)	**RTE** (Acc.)	**MRPC** (F1)	**AVG**
Zero-Shot Setting: No task-specific data (neither labeled nor unlabeled).								
Prompting†	$50.8_{0.0}$/$51.7_{0.0}$	$49.7_{0.0}$	$50.8_{0.0}$	$83.6_{0.0}$	$2.0_{0.0}$	$51.3_{0.0}$	$61.9_{0.0}$	50.1
SuperGen	$\mathbf{72.3}_{0.5}$/$\mathbf{73.8}_{0.5}$	$\mathbf{66.1}_{1.1}$	$\mathbf{73.3}_{1.9}$	$\mathbf{92.8}_{0.6}$	$\mathbf{32.7}_{5.5}$	$\mathbf{65.3}_{1.2}$	$82.2_{0.5}$	**69.4**
- data selection	$63.7_{1.5}$/$64.2_{1.6}$	$62.3_{2.2}$	$63.9_{3.2}$	$91.3_{2.0}$	$30.5_{8.8}$	$62.4_{1.5}$	$81.6_{0.2}$	65.1
- label smooth	$70.7_{0.8}$/$72.1_{0.7}$	$65.1_{0.9}$	$71.4_{2.5}$	$91.0_{0.9}$	$9.5_{1.0}$	$64.8_{1.1}$	$\mathbf{83.0}_{0.7}$	65.2
- temporal ensemble	$62.0_{4.6}$/$63.6_{4.8}$	$63.9_{0.3}$	$72.4_{2.0}$	$92.5_{0.9}$	$23.5_{7.0}$	$63.5_{1.0}$	$78.8_{2.2}$	65.3
Few-Shot Setting: Use 32 labeled samples/class (half for training and half for development).								
Fine-tuning†	$45.8_{6.4}$/$47.8_{6.8}$	$60.7_{4.3}$	$60.2_{6.5}$	$81.4_{3.8}$	$\mathbf{33.9}_{14.3}$	$54.4_{3.9}$	$76.6_{2.5}$	59.1
Manual prompt†	$68.3_{2.3}$/$70.5_{1.9}$	$65.5_{5.3}$	$64.5_{4.2}$	$92.7_{0.9}$	$9.3_{7.3}$	$69.1_{3.6}$	$74.5_{5.3}$	63.6
+ demonstration†	$\mathbf{70.7}_{1.3}$/$\mathbf{72.0}_{1.2}$	$\mathbf{69.8}_{1.8}$	$\mathbf{69.2}_{1.9}$	$92.6_{0.5}$	$18.7_{8.8}$	$68.7_{2.3}$	$77.8_{2.0}$	66.9
Auto prompt†	$68.3_{2.5}$/$70.1_{2.6}$	$67.0_{3.0}$	$68.3_{7.4}$	$92.3_{1.0}$	$14.0_{14.1}$	$\mathbf{73.9}_{2.2}$	$76.2_{2.3}$	65.8
+ demonstration†	$70.0_{3.6}$/$72.0_{3.1}$	$67.7_{5.8}$	$68.5_{5.4}$	$\mathbf{93.0}_{0.6}$	$21.8_{15.9}$	$71.1_{5.3}$	$\mathbf{78.1}_{3.4}$	**67.3**

ZeroGen: Efficient Zero-shot Learning via Dataset Generation

- In comparison with SuperGen:
  - A similar-size generator (e.g., GPT-2 XL) and a smaller classifier (e.g., LSTM)
  - More tasks (e.g., Question Answering)



(a) Prompt-based Zero-shot Learning



(b) Efficient Zero-shot Learning via Dataset Generation

Ye, J., Gao, J., Li, Q., Xu, H., Feng, J., Wu, Z., Yu, T., & Kong, L. "ZeroGen: Efficient Zero-shot Learning via Dataset Generation", EMNLP'22.

ZeroGen: Experiment Results

- On RTE, ZeroGen already outperforms the supervised model.

