Measuring Fine-Grained Domain Relevance of Terms: A Hierarchical Core-Fringe Approach

Jie Huang, Kevin Chen-Chuan Chang, Jinjun Xiong, Wen-mei Hwu

University of Illinois at Urbana-Champaign
IBM Thomas J. Watson Research Center
IBM-Illinois Center for Cognitive Computing Systems Research (C3SR)
Fine-grained domain relevance:
The degree that a term is relevant to a given domain, and the given domain can be broad or narrow.
Applications

- Computer Science
  - Database
  - AI
  - Algorithm
  - ML
  - NLP
  - ...

Domain taxonomy construction

Document classification

- Machine translation, Natural language generation, Lexical semantics, ...

Domain-specific term extraction
Method: Overview

- **Seed terms**: machine learning, deep learning, few-shot learning, quantum mechanics...
- **Query terms**: few-shot learning, quantum chemistry...
- **Graph construction**
- **Model training**
- **Domain relevance**: 0.877, 0.001...
- **CFL, HiCFL**

- **Offline**
- **Online**
Core-Anchored Semantic Graph

Core terms: Terms associated with rich description information (Wikipedia article pages), e.g., *Machine Learning*

Fringe terms: Terms without rich description information, e.g., *Few-Shot Learning* (usually the long-tail ones)
Core-Anchored Semantic Graph

Connect fringe terms to relevant core terms via document ranking

“Bridge domain relevance of terms through term relevance”
Core-Fringe Learning (CFL)

Assume labels of core terms (domain-relevant or not) are available:

- \( a) \) propagate features of terms via term graph
- \( b) \) use labels of core terms for supervision

Graph convolution:

\[
h_i^{(l+1)} = \phi \left( \sum_{j \in N_i \cup \{i\}} \frac{1}{c_{ij}} W_c^{(l)} h_j^{(l)} + b_c^{(l)} \right)
\]

Loss:

\[
\mathcal{L} = - \sum_{i \in V_{\text{core}}} (y_i \log z_i + (1 - y_i) \log(1 - z_i))
\]

Domain Relevance
Hierarchical Core-Fringe Learning (HiCFL)

Hierarchy of domains: CS -> AI -> ML

“An ML term should also be relevant to CS” => Hierarchical Learning

Global information:
\[ z_p = \sigma(W^{(l_p)}_p a^{(l_p)}_p + b^{(l_p)}_p) \]
\[ a^{(l+1)}_p = \phi(W^{(l)}_p [a^{(l)}_p; h^{(l_c)}] + b^{(l)}_p) \]

Local information:
\[ z^{(l)}_q = \sigma(W^{(l)}_q a^{(l)}_q + b^{(l)}_q) \]
\[ a^{(l)}_q = \phi(W^{(l)}_t a^{(l)}_p + b^{(l)}_t) \]

Loss:
\[ \mathcal{L}_h = \epsilon(z_p, y^{(l_p)}) + \sum_{l=1}^{l_p} \epsilon(z^{(l)}_q, y^{(l)}) \]

Domain relevance:
\[ s = \alpha \cdot z_p + (1 - \alpha) \cdot (z^{(1)}_q \circ z^{(2)}_q, \ldots, z^{(l_p)}_q) \]
Automatic Annotation

Machine learning algorithms and Computer vision are categories in CS

=> Zero-shot learning is domain-relevant!
Hierarchical Positive-Unlabeled Learning

For narrow domains like *Deep Learning*, a category tree might not be available in Wikipedia. *Automatic Annotation is impractical* ⇒ *Hierarchical Positive-Unlabeled (PU) Learning*

For *Deep Learning (DL)*,
Hierarchy of domains: CS -> AI -> ML -> DL

- Automatic Annotation
- Positive: $k$ DL terms provided by users
- Negative: non-ML terms

\[
\mathcal{L}_h = \epsilon(z_p, y^{(l_p)}) + \sum_{l=1}^{l_p} \epsilon(z^{(l)}, y^{(l)})
\]
Method: Online

- **Seed terms**: machine learning, deep learning, few-shot learning, quantum mechanics, ...

