From Fundamentals to Recent Advances
A Tutorial on Keyphrasification

Part 3.2 Domain Adaptation for Keyphrase Generation

Rui Meng, Debanjan Mahata, Florian Boudin

ECIR 2022

Part III
Advanced Topics on Keyphrasification

Outline of Part III

Section 1 - Keyphrase Generation for IR

Section 2 - Domain Adaptation for Keyphrase Generation

Section 3 - Learning Better Keyphrase Representations

Section 4 - Conclusion and Q&A

ECIR 2022

Section 2

Resource-efficient Domain Adaptation
for Keyphrase Generation

Status Quo of Keyphrase Generation

- Current models use lots of annotated data for training
  - KP20k dataset, 500k CS scientific papers, keyphrases annotated by authors
  - KPtimes dataset, 260k news articles, keyphrases curated by editor

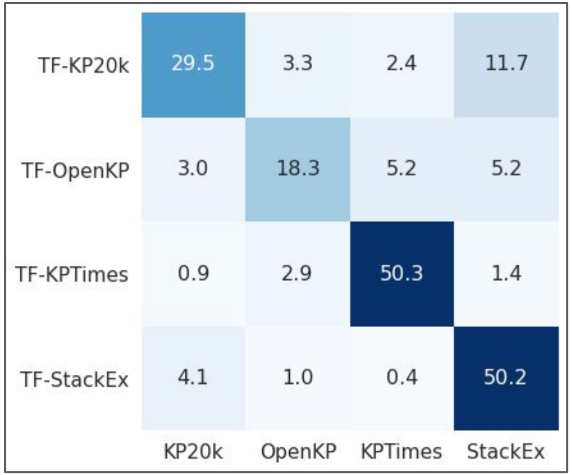
- Are models trained with data of a certain domain (e.g. paper) can be directly transferred to other domains?

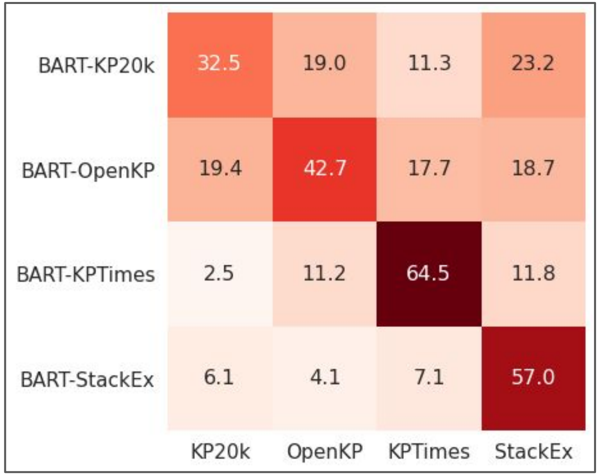
Is KP model transferable across domains?

- Investigate transferability across domains
    - Train a KP generation model in domain A, test it in domain B
        - Transformer (1) 6+6 layers trained from scratch, (2) 12+12 layers initialized from BART
- Four KP datasets in different domains
    - KP20k (CS papers): #doc=514k, absent_kp=36.7%
    - OpenKP (web pages): #doc=135k, absent_kp=2.0%
    - KPtimes (news articles): #doc=260k, absent_kp=52.0%
    - StackExchange (CS Q&A posts): #doc=299k, absent_kp=42.3%

Transferability of KPG models

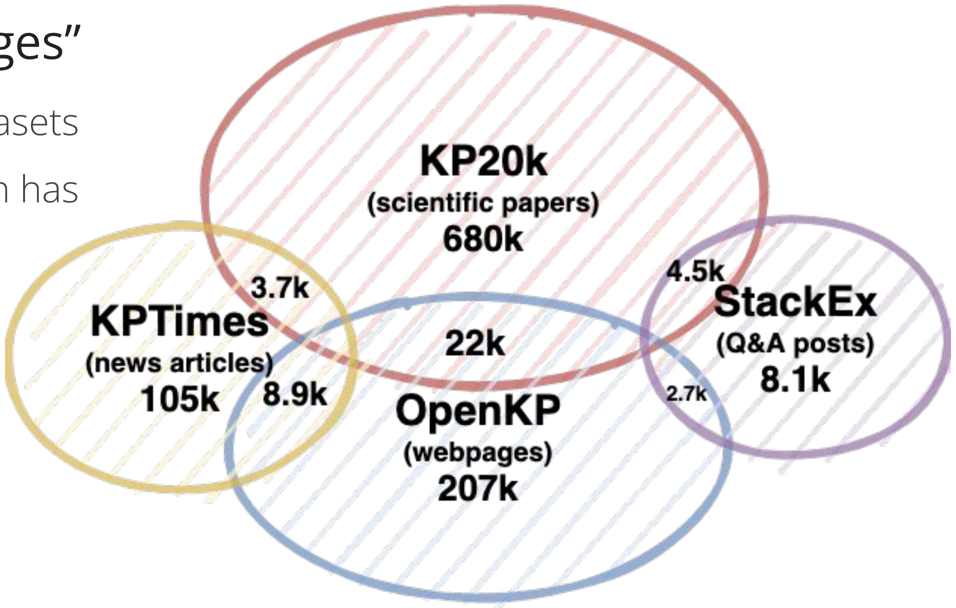
- KPG models do not transfer well across domains
  - Transformers trained from scratch show little transferability
  - Models trained with BART generalize better, but large gaps remain