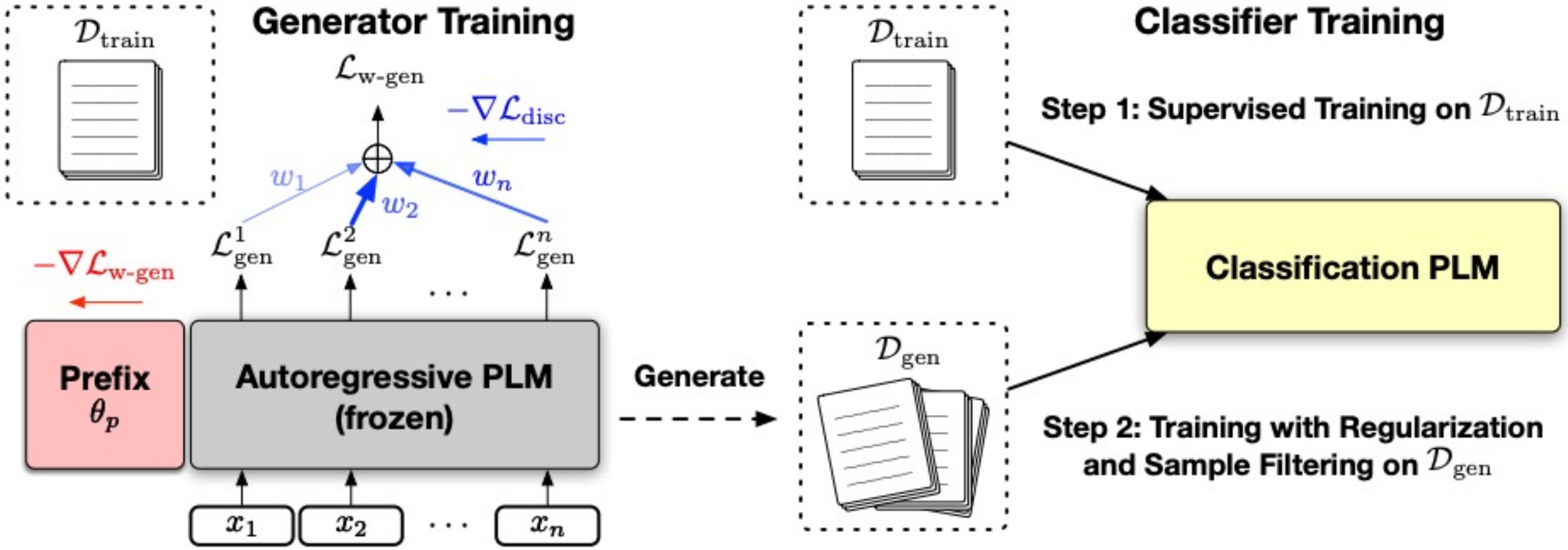
PLM	TAM	#Param	Setting	IMDb	SST-2	SQuAD	AdversarialQA	QNLI	RTE
#Gold Data				25k	6.7k	87k	30k	105k	2.5k
-	DistilBERT	66M	SUPERVISED	87.24	89.68	76.28/84.67	18.6/29.85	88.05	58.12
	LSTM	~7M		84.60	76.30	41.86/57.22	5.37/11.86	69.00	54.87
GPT2	-	117M	PROMPTING	51.52	52.52	0.80/4.93	0.37/2.58	50.60	**52.70**
	DistilBERT	66M	ZEROGEN	**73.24**	**80.39**	**16.44/21.83**	**5.20/8.26**	**55.32**	50.54
	LSTM	~7M		69.60	70.40	4.94/8.53	1.00/3.83	51.03	49.10
GPT2-Large	-	762M	PROMPTING	80.20	**87.84**	3.53/10.78	1.47/5.16	55.10	54.51
	DistilBERT	66M	ZEROGEN	**83.56**	85.44	**23.87/29.82**	**5.93/9.63**	**69.32**	**58.48***
	LSTM	~7M		78.20	75.10	8.01/12.77	2.33/5.24	51.27	56.68*
GPT2-XL	-	1.5B	PROMPTING	80.64	**89.22**	4.61/13.32	2.13/6.30	60.60	57.04
	DistilBERT	66M	ZEROGEN	**84.28**	87.27	**25.50/31.53**	**6.33/9.96**	**71.19**	**59.93***
	LSTM	~7M		79.80	78.40*	12.35/18.66	3.23/6.34	52.26	58.85*

