UCPhrase: Unsupervised Context-aware Phrase Tagging

Xiaotao Gu*, **Zihan Wang***, Zhenyu Bi, Yu Meng, Liyuan Liu, Jiawei Han, Jingbo Shang

University of Illinois Urbana Champaign, University of California San Diego

ziw224@ucsd.edu

06.25.2021

Why do we need Phrases?

Unigram words are ambiguous

Phrases help understand better

- Phrase tagging is the task of identifying phrases in sentences.
- Can be useful for Entity Recognition, Text Classification, Information Retrieval, etc.

Challenges

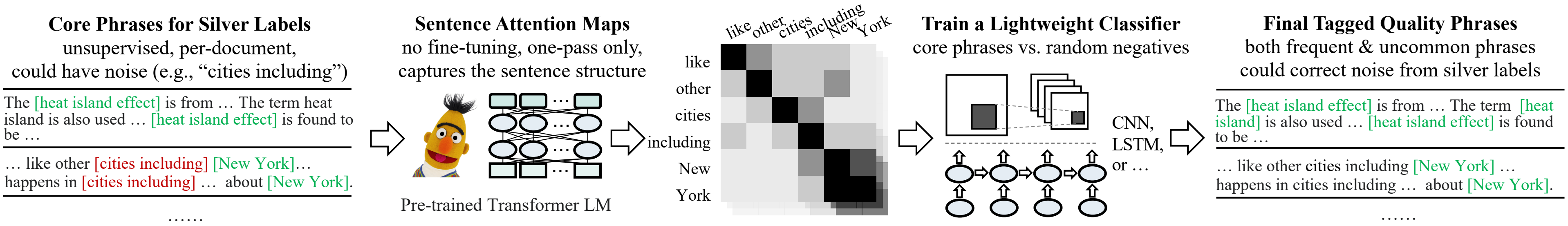
context awareness

?		Spacy/StanfordNLP
TopMine	AutoPhrase	
unsupervised	*distantly supervised*	*supervised*

supervision

- Tradeoff between context awareness and supervision
  - Supervised taggers require large scale human annotations.
  - Statistics based unsupervised/distantly supervised models do not need human annotation, but are context-agnostic and require enough frequencies
- Is there a model that is both context aware and unsupervised?

UCPhrase Overview



Core Phrase Mining

- How do human readers accumulate new phrases?

Doc1: …a study about [heat island effect]… The [heat island effect] arises because the buildings…of their [heat island effect]…
Doc2: …propose to extract [core phrases]… robust to potential noise in [core phrases]… the surface names of [core phrases]…

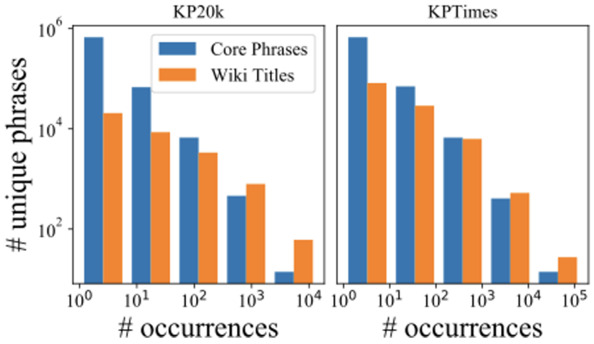
- We look for repeatedly used word sequences in a document, which are likely to be phrases by definition
  - Even without any prior knowledge we can recognize these consistently used patterns from a document

Core Phrase Mining

- Independently mine **max word sequential patterns**…
    - filter out uninformative patterns (e.g. “of a”) with a stopword list
- …within each document.
    - preserve contextual completeness (“biomedical data mining” vs. “data mining”)
    - avoid potential noises from propagating to the entire corpus
- These phrases are called Core Phrases.