Transferability of KPG models

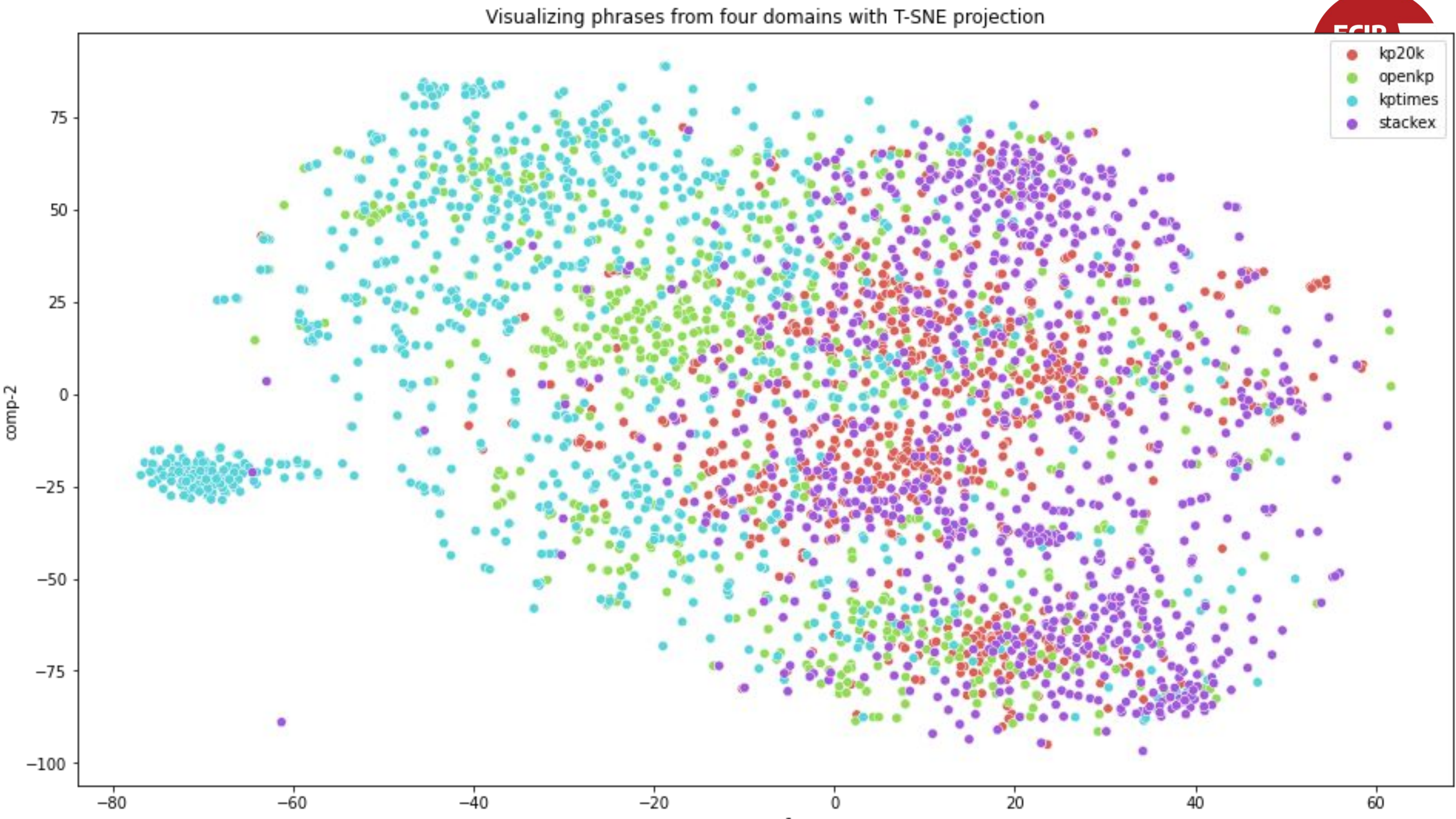
- Datasets speak in different “languages”
  - Small overlap of keyphrases between datasets
  - In the real world, each domain/application has specific keyphrases of interest
    - News -> entities
    - QA forum -> topics/categories