Outline

- Why do we care weakly-supervised text classification/NLU?
- Weakly-supervised text classification
  - ConWea [ACL’20], LOTClass [EMNLP’20], X-Class [NAACL’21], PromptClass [arXiv’23]
- Weakly-supervised structure-enhanced text classification
  - Taxonomy-enhanced: TaxoClass [NAACL’21]
  - Metadata-enhanced: MICoL [WWW’22], MAPLE [WWW’23]
- Weakly-supervised NLU
  - Zero-shot: SuperGen [NeurIPS’22], ZeroGen [EMNLP’22]
  - Few-shot: FewGen [ICML’23]

FewGen: Augmentation-Enhanced Few-Shot Learning

- Tune a generative PLM (GPT-like) on the small few-shot training set using prefix-tuning
- Use the tuned PLM to create novel training data
- Fine-tune a classification PLM on both the few-shot and synthetic training sets



Meng, Y., Michalski, M., Huang, J., Zhang, Y., Abdelzaher, T., & Han, J. “Tuning Language Models as Training Data Generators for Augmentation-Enhanced Few-Shot Learning”, ICML’23.

FewGen: Emphasizing Label Distinction in Generator Tuning

- How to emphasize label discriminativeness for generator tuning?
- Weighted generator tuning objective:

$$\min_{\boldsymbol{\theta}_{p_l}} \mathcal{L}_{\text{w-gen}}, \quad \mathcal{L}_{\text{w-gen}}(\boldsymbol{\theta}_{p_l}; \boldsymbol{w}) = -\sum_{j=1}^{n} \underbrace{w_j}_{\text{Token weights}} \mathcal{L}_{\text{gen}}^{j}(\boldsymbol{\theta}_{p_l}), \quad \underbrace{\mathcal{L}_{\text{gen}}^{j}(\boldsymbol{\theta}_{p_l}) = \log p_{\boldsymbol{\theta}_{p_l}}(x_j | \boldsymbol{x}_{<j})}_{\text{Generator loss on each token}}.$$

- Intuitively, important and label-distinctive tokens should be assigned higher weights (e.g., in sentiment classification, one would expect “good/bad” to be more label-discriminative than “the movie”).
- How to set token weights?
  - Manually designing weighting rules likely requires task-specific knowledge and nontrivial tuning

FewGen: Automatically Learning Token Weights via Meta-Learning

- How to automatically learn token weights?

$$\min_{\boldsymbol{\theta}_{p_l}} \mathcal{L}_{\text{w-gen}}, \quad \mathcal{L}_{\text{w-gen}}(\boldsymbol{\theta}_{p_l}; \boldsymbol{w}) = -\sum_{j=1}^{n} w_j \mathcal{L}_{\text{gen}}^{j}(\boldsymbol{\theta}_{p_l}), \quad \mathcal{L}_{\text{gen}}^{j}(\boldsymbol{\theta}_{p_l}) = \log p_{\boldsymbol{\theta}_{p_l}}(x_j|\boldsymbol{x}_{<j}).$$

Generator loss on each token

Parameterize as learnable hyperparameters

- Formulate a bi-level optimization problem using the idea of meta-learning

$$\boldsymbol{\theta}_p^*(\boldsymbol{\omega}) = \underset{\boldsymbol{\theta}_p}{\operatorname{argmin}}\, \mathcal{L}_{\text{w-gen}}, \quad \mathcal{L}_{\text{w-gen}}(\boldsymbol{\theta}_p; \boldsymbol{\omega}) = -\sum_{j=1}^{n} w_j(\boldsymbol{\omega}) \mathcal{L}_{\text{gen}}^{j}(\boldsymbol{\theta}_p) \longrightarrow$$