Quality of Core Phrases

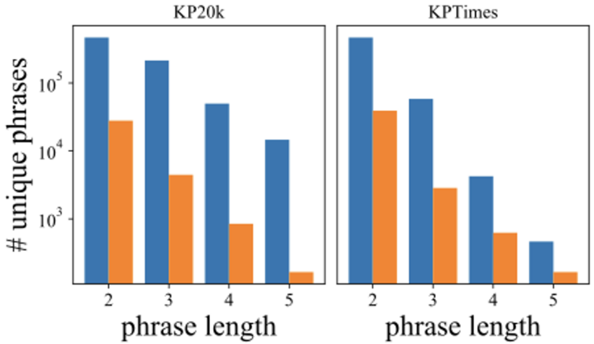
- Advantages of core phrases over distant supervision
  - Independent of KB
  - Better **quantity** and **diversity**
  - Better **contextual completeness**



Distant Supervision based on Wiki Entities

Doc1: … study about heat [island effect] … The heat [island effect] arises because the buildings…of their heat [island effect]…

Doc2: … propose to extract core phrases … robust to potential noise in core phrases … the surface names of core phrases…



Quality of Core Phrases

Examples from publications

user actions	embedded systems
shared applications	group decision making
ascillation mode	jit compilers
quantization noise	aggregation operator
hqcrff-based modulator	archival records
dynamic range	recordkeeping metadata
business reporting language	case study
ontology representation	digital preservation
self-organizing map	confidence intervals
movement threshold	learning process
location update	adaptive subspace iteration
wireless communication networks	propositional formula
ping-pong lu effect	security protocols
sensory input	singular superlinear boundary
complement graph	**parallel generation**
high resolution clich	surface grids
cellular automata	structured model reduction
white noise	initial organizational decisions
java virtual machines	power consumption

Examples from news articles

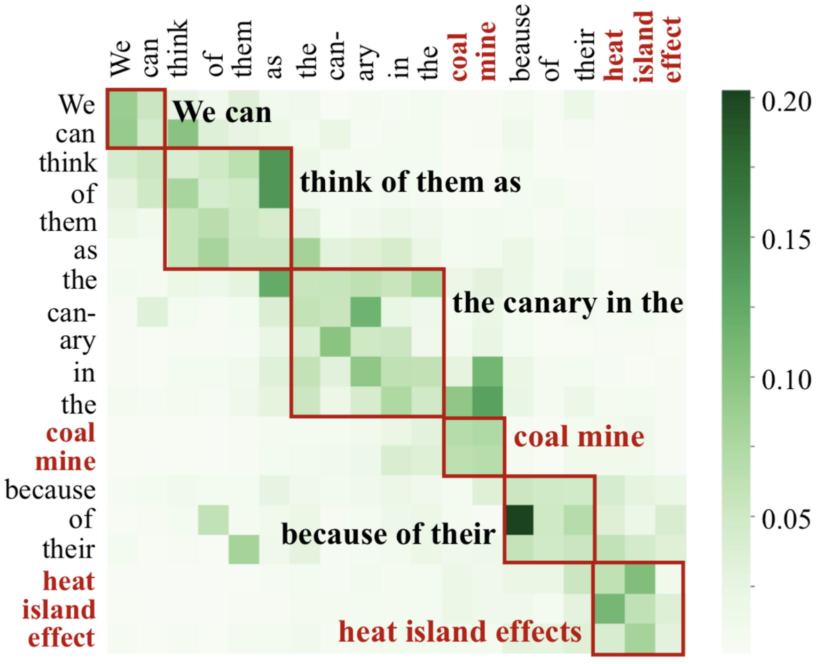
paul manafort	21st century
chief speechwriter	playoff series
campaign chairman	energy department
silver linings	world economic crisis
staff members	**mohawk river**
stephen miller	high school
bellevue hospital	criminal investigation
redistricting commission	cubic meters
dallas hospital	gas prices
ebola patients	lloyds banking groups
fellow democrat	private ownership
pulaski meat products	retail investors
push-button locks	**royal bank**
jiang tianyong	payment system
amnesty international	european central bank
human rights	countries including
jason collins	euro zone countries
district attorney	brookhaven national laboratory
united states	solar system