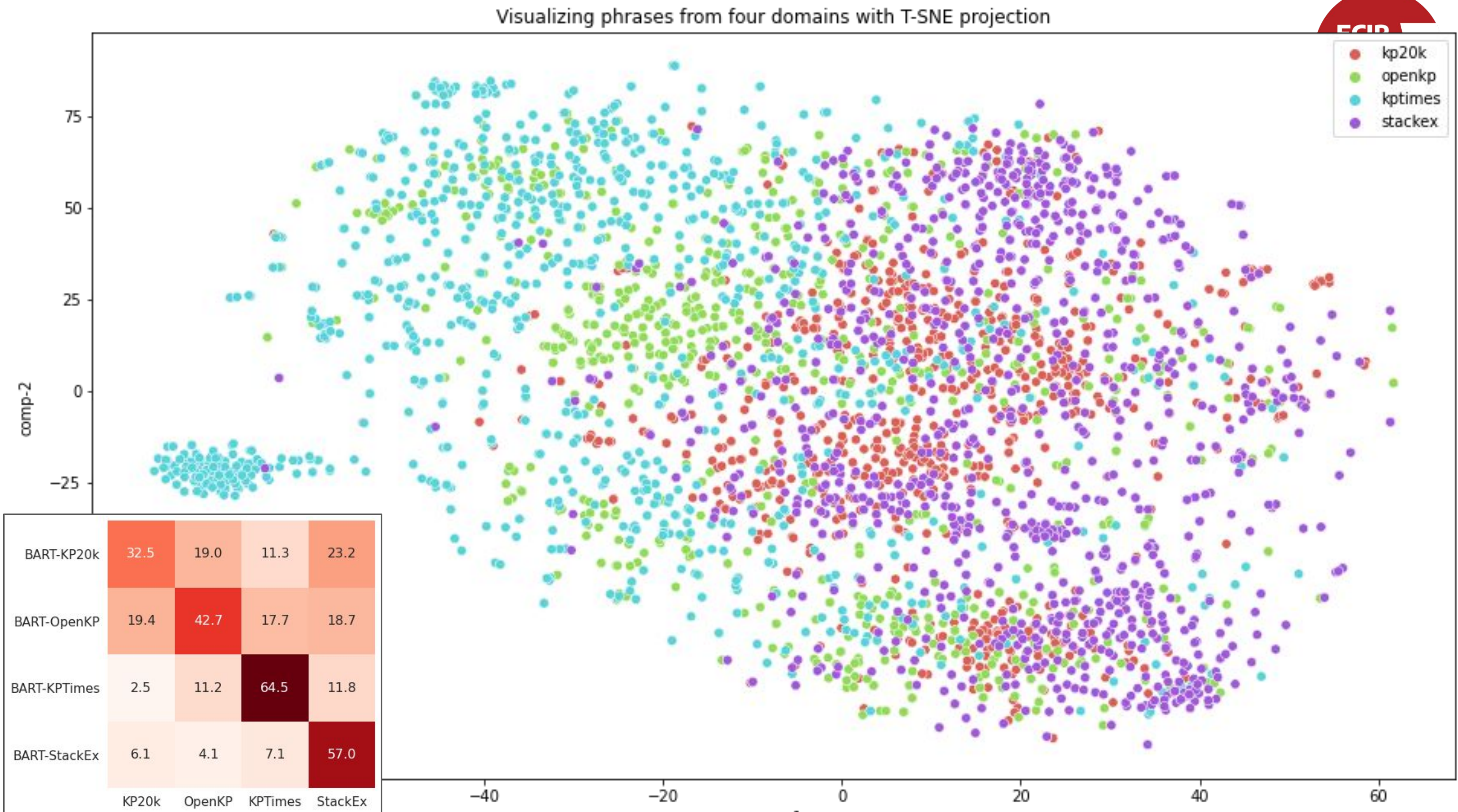


Transferability of KPG models

- Frequent phrases in each domain

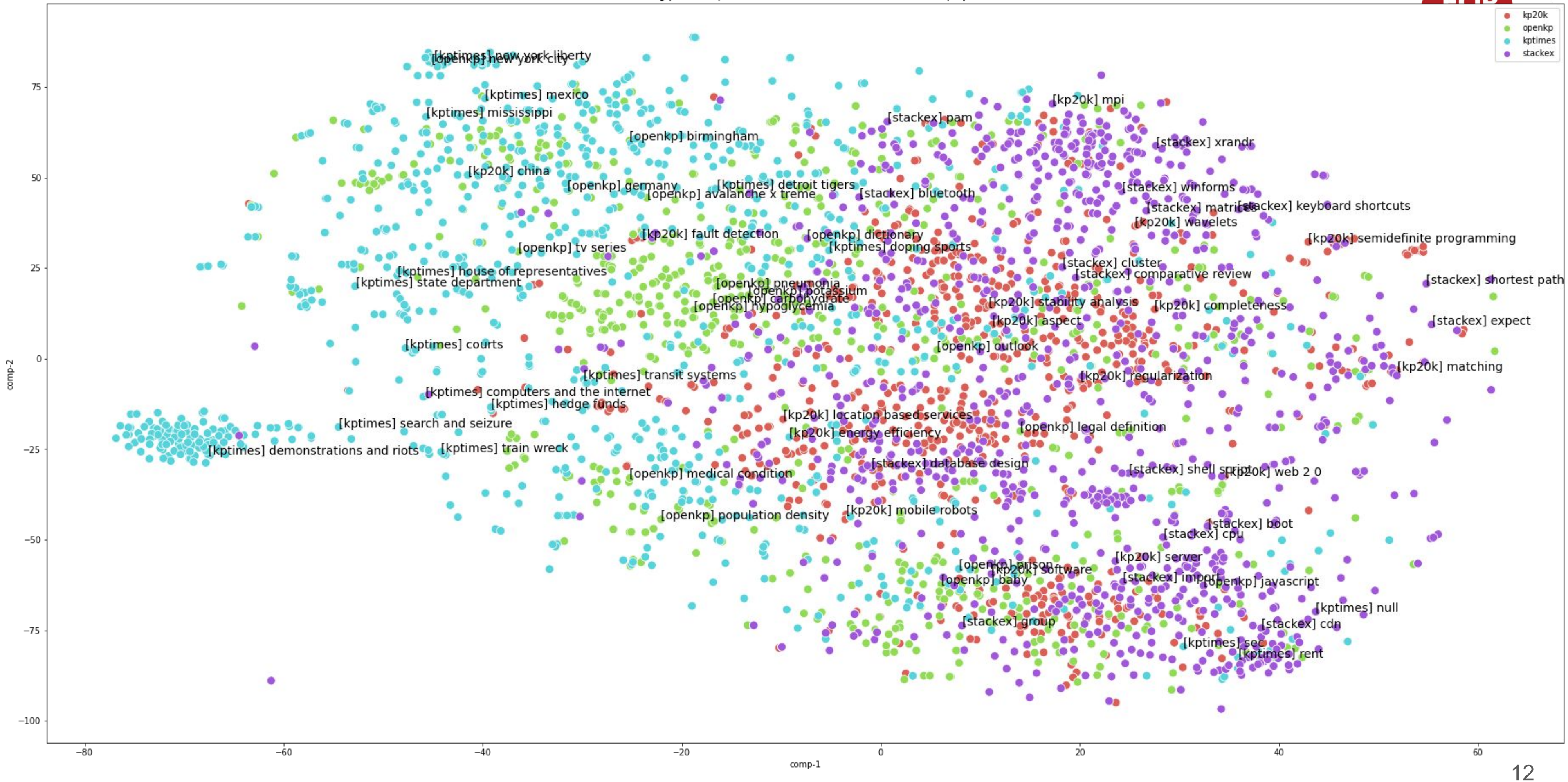
KP20k	OpenKP	KPTimes	StackEx
paper	dictionary	baseball	c
performance	definition	football	linux
design	united states	basketball	bash
simulation	recipe	computers and the internet	java
systems	weather	china	python
algorithms	definitions	nyc	javascript
algorithm	difference	terrorism	shell script
optimization	meaning	politics and government	debian
scheduling	recipes	soccer	algorithms
classification	error	new york city	shell
timing	california	us politics	seo
data mining	symptoms	economic conditions and trends	php
use	calories	barack obama	ubuntu
applications	history	russia	performance
genetic algorithm	quizlet	2016 presidential election	centos
data	new york	obama barack	networking
model	florida	united states politics and government	ssh
neural networks	nutrition facts	tennis	object oriented
computation	texas	golf	text processing
clustering	windows 10	international relations	beginner


Visualizing phrases from four domains with T-SNE projection
kp20k
openkp
kptimes
stackex
comp-2
75
50
25
0
−25
−50
−75
−100
−80
−60
−40
−20
0
20
40
60



	KP20k	OpenKP	KPTimes	StackEx
BART-KP20k	32.5	19.0	11.3	23.2
BART-OpenKP	19.4	42.7	17.7	18.7
BART-KPTimes	2.5	11.2	64.5	11.8
BART-StackEx	6.1	4.1	7.1	57.0

Visualizing phrase representations from four domains with T-SNE projection.