Generator tuned under token weights

$$\boldsymbol{\omega}^* = \underset{\boldsymbol{\omega}}{\operatorname{argmin}}\, \mathcal{L}_{\text{disc}}, \quad \mathcal{L}_{\text{disc}}(\boldsymbol{\theta}_p^*(\boldsymbol{\omega})) = -\frac{1}{n}\sum_{j=1}^{n} \mathcal{L}_{\text{disc}}^{j}(\boldsymbol{\theta}_p^*(\boldsymbol{\omega})) \longrightarrow$$

Token weights automatically learned to emphasize label discriminativeness

FewGen: Experiment Results

- 5+ average points higher than the best few-shot baseline without augmentation
- 3+ average points higher than the best augmentation baseline (GPT3Mix)

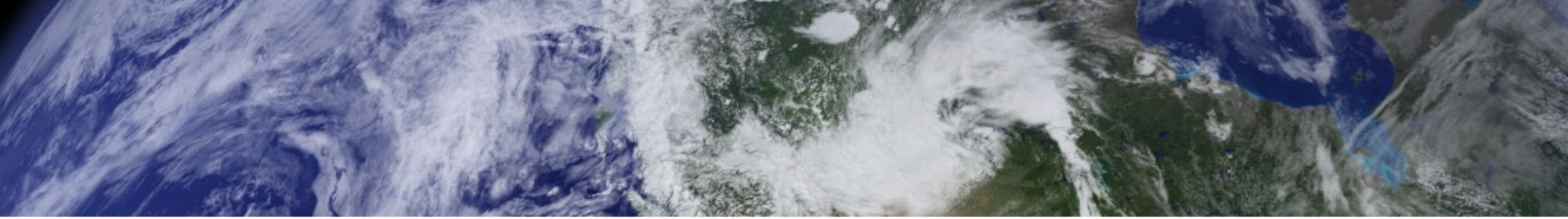
Method	**MNLI-(m/mm)** (Acc.)	**QQP** (F1)	**QNLI** (Acc.)	**SST-2** (Acc.)	**CoLA** (Matt.)	**RTE** (Acc.)	**MRPC** (F1)	**AVG**
Methods without Augmentation: Few-shot samples are directly used for classifier tuning or as demonstrations for inference								
Prompting[†]	50.8/51.7	49.7	50.8	83.6	2.0	51.3	61.9	50.1
Fine-Tuning[†]	45.8$_{6.4}$/47.8$_{6.8}$	60.7$_{4.3}$	60.2$_{6.5}$	81.4$_{3.8}$	33.9$_{14.3}$	54.4$_{3.9}$	76.6$_{2.5}$	59.1
In-Context[†]	52.0$_{0.7}$/53.4$_{0.6}$	36.1$_{5.2}$	53.8$_{0.4}$	84.8$_{1.3}$	−1.5$_{2.4}$	60.4$_{1.4}$	45.7$_{6.0}$	47.4
LM-BFF (Man.)[†]	68.3$_{2.3}$/70.5$_{1.9}$	65.5$_{5.3}$	64.5$_{4.2}$	92.7$_{0.9}$	9.3$_{7.3}$	69.1$_{3.6}$	74.5$_{5.3}$	63.6
+ demonstration[†]	70.7$_{1.3}$/72.0$_{1.2}$	69.8$_{1.8}$	69.2$_{1.9}$	92.6$_{0.5}$	18.7$_{8.8}$	68.7$_{2.3}$	77.8$_{2.0}$	66.9
LM-BFF (Auto)[†] (w. 2.9B T5)	68.3$_{2.5}$/70.1$_{2.6}$	67.0$_{3.0}$	68.3$_{7.4}$	92.3$_{1.0}$	14.0$_{14.1}$	**73.9**$_{2.2}$	76.2$_{2.3}$	65.8
+ demonstration[†] (w. 2.9B T5)	70.0$_{3.6}$/72.0$_{3.1}$	67.7$_{5.8}$	68.5$_{5.4}$	93.0$_{0.6}$	21.8$_{15.9}$	71.1$_{5.3}$	78.1$_{3.4}$	67.3
P-Tuning[‡]	61.5$_{2.1}$/–	65.6$_{3.0}$	64.3$_{2.8}$	92.2$_{0.4}$	–	–	74.5$_{7.6}$	–
DART[‡]	67.5$_{2.6}$/–	67.8$_{3.2}$	66.7$_{3.7}$	93.5$_{0.5}$	–	–	78.3$_{4.5}$	–
Methods with Augmentation: Few-shot samples are used for creating synthesized samples and for classifier tuning								
MixText	65.1$_{2.6}$/66.2$_{2.8}$	60.6$_{3.9}$	68.4$_{5.1}$	89.1$_{2.3}$	12.8$_{9.2}$	66.5$_{4.1}$	64.6$_{7.6}$	61.1
Back Translation (w. trained Marian)	66.9$_{4.6}$/68.3$_{3.8}$	59.8$_{4.6}$	67.8$_{4.9}$	91.1$_{1.9}$	7.5$_{3.7}$	62.4$_{5.3}$	68.0$_{11.2}$	60.6
GPT3Mix (w. 175B GPT3)	61.5$_{3.2}$/62.6$_{2.2}$	70.4$_{1.9}$	69.2$_{0.3}$	**93.6**$_{0.6}$	**48.9**$_{1.9}$	70.4$_{10.0}$	69.9$_{12.4}$	69.2
Generator Fine-Tuning (w. 1.6B CTRL)	68.9$_{5.1}$/70.8$_{5.3}$	60.4$_{8.7}$	70.9$_{4.1}$	91.2$_{1.2}$	18.8$_{10.0}$	66.1$_{4.4}$	60.8$_{15.4}$	62.6
FewGen (w. 1.6B CTRL)	**75.7**$_{1.6}$/**77.1**$_{1.0}$	**71.5**$_{1.7}$	**76.3**$_{4.4}$	93.1$_{0.8}$	40.0$_{7.5}$	71.2$_{2.4}$	**81.1**$_{2.5}$	**72.8**
Fully Supervised Fine-Tuning[†]	*89.8/89.5*	*81.7*	*93.3*	*95.0*	*62.6*	*80.9*	*91.4*	*84.9*

