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

- 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?
### Core Phrases for Silver Labels
unsupervised, per-document, could have noise (e.g., “cities including”)

The [heat island effect] is from … The term heat island is also used … [heat island effect] is found to be …

… like other [cities including] [New York]… happens in [cities including] … about [New York].

### Sentence Attention Maps
no fine-tuning, one-pass only, captures the sentence structure

Pre-trained Transformer LM

### Train a Lightweight Classifier
core phrases vs. random negatives

like other cities including New York

### Final Tagged Quality Phrases
both frequent & uncommon phrases could correct noise from silver labels

The [heat island effect] is from … The term [heat island] is also used … [heat island effect] is found to be …

… like other cities including [New York] … happens in cities including … about [New York].
Core Phrase Mining

• How do human readers accumulate new phrases?

\[
\text{Doc1: } \ldots \text{a study about [heat island effect]} \ldots \ \text{The [heat island effect] arises because the buildings...of their [heat island effect]} \ldots
\]

\[
\text{Doc2: } \ldots \text{propose to extract [core phrases]} \ldots \ \text{robust to potential noise in [core phrases]} \ldots \ \text{the surface names of [core phrases]} \ldots
\]

• 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.

---

Core Phrases for Silver Labels
unsupervised, per-document, could have noise (e.g., “cities including”)
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
- shared applications
- ascillation mode
- quantization noise
- hqcrff-based modulator
- dynamic range
- business reporting language
- ontology representation
- self-organizing map
- movement threshold
- location update
- wireless communication networks
- ping-pong lu effect
- sensory input
- complement graph
- high resolution clich
- cellular automata
- white noise
- java virtual machines
- embedded systems
- group decision making
- jit compilers
- aggregation operator
- archival records
- recordkeeping metadata
- case study
- digital preservation
- confidence intervals
- learning process
- adaptive subspace iteration
- propositional formula
- security protocols
- singular superlinear boundary
- parallel generation
- surface grids
- structured model reduction
- initial organizational decisions
- power consumption

Examples from news articles

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

**Ranked by TF-IDF**
- Transportation Safety Board
- Richard Healing
- safety consultant

*Rec. = 100%*  

*F1 @ 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%, F1 = 57.2%*  

(average over all annotators)

---

**Coarse**  

**Fine-grained**
### 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

**Ranked by TF-IDF**
- Transportation Safety Board
- Richard Healing
- safety consultant

*Rec. = 100%  
F₁ = 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%, F₁ = 57.2%  
(average over all annotators)*

---

*Coarse ➔ Fine-grained*
# Evaluation: tasks

<table>
<thead>
<tr>
<th>Task I. Corpus-level Phrase Ranking</th>
<th>Task II. Document-level Keyphrase Extraction</th>
<th>Task III. Sentence-level Phrase Tagging</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Extracted Top Phrases</strong></td>
<td><strong>Doc1 Gold Keyphrases:</strong></td>
<td>*<em>Human Annotators (<em>3):</em></em></td>
</tr>
<tr>
<td>- Support Vector Machine</td>
<td>- Richard Healing</td>
<td>- Support Vector Machine is a member</td>
</tr>
<tr>
<td>- information extraction</td>
<td>- Transportation Safety Board</td>
<td>of [supervised classifiers] widely used</td>
</tr>
<tr>
<td>- information extraction systems</td>
<td></td>
<td>in [information extraction systems].</td>
</tr>
<tr>
<td>- supervised classifier</td>
<td><strong>Tagged phrases as candidates</strong></td>
<td><strong>System Prediction:</strong></td>
</tr>
<tr>
<td>- safety consultant</td>
<td>- Richard Healing</td>
<td>- Support Vector Machine is a [member</td>
</tr>
<tr>
<td>- Richard Healing</td>
<td>- former member</td>
<td>of] [supervised classifiers] widely used</td>
</tr>
<tr>
<td>- member of</td>
<td>- Transportation Safety Board</td>
<td>in [information extraction] systems.</td>
</tr>
<tr>
<td>- Transportation Safety Board</td>
<td></td>
<td>**Rec. = 66.7%, Prec. = 50%, $F_1 = 57.2%$$\text{ (average over all annotators)}$$</td>
</tr>
<tr>
<td>- used in</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Prec. @ 10 = 80%</strong></td>
<td><strong>Ranked by TF-IDF</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Transportation Safety Board</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Richard Healing</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- safety consultant</td>
<td></td>
</tr>
<tr>
<td><strong>Coarse</strong></td>
<td></td>
<td><strong>Fine-grained</strong></td>
</tr>
</tbody>
</table>
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 testing

- 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.