Motivation

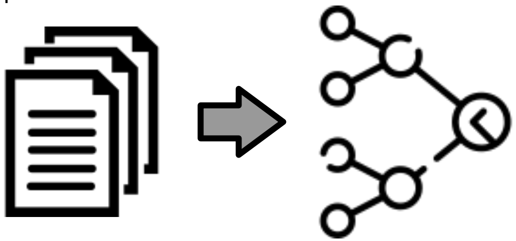
- If one-size-fits-all models don’t exist
  - We may need to build KPG models dedicated to each target domain
  - Ideally we only need limited annotations for each domain

- Structure of language is universal
  - Can we learn domain-general phraseness with distant supervision?
  - Subsequently bootstrap target-domain keyphrase training?

Problem Formulation

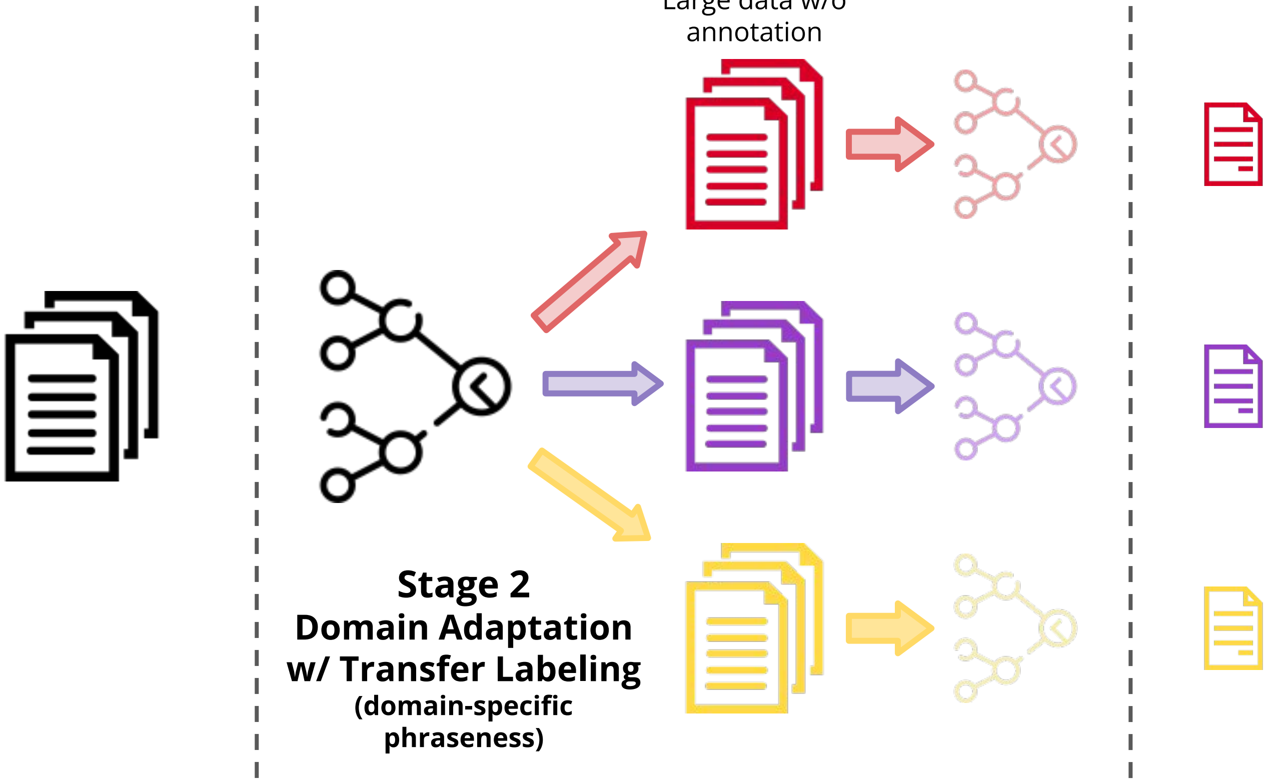
Problem Formulation

General-domain text w/
noisy phrase annotation

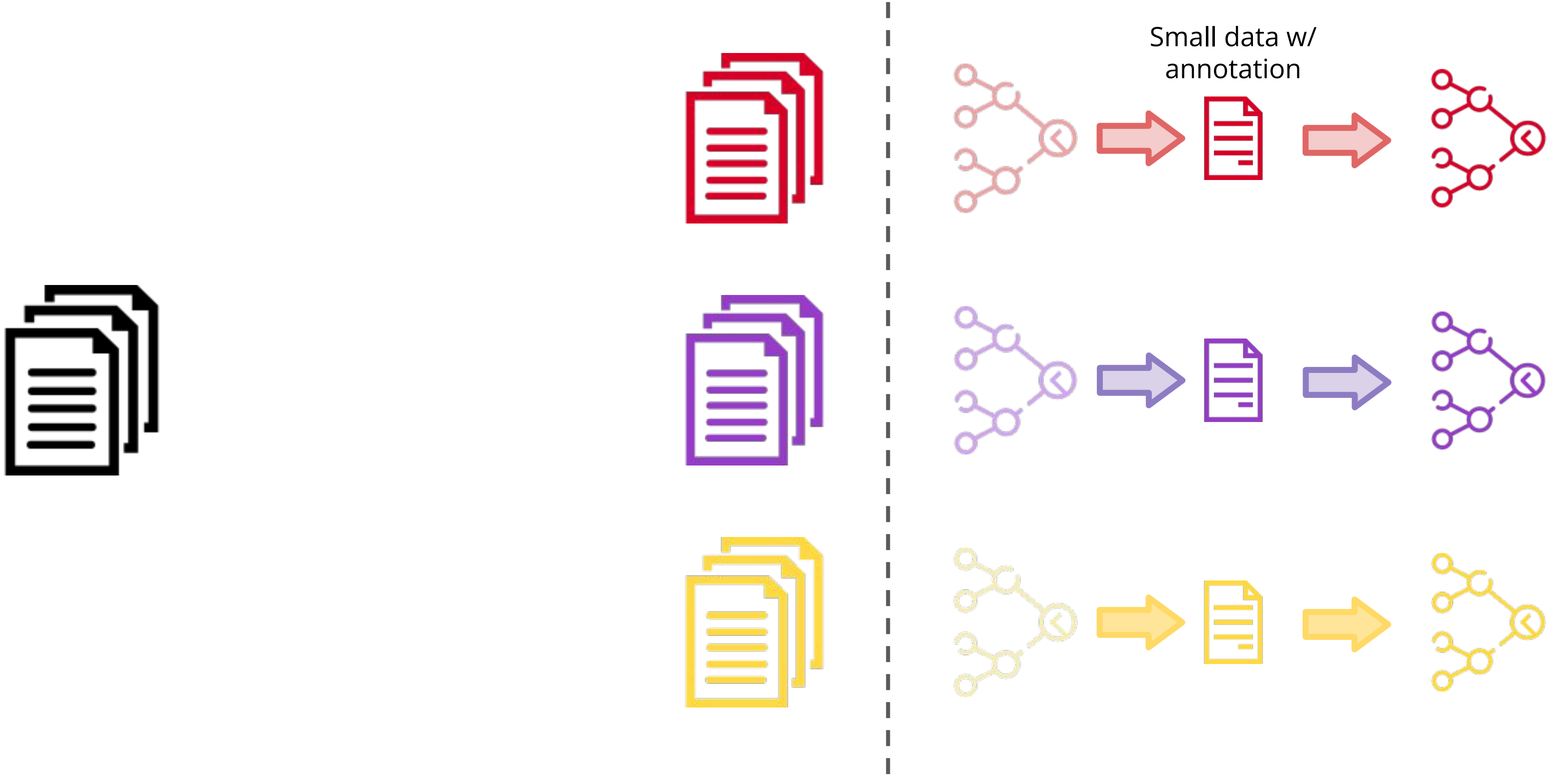


Stage 1
Domain-General
Phrase Pre-Training
(domain-general phraseness)

Problem Formulation



Problem Formulation



Stage 3
Supervised Fine-tuning
(domain-specific keyness)

Our Method

Stage 1. **PT**: Domain-General Phrase-Level Pre-Training

- Utilize "natural" mention annotations in Wikipedia for pre-training KPG models
- Learn knowledge of general-domain phraseness

Stage 2. **DA**: Domain Adaptation with Transfer Labeling

- Derive weak annotation in target domains with pre-trained KPG models
- Acquire domain-specific phrase knowledge

Stage 3. **FT**: Fewshot Supervised Fine-Tuning

- Fine-tune the KPG model with a small amount of annotated keyphrase data in target domain

Stage 1 - Domain-General Phrase-Level Pre-Training

- Wikipedia contains rich annotation of entity mentions and categories