References

- Meng, Y., Shen, J., Zhang, C., & Han, J. “Weakly-supervised neural text classification”, CIKM’18
- Mekala, D. & Shang, J. “Contextualized Weak Supervision for Text Classification”, ACL’20
- Meng, Y., Zhang, Y., Huang, J., Xiong, C., Ji, H., Zhang, C., & Han, J. “Text Classification Using Label Names Only: A Language Model Self-Training Approach”, EMNLP’20
- Wang, Z., Mekala, D., & Shang, J. “X-Class: Text Classification with Extremely Weak Supervision”, NAACL’21
- Zhang, Y., Jiang, M., Meng, Y., Zhang, Y., & Han, J. “PromptClass: Weakly-Supervised Text Classification with Prompting Enhanced Noise-Robust Self-Training”, arXiv’23
- Shen, J., Qiu, W., Meng, Y., Shang, J., Ren, X., & Han, J., “TaxoClass: Hierarchical Multi-Label Text Classification Using Only Class Names”, NAACL’21
- Zhang, Y., Shen, Z., Wu, C., Xie, B., Wang, Y., Wang, K., & Han, J. "Metadata-Induced Contrastive Learning for Zero-Shot Multi-Label Text Classification", WWW’22
- Zhang, Y., Jin, B., Zhu, Q., Meng, Y., & Han, J. "The Effect of Metadata on Scientific Literature Tagging: A Cross-Field Cross-Model Study", WWW’23
- Meng, Y., Huang, J., Zhang, Y., & Han, J. “Generating Training Data with Language Models: Towards Zero-Shot Language Understanding”, NeurIPS’22.

References

- Ye, J., Gao, J., Li, Q., Xu, H., Feng, J., Wu, Z., Yu, T., & Kong, L. “ZeroGen: Efficient Zero-shot Learning via Dataset Generation”, EMNLP’22
- Meng, Y., Michalski, M., Huang, J., Zhang, Y., Abdelzaher, T., & Han, J. “Tuning Language Models as Training Data Generators for Augmentation-Enhanced Few-Shot Learning”, ICML’23



Q&A