<table>
<thead>
<tr>
<th>Statistics</th>
<th>KP20k</th>
<th>KPTimes</th>
</tr>
</thead>
<tbody>
<tr>
<td># documents</td>
<td>527,090</td>
<td>259,923</td>
</tr>
<tr>
<td># words per document</td>
<td>176</td>
<td>907</td>
</tr>
<tr>
<td>Test Set</td>
<td></td>
<td></td>
</tr>
<tr>
<td># documents</td>
<td>20,000</td>
<td>20,000</td>
</tr>
<tr>
<td># multi-word keyphrases</td>
<td>37,289</td>
<td>24,920</td>
</tr>
<tr>
<td># unique</td>
<td>24,626</td>
<td>8,970</td>
</tr>
<tr>
<td># absent in training corpus</td>
<td>4,171</td>
<td>2,940</td>
</tr>
</tbody>
</table>
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;
### Table 2: Evaluation results (%) of three tasks for all compared methods on datasets on two domains.

<table>
<thead>
<tr>
<th>Method Type</th>
<th>Method Name</th>
<th>Task I: Phrase Ranking</th>
<th>Task II: KP Extract.</th>
<th>Task III: Phrase Tagging</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>KP20k</td>
<td>KPTimes</td>
<td>KP20k</td>
</tr>
<tr>
<td></td>
<td></td>
<td>P@5K</td>
<td>P@50K</td>
<td>P@5K</td>
</tr>
<tr>
<td>Pre-trained</td>
<td>PKE [3]</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td></td>
<td>Spacy [16]</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td></td>
<td>StanfordNLP [26]</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td></td>
<td>AutoPhrase [33]</td>
<td>97.5</td>
<td>96.0</td>
<td>96.5</td>
</tr>
<tr>
<td></td>
<td>Wiki+RoBERTa</td>
<td>100.0</td>
<td>98.5</td>
<td>99.0</td>
</tr>
<tr>
<td>Distantly Supervised</td>
<td>TopMine [8]</td>
<td>81.5</td>
<td>78.0</td>
<td>85.5</td>
</tr>
<tr>
<td></td>
<td>UCPhrase (ours)</td>
<td>96.5</td>
<td>96.5</td>
<td>96.5</td>
</tr>
<tr>
<td>Unsupervised</td>
<td>AutoPhrase [33]</td>
<td>97.5</td>
<td>96.0</td>
<td>96.5</td>
</tr>
<tr>
<td></td>
<td>Wiki+RoBERTa</td>
<td>100.0</td>
<td>98.5</td>
<td>99.0</td>
</tr>
</tbody>
</table>
**Evaluation: performance**

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

<table>
<thead>
<tr>
<th>Method Type</th>
<th>Method Name</th>
<th>Task I: Phrase Ranking</th>
<th>Task II: KP Extract.</th>
<th>Task III: Phrase Tagging</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>KP20k</td>
<td>KP50k</td>
<td>KPTimes</td>
</tr>
<tr>
<td>Pre-trained</td>
<td>PKE [3]</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td></td>
<td>Spacy [16]</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td></td>
<td>StanfordNLP [26]</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>Distantly Supervised</td>
<td>AutoPhrase [33]</td>
<td>97.5</td>
<td>96.0</td>
<td>96.5</td>
</tr>
<tr>
<td></td>
<td>Wiki+RoBERTa</td>
<td><strong>100.0</strong></td>
<td><strong>98.5</strong></td>
<td><strong>99.0</strong></td>
</tr>
<tr>
<td>Unsupervised</td>
<td>TopMine [8]</td>
<td>81.5</td>
<td>78.0</td>
<td>85.5</td>
</tr>
<tr>
<td></td>
<td>UCPhrase (ours)</td>
<td>96.5</td>
<td>96.5</td>
<td>96.5</td>
</tr>
</tbody>
</table>