- Covers a wide spectrum of topics
- Train models for general-domain phraseness

Wikipedia Text

Reinforcement learning (**RL**) is an area of machine learning concerned with how software agents ought to take actions in an environment in order to maximize the notion of cumulative reward. Reinforcement learning is one of three basic machine learning paradigms, alongside supervised learning and unsupervised learning

Categories: Reinforcement learning | Markov models | Belief revision

Stage 1 - Domain-General Phrase-Level Pre-Training

ECIR 2022

- Convert each wikipedia text to a src-tgt pair
- Corrupt source sequence by randomly masking phrases or text spans

Original Text

Reinforcement learning (**RL**) is an area of machine learning concerned with how software agents ought to take actions in an environment in order to maximize the notion of cumulative reward. Reinforcement learning is one of three basic machine learning paradigms, alongside supervised learning and unsupervised learning

Categories: Reinforcement learning | Markov models | Belief revision

Source

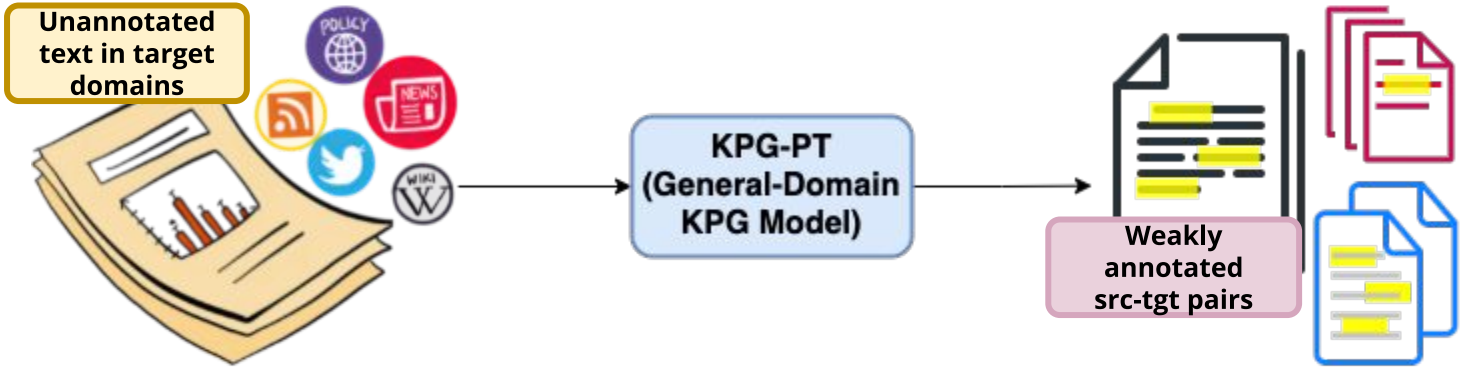
<present>4<category>2<absent>1<sep> **Reinforcement learning (RL)** is an area of **<mask>** concerned with how **<mask>** ought to take **actions** in an environment in order to maximize **<mask>** reward ...