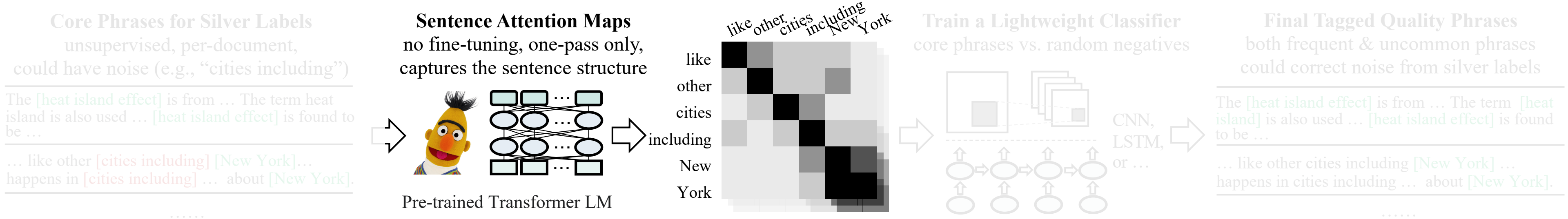
Learning with Silver Labels

- What features can the model learn to distinguish phrases?
  - Statistics: frequency, word-word co-occurrence, inverse document frequency
    - requires enough frequency to be a stable signal
    - does not generalize well to emerging, new phrases.
  - Embedding-based Features: from a pre-trained language model (BERT)
    - embedding features are word identifiable -- it tells you which word you are looking at
    - easy to rigidly memorize all seen phrases / words in the training set
    - a dictionary matching model can easily achieve 0% training error, but cannot generalize to unseen phrases

Attention Features

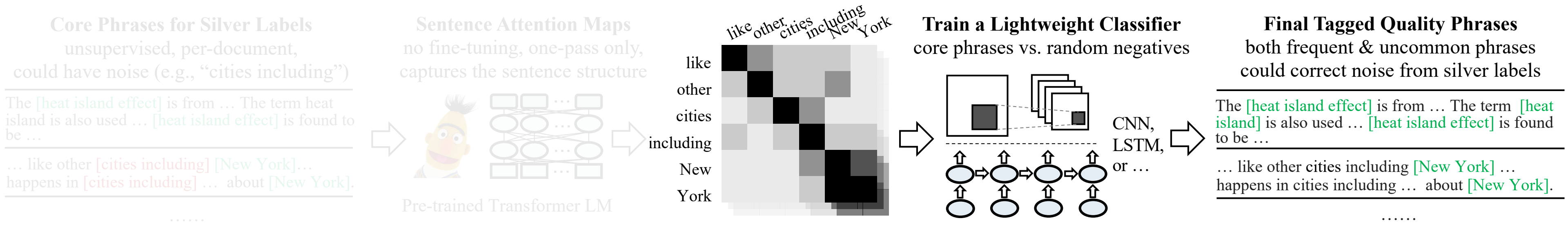
- From BERT, we also have attentions:
  - capture connections between tokens
  - the **attention map** of a sentence vividly visualizes its **inner structure**
  - high quality phrases should have **distinct attention patterns** from ordinary spans





Phrase Tagging

- Given a sentence, treat all possible n-grams as candidates
- For each candidate of length K extract its K*K attention map as feature
  - each attention head from each layer of a Transformer model will generate one attention map
  - for a RoBERTa base model, each candidate will have a (12*12 x K*K) = (144 x K*K) attention map
- Train a lightweight 2-layer CNN model for binary classification: is a phrase or not
- Training is as **fast** as one inference pass of the LM through the corpus (CNN training time is almost negligible)



Evaluation: tasks

UC San Diego

Task I. Corpus-level Phrase Ranking

Extracted Top Phrases

- Support Vector Machine
- information extraction
- information extraction systems
- supervised classifier
- safety consultant
- Richard Healing
- member of
- Transportation Safety Board
- used in

…..

Prec. @ 10 = 80%

Task II. Document-level Keyphrase Extraction

Doc1 Gold Keyphrases:

- Richard Healing
- Transportation Safety Board

Tagged phrases as candidates

- Richard Healing
- former member
- Transportation Safety Board

Rec. = 100%

Ranked by TF-IDF

- Transportation Safety Board
- Richard Healing
- safety consultant

F_1@ 3 = 80%

Task III. Sentence-level Phrase Tagging

Human Annotators (* 3):

[Support Vector Machine] is a member of [supervised classifiers] widely used in [information extraction systems] .

System Prediction:

[Support Vector Machine] is a [member of] [supervised classifiers] widely used in [information extraction] systems.