- **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>
<thead>
<tr>
<th>Method Type</th>
<th>Method Name</th>
<th>Task I: Phrase Ranking</th>
<th>Task II: KP Extract.</th>
<th>Task III: Phrase Tagging</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>KP20k P@5K</td>
<td>KPTimes P@5K</td>
<td>KP20K Rec.</td>
</tr>
<tr>
<td>Pre-trained</td>
<td>PKE [3]</td>
<td>– – – –</td>
<td>– – – –</td>
<td>57.1 12.6</td>
</tr>
<tr>
<td></td>
<td>Spacy [16]</td>
<td>– – – –</td>
<td>– – – –</td>
<td>59.5 15.3</td>
</tr>
<tr>
<td></td>
<td>StanfordNLP [26]</td>
<td>– – – –</td>
<td>– – – –</td>
<td>51.7 13.9</td>
</tr>
<tr>
<td>Distantly Supervised</td>
<td>AutoPhrase [33]</td>
<td>97.5 96.0</td>
<td>96.5 95.5</td>
<td>62.9 18.2</td>
</tr>
<tr>
<td></td>
<td>Wiki+RoBERTa</td>
<td><strong>100.0</strong></td>
<td><strong>98.5</strong></td>
<td><strong>99.0</strong></td>
</tr>
<tr>
<td>Unsupervised</td>
<td>TopMine [8]</td>
<td>81.5 78.0</td>
<td>85.5 71.0</td>
<td>53.3 15.0</td>
</tr>
<tr>
<td></td>
<td>UCPhrase (ours)</td>
<td>96.5 96.5</td>
<td>96.5 95.5</td>
<td><strong>72.9</strong></td>
</tr>
</tbody>
</table>

- 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.

<table>
<thead>
<tr>
<th>Method Type</th>
<th>Method Name</th>
<th>Task I: Phrase Ranking</th>
<th></th>
<th>Task II: KP Extract.</th>
<th></th>
<th>Task III: Phrase Tagging</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>KP20k</td>
<td>P@5K</td>
<td>P@50K</td>
<td>KPTimes</td>
<td>P@5K</td>
<td>P@50K</td>
</tr>
<tr>
<td>Pre-trained</td>
<td>PKE [3]</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>57.1</td>
<td>12.6</td>
</tr>
<tr>
<td></td>
<td>Spacy [16]</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>59.5</td>
<td>15.3</td>
</tr>
<tr>
<td></td>
<td>StanfordNLP [26]</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>51.7</td>
<td>13.9</td>
</tr>
<tr>
<td>Distantly Supervised</td>
<td>AutoPhrase [33]</td>
<td>97.5</td>
<td>96.0</td>
<td>96.5</td>
<td>95.5</td>
<td>62.9</td>
<td>18.2</td>
</tr>
<tr>
<td></td>
<td>Wiki+RoBERTa</td>
<td>100.0</td>
<td>98.5</td>
<td>99.0</td>
<td>96.5</td>
<td>73.0</td>
<td>19.2</td>
</tr>
<tr>
<td>Unsupervised</td>
<td>TopMine [8]</td>
<td>81.5</td>
<td>78.0</td>
<td>85.5</td>
<td>71.0</td>
<td>53.3</td>
<td>15.0</td>
</tr>
<tr>
<td></td>
<td>UCPhrase (ours)</td>
<td>96.5</td>
<td>96.5</td>
<td>96.5</td>
<td>95.5</td>
<td>72.9</td>
<td>19.7</td>
</tr>
</tbody>
</table>

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

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

<table>
<thead>
<tr>
<th>Design Choices</th>
<th>KP Extract.</th>
<th></th>
<th>Phrase Tagging</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>KP20k</td>
<td>KPTimes</td>
<td>KP20k</td>
<td>KPTimes</td>
</tr>
<tr>
<td>supervision</td>
<td>Rec.</td>
<td>F$_{1@10}$</td>
<td>Rec.</td>
<td>F$_{1@10}$</td>
</tr>
<tr>
<td>UCPHrse</td>
<td>72.9</td>
<td>19.7</td>
<td>83.4</td>
<td>10.9</td>
</tr>
<tr>
<td>Wiki core</td>
<td>68.7</td>
<td>17.7</td>
<td>79.4</td>
<td>10.7</td>
</tr>
<tr>
<td>Wiki embedding</td>
<td>73.0</td>
<td>19.2</td>
<td>64.5</td>
<td>9.4</td>
</tr>
<tr>
<td>core attention</td>
<td>79.3</td>
<td>19.7</td>
<td>78.7</td>
<td>10.2</td>
</tr>
<tr>
<td>core embedding</td>
<td>80.3</td>
<td>19.7</td>
<td>73.9</td>
<td>9.9</td>
</tr>
</tbody>
</table>

- 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 prone 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