Target

Reinforcement learning (RL) <sep> machine learning <sep> software agents <sep> actions <sep> Markov models <sep> Belief revision <sep> the notion of cumulative

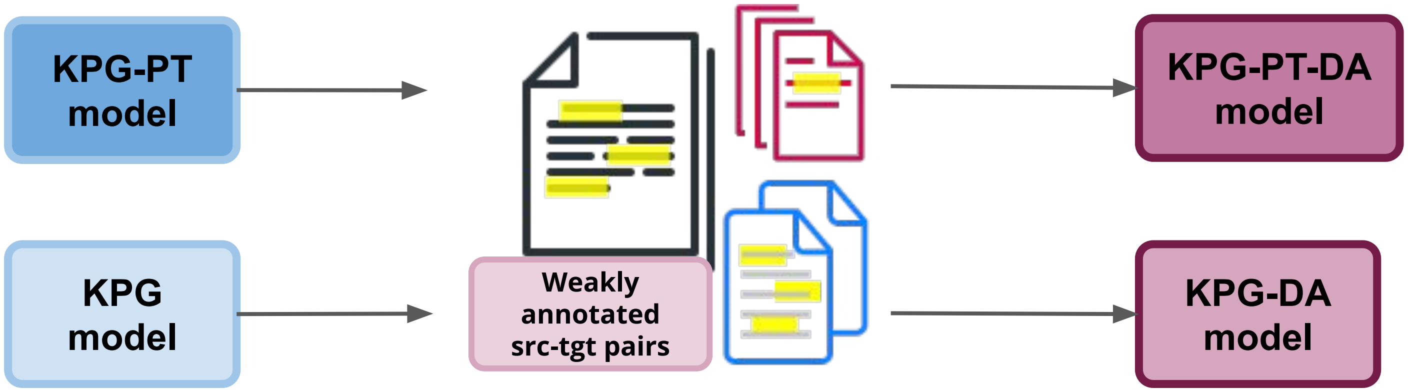
Stage 2.1 - Transfer Labeling

- Generate pseudo-keyphrases with KPG-PT models for target domains
    - Can easily scale up with un-annotated documents



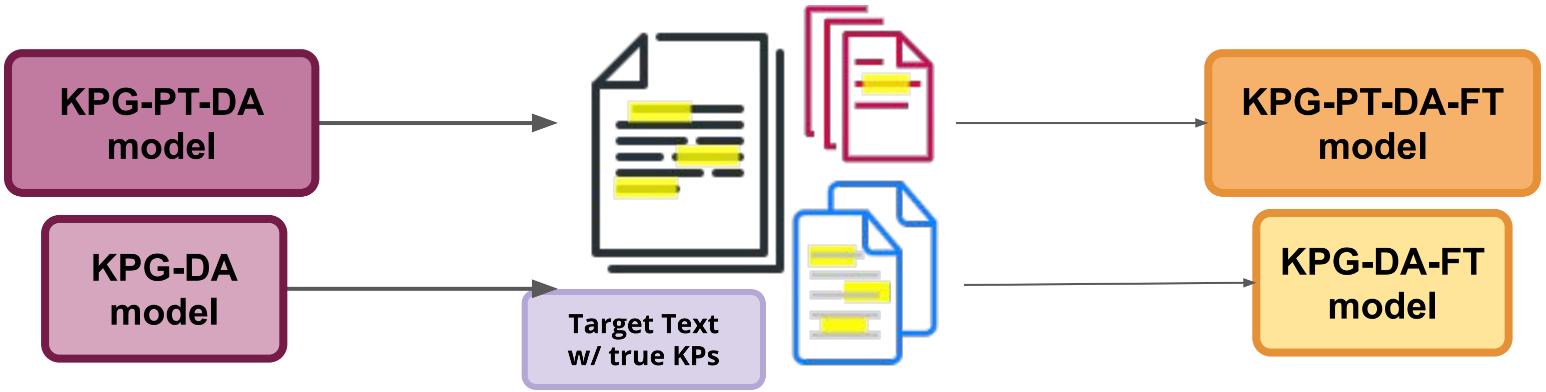
Stage 2.2 - Target Domain Adaptation w/ Transferred Labels

- Adapt KPG models to a target domain with generated pseudo keyphrases



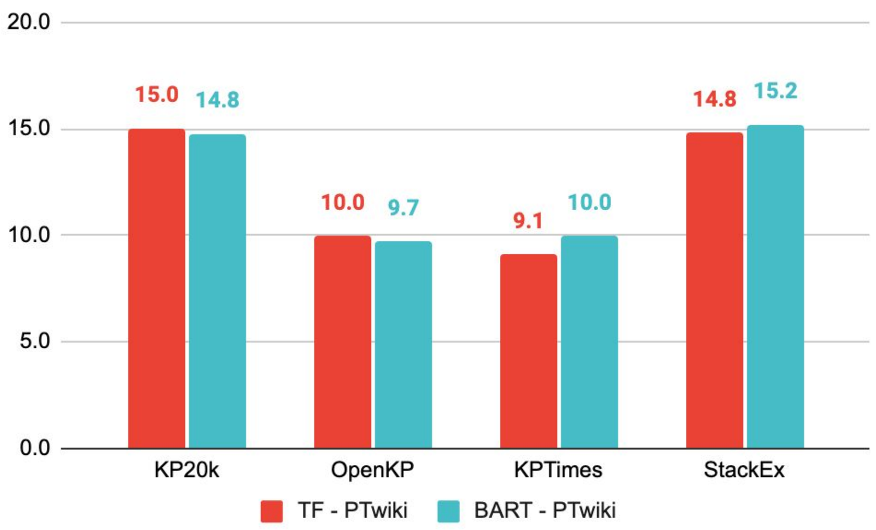
Stage 3 - Supervised Fine-Tuning

- Fine-tune KPG models with true annotated data
- Few-shot learning: 100/1k/10k annotated data examples



Result 1 - Zero-shot Scores of Stage-1

- Stage 1 only
  - **PT-wiki**: Pre-Training with Wikipedia Distant Supervision
  - Wikipedia pretraining can achieve decent zero-shot performance



Zero-shot Prediction

CALCULATE TAX SAVINGS

Do I Pay a Penalty on Early Withdrawal From a Thrift Savings Plan for College Tuition on My Taxes?

By Naomi Smith

For most federal and military employees, the Thrift Savings Plan offers comparable benefits to private-sector individual retirement accounts and 401(k) plans. However, the options for penalty-free early withdrawals are not as generous as with other retirement plans. The TSP allows you to withdraw your money early, but if it's going for college tuition you'll get stuck with a 10 percent penalty as well as any taxes owed on the distribution. You may find other options more advantageous.