Rec. = 66.7%, Prec. = 50%, F_1 = 57.2%
(average over all annotators)

Coarse → *Fine-grained*

Evaluation: tasks

Task I. Corpus-level Phrase Ranking

Extracted Top Phrases

- Support Vector Machine
- information extraction
- information extraction systems
- supervised classifier
- safety consultant
- Richard Healing
- member of
- Transportation Safety Board
- used in

…..

Prec. @ 10 = 80%

Task II. Document-level Keyphrase Extraction

Doc1 Gold Keyphrases:

- Richard Healing
- Transportation Safety Board

Tagged phrases as candidates

- Richard Healing
- former member
- Transportation Safety Board

Rec. = 100%

Ranked by TF-IDF

- Transportation Safety Board
- Richard Healing
- safety consultant

F_1@ 3 = 80%

Task III. Sentence-level Phrase Tagging

Human Annotators (* 3):

[Support Vector Machine] is a member of [supervised classifiers] widely used in [information extraction systems] .

System Prediction:

[Support Vector Machine] is a [member of] [supervised classifiers] widely used in [information extraction] systems.

Rec. = 66.7%, Prec. = 50%, F_1 = 57.2% (average over all annotators)

Coarse → *Fine-grained*

Evaluation: tasks

UC San Diego

Task I. Corpus-level Phrase Ranking

Extracted Top Phrases

- Support Vector Machine
- information extraction
- information extraction systems
- supervised classifier
- safety consultant
- Richard Healing
- member of
- Transportation Safety Board
- used in

.....

Prec. @ 10 = 80%

Task II. Document-level Keyphrase Extraction

Doc1 Gold Keyphrases:

- Richard Healing
- Transportation Safety Board

Tagged phrases as candidates

- Richard Healing
- former member
- Transportation Safety Board

Rec. = 100%

Ranked by TF-IDF

- Transportation Safety Board
- Richard Healing
- safety consultant

F_1@ 3 = 80%

Task III. Sentence-level Phrase Tagging

Human Annotators (* 3):

[Support Vector Machine] is a member of [supervised classifiers] widely used in [information extraction systems] .

System Prediction:

[Support Vector Machine] is a [member of] [supervised classifiers] widely used in [information extraction] systems.

Rec. = 66.7%, Prec. = 50%, F_1 = 57.2%
(average over all annotators)

Coarse → *Fine-grained*

Evaluation: datasets

- Use largest existing keyphrase extraction datasets for evaluation
- Only use the unlabeled training corpus for model learning

- KP20k
  - CS publications, 176 words per doc
  - 527,000 docs for training, 20,000 docs for testin
- KPTimes
  - news articles, 907 words per doc
  - 259,923 docs for training, 20,000 docs for testing

Table 1: Dataset statistics on KP20k and KPTimes.

Statistics	**KP20k**	**KPTimes**
	Train Set	
# documents	527,090	259,923
# words per document	176	907
	Test Set	
# documents	20,000	20,000
# multi-word keyphrases	37,289	24,920
# unique	24,626	8,970
# absent in training corpus	4,171	2,940

Evaluation: compared methods

- Unsupervised Methods
  - **UCPhrase**: our method;
  - **TopMine**: statistics-based topical phrase mining;
- Distantly Supervised (+wiki)
  - **AutoPharse**: statistics-based classifier + POS-guided phrase segmentation model;
  - **Wiki+RoBERTa**: distant supervision + RoBERTa embedding as features + early stopping;
- Pre-trained Phrase Taggers
  - **StanfordNLP**: chunking model with pre-trained POS-tagging model;
  - **Spacy**: industrial library with an off-the-shelf chunking model based on dependency parsing and POS tagging;

Evaluation: performance

Table 2: Evaluation results (%) of three tasks for all compared methods on datasets on two domains.