Present predicted KPs

Roth IRA
TSP Loans
Early Withdrawal
401 k
Rollover to an IRA
Thrift

Absent predicted KPs

Traditional IRA
Personal financial problems
Financial savers

Zero-shot Prediction

Apple Profit Soars 73% as Sales Rise

Apple shipped 1.76 million Macs in the quarter, including those on sale at the Apple Store in Tokyo, where a boy looked at a MacBook Pro laptop.
Tomohiro Ohsumi/Bloomberg News

By Laurie J. Flynn

July 26, 2007

SAN FRANCISCO, July 25 — Apple on Wednesday reported a 73 percent jump in quarterly profit on strong sales of Macs and iPods, beating Wall Street forecasts. It also alleviated some concerns about early sales of the iPhone.

Investors were spooked on Tuesday when AT&T, which provides service for the phone, said it had activated just 146,000 iPhones in the day and a half from its release to the end of the quarter, far fewer than some analysts had expected. That sent Apple's stock down 6 percent.

Present predicted KPs

AT & T
Thomson Financial
Macs
chief financial officer
Timothy D Cook
iPhone

Absent predicted KPs

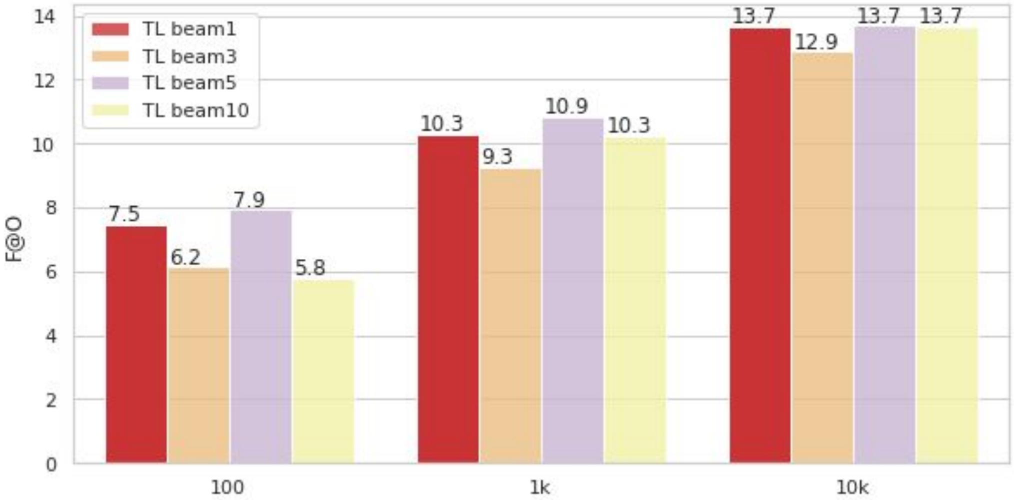
Apple Inc
Corporate affairs
First quarter
Computer companies of the United States

Result 2 - Strategies for Domain Adaptation

- Settings
  - Train Transformer (from scratch) with 100k pseudo-labelled documents (CS papers)
  - 40k steps
- Compare three strategies
  - Transfer Labeling (TL)
  - Noun Phrase (NP): extracted with Spacy, randomly select K phrase
  - Random Span (RS): same as T5
- Report Stage 2 + Stage 3 (DA+FT)

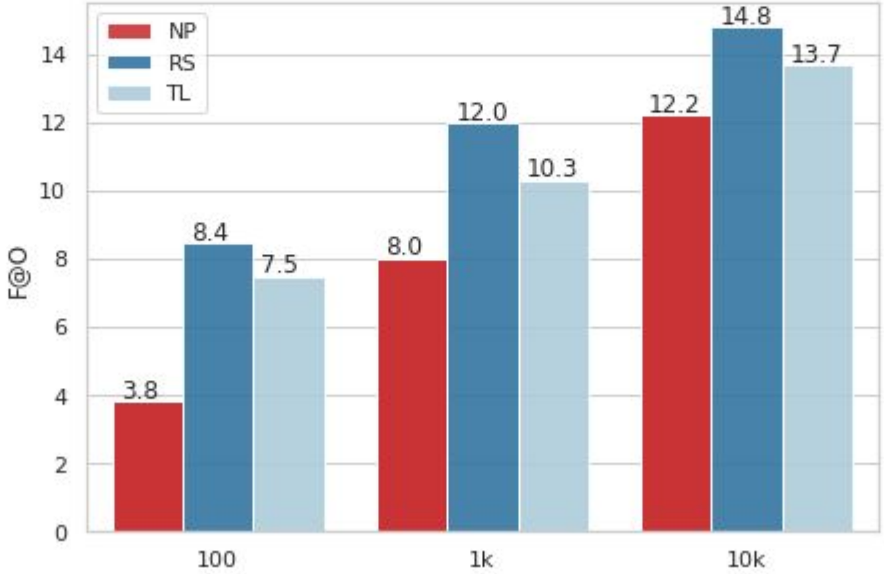
Result 2 - Strategies for Domain Adaptation

- Impact of beam width on transferred labels
  - Larger beam width leads to more generated phrases, but also more noise
  - Use greedy decoding (beam=1) for its efficiency



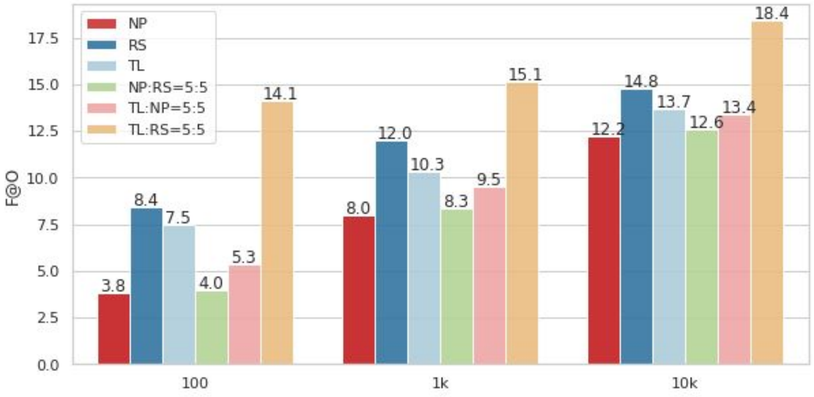
Result 2 - Strategies for Domain Adaptation