Method Type	Method Name	Task I: Phrase Ranking				Task II: KP Extract.				Task III: Phrase Tagging					
		KP20k		KPTimes		KP20K		KPTimes		KP20k			KPTimes		
		P@5K	P@50K	P@5K	P@50K	Rec.	F_1@10	Rec.	F_1@10	Prec.	Rec.	F_1	Prec.	Rec.	F_1
Pre-trained	PKE [3]	–	–	–	–	57.1	12.6	61.9	4.4	54.1	63.9	58.6	56.1	62.2	59.0
	Spacy [16]	–	–	–	–	59.5	15.3	60.8	8.6	56.3	68.7	61.9	61.9	62.9	62.4
	StanfordNLP [26]	–	–	–	–	51.7	13.9	60.8	8.7	48.3	60.7	53.8	56.9	60.3	58.6
Distantly Supervised	AutoPhrase [33]	97.5	96.0	96.5	95.5	62.9	18.2	77.8	10.3	55.2	45.2	49.7	44.2	47.7	45.9
	Wiki+RoBERTa	**100.0**	**98.5**	**99.0**	**96.5**	**73.0**	19.2	64.5	9.4	58.1	64.2	61.0	60.9	65.6	63.2
Unsupervised	TopMine [8]	81.5	78.0	85.5	71.0	53.3	15.0	63.4	8.5	39.8	41.4	40.6	32.0	36.3	34.0
	UCPhrase (ours)	96.5	96.5	96.5	95.5	72.9	**19.7**	**83.4**	**10.9**	**69.9**	**78.3**	**73.9**	**69.1**	**78.9**	**73.5**

Evaluation: performance

Table 2: Evaluation results (%) of three tasks for all compared methods on datasets on two domains.

Method Type	Method Name	Task I: Phrase Ranking				Task II: KP Extract.				Task III: Phrase Tagging					
		KP20k		KPTimes		KP20K		KPTimes		KP20k			KPTimes		
		P@5K	P@50K	P@5K	P@50K	Rec.	F_1@10	Rec.	F_1@10	Prec.	Rec.	F_1	Prec.	Rec.	F_1
Pre-trained	PKE [3]	–	–	–	–	57.1	12.6	61.9	4.4	54.1	63.9	58.6	56.1	62.2	59.0
	Spacy [16]	–	–	–	–	59.5	15.3	60.8	8.6	56.3	68.7	61.9	61.9	62.9	62.4
	StanfordNLP [26]	–	–	–	–	51.7	13.9	60.8	8.7	48.3	60.7	53.8	56.9	60.3	58.6
Distantly Supervised	AutoPhrase [33]	97.5	96.0	96.5	95.5	62.9	18.2	77.8	10.3	55.2	45.2	49.7	44.2	47.7	45.9
	Wiki+RoBERTa	**100.0**	**98.5**	**99.0**	**96.5**	**73.0**	19.2	64.5	9.4	58.1	64.2	61.0	60.9	65.6	63.2
Unsupervised	TopMine [8]	81.5	78.0	85.5	71.0	53.3	15.0	63.4	8.5	39.8	41.4	40.6	32.0	36.3	34.0
	UCPhrase (ours)	96.5	96.5	96.5	95.5	72.9	**19.7**	**83.4**	**10.9**	**69.9**	**78.3**	**73.9**	**69.1**	**78.9**	**73.5**

- Distantly Supervised methods performs the best on Phrase Ranking
  - Understandable, since phrases directly from Wikipedia will be assigned a high score.
  - UCPhrase have a good enough quality.

Evaluation: performance

Table 2: Evaluation results (%) of three tasks for all compared methods on datasets on two domains.

Method Type	Method Name	Task I: Phrase Ranking				Task II: KP Extract.				Task III: Phrase Tagging					
		KP20k		KPTimes		KP20K		KPTimes		KP20k			KPTimes		
		P@5K	P@50K	P@5K	P@50K	Rec.	F_1@10	Rec.	F_1@10	Prec.	Rec.	F_1	Prec.	Rec.	F_1
Pre-trained	PKE [3]	–	–	–	–	57.1	12.6	61.9	4.4	54.1	63.9	58.6	56.1	62.2	59.0
	Spacy [16]	–	–	–	–	59.5	15.3	60.8	8.6	56.3	68.7	61.9	61.9	62.9	62.4
	StanfordNLP [26]	–	–	–	–	51.7	13.9	60.8	8.7	48.3	60.7	53.8	56.9	60.3	58.6
Distantly Supervised	AutoPhrase [33]	97.5	96.0	96.5	95.5	62.9	18.2	77.8	10.3	55.2	45.2	49.7	44.2	47.7	45.9
	Wiki+RoBERTa	**100.0**	**98.5**	**99.0**	**96.5**	**73.0**	19.2	64.5	9.4	58.1	64.2	61.0	60.9	65.6	63.2
Unsupervised	TopMine [8]	81.5	78.0	85.5	71.0	53.3	15.0	63.4	8.5	39.8	41.4	40.6	32.0	36.3	34.0
	UCPhrase (ours)	96.5	96.5	96.5	95.5	72.9	**19.7**	**83.4**	**10.9**	**69.9**	**78.3**	**73.9**	**69.1**	**78.9**	**73.5**

- UCPhrase finds keyphrases much better in documents
    - Much more keyphrases found in the KPTimes dataset than any other methods

Evaluation: performance

Table 2: Evaluation results (%) of three tasks for all compared methods on datasets on two domains.

Method Type	Method Name	Task I: Phrase Ranking				Task II: KP Extract.				Task III: Phrase Tagging					
		KP20k		KPTimes		KP20K		KPTimes		KP20k			KPTimes		
		P@5K	P@50K	P@5K	P@50K	Rec.	F_1@10	Rec.	F_1@10	Prec.	Rec.	F_1	Prec.	Rec.	F_1
Pre-trained	PKE [3]	–	–	–	–	57.1	12.6	61.9	4.4	54.1	63.9	58.6	56.1	62.2	59.0
	Spacy [16]	–	–	–	–	59.5	15.3	60.8	8.6	56.3	68.7	61.9	61.9	62.9	62.4
	StanfordNLP [26]	–	–	–	–	51.7	13.9	60.8	8.7	48.3	60.7	53.8	56.9	60.3	58.6
Distantly Supervised	AutoPhrase [33]	97.5	96.0	96.5	95.5	62.9	18.2	77.8	10.3	55.2	45.2	49.7	44.2	47.7	45.9
	Wiki+RoBERTa	**100.0**	**98.5**	**99.0**	**96.5**	**73.0**	19.2	64.5	9.4	58.1	64.2	61.0	60.9	65.6	63.2
Unsupervised	TopMine [8]	81.5	78.0	85.5	71.0	53.3	15.0	63.4	8.5	39.8	41.4	40.6	32.0	36.3	34.0
	UCPhrase (ours)	96.5	96.5	96.5	95.5	72.9	**19.7**	**83.4**	**10.9**	**69.9**	**78.3**	**73.9**	**69.1**	**78.9**	**73.5**

- UCPhrase performs the best in sentence level Phrase Tagging
    - Shines in more fine-grained tasks: gives more diverse, low frequency phrases.

Evaluation: ablation study

UC San Diego

Table 3: Ablation study of UCPhrase model variants (%).

	Design Choices			KP Extract.				Phrase Tagging					
				KP20k		KPTimes		KP20k			KPTimes		
	supervision	feature	fine-tune	Rec.	F_1@10	Rec.	F_1@10	Prec.	Rec.	F_1	Prec.	Rec.	F_1
UCPhrase	core	attention	no	72.9	**19.7**	**83.4**	**10.9**	69.9	**78.3**	**73.9**	**69.1**	**78.9**	**73.5**
Variants	Wiki	attention	no	68.7	17.7	79.4	10.7	**72.1**	71.9	72.0	64.1	67.6	65.8
	Wiki	embedding	no	73.0	19.2	64.5	9.4	60.9	65.6	63.2	60.9	65.6	63.2
	core	embedding	no	79.3	**19.7**	78.7	10.2	68.4	74.6	71.4	55.7	64.8	59.9
	core	embedding	yes	**80.3**	**19.7**	73.9	9.9	68.6	74.8	71.6	53.3	64.5	59.0

- Varying Supervision (core, Wiki) and Feature (attention, embedding)
  - Using Core Phrases is better than using Wiki titles (no matter the choice of feature).
  - Using Attention is better than using Embeddings (no matter the choice of supervision).

Conclusions & Future Work

- Core Phrase mining
    - Finds silver label phrases
    - More diverse than string matching
- Attention features
    - Rich linguistic knowledge from LMs.
    - Less prune to overfit than embeddings.
- Pseudo data + attention features is worth exploring in other text mining tasks:
    - coreference resolution, dependency parsing, named entity recognition

All data & code are available at https://github.com/xgeric/UCPhrase-exp