- Compare transferred labels, noun phrases and random spans
- DA w/ TL outperforms NP significantly, but works worse than RS



Result 2 - Strategies for Domain Adaptation

- Domain adaptation (Stage-2 only) with three strategies
  - TL: transferred labels
  - NP: noun phrases
  - RS: random spans
- Takeaways
  - DA w/ TL outperforms NP significantly, but works worse than RS
  - TL+RS can be complementary
  - We blend TL+RS in Domain Adaptation for better generalizability



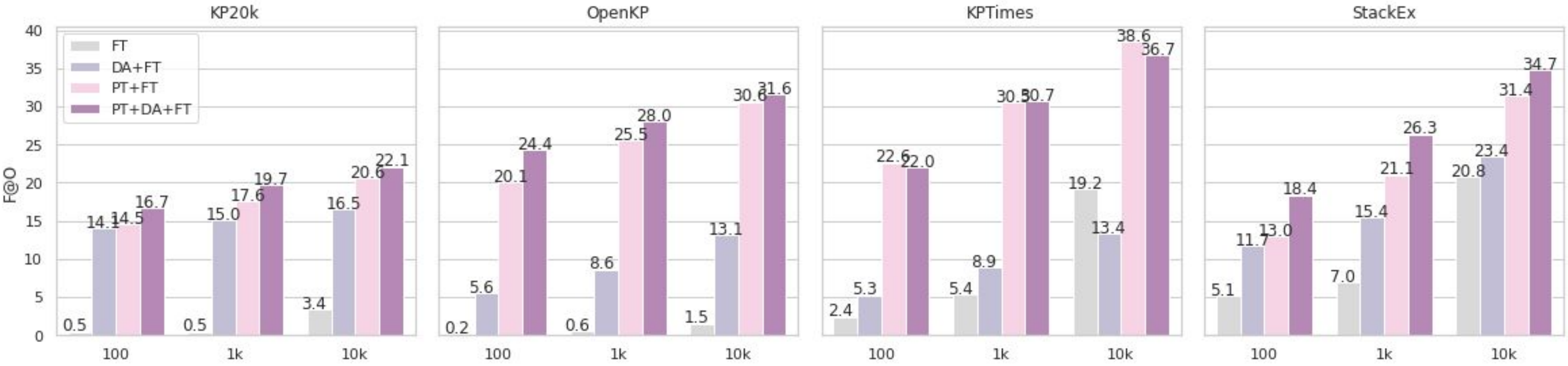
Result 3 - Few-shot KPG

- Ablation of Stage 1/2/3
- Domain adaptation (Stage 2)
  - Adapt to each target domain with 100k documents
  - Strategy of pseudo-keyphrase: TL:RS=5:5

ECIR 2022

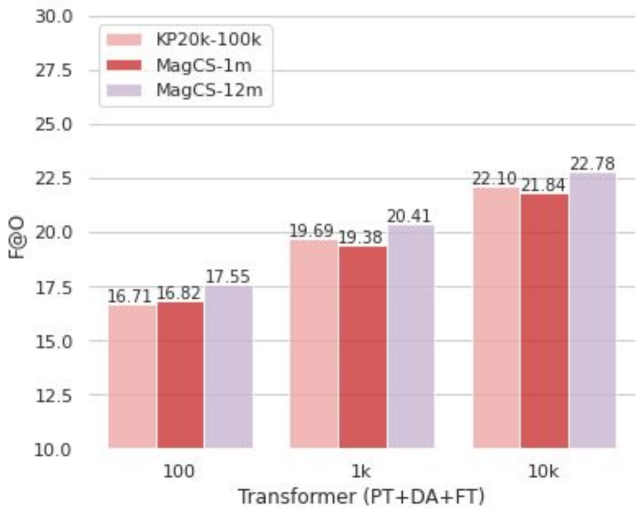
Result 3 - Few-shot KPG

- PT leads to large boost on OpenKP and KPTimes (PT+FT, Stage 1+3)
- Combining pretraining and domain adaptation achieves the best (PT+DA+FT, Stage 1+2+3)



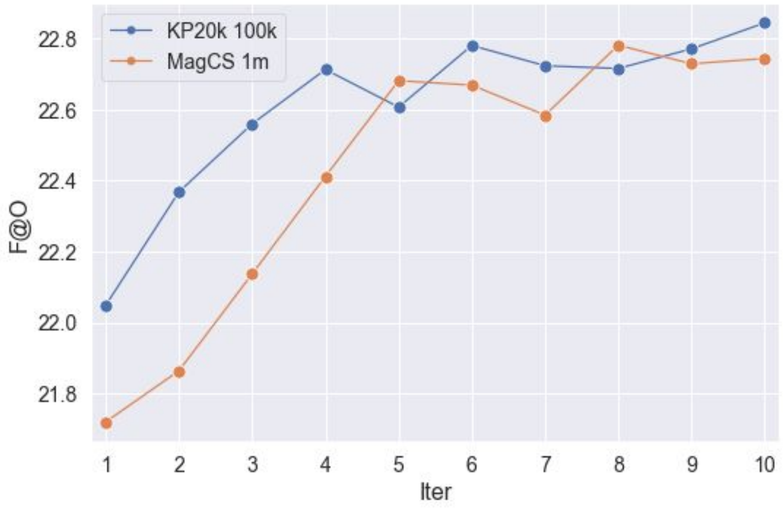
Result 4 - Scale-up with More Data for DA

- Microsoft Academic Graph (MAG)
  - A large academic database (116M paper records)
- Scale up transfer labeling with 12M CS papers
  - KP20k-100k / MagCS-1m / MagCS-12m



Result 5 - Self-training

- Setting
  - PT + $(DA)^{N\_iter}$ + FT
  - Iter 1-5: lr=1e-5, step=20k
  - Iter 6-10: lr=5e-6, step=10k



Part III

Learning Better Keyphrase Representations