- **Graph construction**

- **Model training**: CFL, HiCFL

- **Query terms**: few-shot learning, quantum chemistry, ...

- **Domain relevance**: online

Values: 0.877, 0.001, ...
Method: Online

- **Seed Terms**
  - machine learning
  - deep learning
  - few-shot learning
  - quantum mechanics

- **Query Terms**
  - few-shot learning
  - quantum chemistry

- **Domain Relevance**
  - 0.877
  - 0.001

- **Graph Construction**

- **Model Training**
  - CFL
  - HiCFL

- **Online**

- **Offline**
Experiments: Overview

**Comparison to Baselines:** Compare with existing methods on *Automatic Term Extraction*

**Comparison to Human Performance:** Compare with human professionals

**Case Studies:** Case studies for ML and DL

| domain | #terms   | core ratio |
|--------|----------|------------|
| CS     | ML       | 113,038    | 27.7%      |
| Phy    | QM       | 416,431    | 12.1%      |
| Math   | AA       | 103,984    | 26.4%      |

*Statistics of data.* 3 broad domains, 3 narrow domains
Comparison to Baselines

|                   | Computer Science | Physics      | Mathematics   |
|-------------------|------------------|--------------|---------------|
|                   | ROC-AUC  | PR-AUC  | ROC-AUC  | PR-AUC  | ROC-AUC  | PR-AUC  |
| RDF               | 0.714   | 0.417   | 0.736   | 0.496   | 0.694   | 0.579   |
| LR                | G       | 0.802±0.000 | 0.535±0.000 | 0.822±0.000 | 0.670±0.000 | 0.854±0.000 | 0.769±0.000 |
| MLP               | S       | 0.819±0.003 | 0.594±0.003 | 0.853±0.001 | 0.739±0.004 | 0.868±0.000 | 0.803±0.001 |
| MLP               | G       | 0.863±0.001 | 0.674±0.002 | 0.874±0.001 | 0.761±0.003 | 0.904±0.001 | 0.846±0.002 |
| MLP               | SG      | 0.867±0.001 | 0.667±0.002 | 0.875±0.001 | 0.765±0.002 | 0.904±0.001 | 0.843±0.003 |
| MC                | SG      | 0.868±0.002 | 0.664±0.006 | 0.877±0.003 | 0.768±0.004 | 0.903±0.001 | 0.843±0.002 |
| CFL               | G       | 0.885±0.001 | 0.712±0.002 | 0.905±0.000 | 0.812±0.002 | 0.918±0.001 | 0.870±0.002 |
| CFL               | C       | 0.883±0.001 | 0.708±0.002 | 0.901±0.000 | 0.800±0.001 | 0.919±0.001 | 0.879±0.002 |

S and G indicate the corpus used. S: domain-specific corpus, G: general corpus, SG: both. C means the pre-trained compositional GloVe embeddings are used.

CFL outperforms baselines significantly
⇒ Core-anchored semantic graph and feature aggregation are helpful!
⇒ Domain relevance can be bridged via term relevance!
Comparison to Baselines

|         | Machine Learning | Quantum Mechanics | Abstract Algebra |
|---------|------------------|-------------------|-----------------|
|         | ROC-AUC | PR-AUC | ROC-AUC | PR-AUC | ROC-AUC | PR-AUC |
| LR      | G       | 0.917±0.000 | 0.346±0.000 | 0.879±0.000 | 0.421±0.000 | 0.872±0.000 | 0.525±0.000 |
| MLP     | S       | 0.902±0.001 | 0.453±0.009 | 0.903±0.001 | 0.545±0.004 | 0.910±0.000 | 0.641±0.007 |
| MLP     | G       | 0.932±0.001 | 0.562±0.010 | 0.922±0.001 | 0.587±0.014 | 0.923±0.000 | 0.658±0.006 |
| MLP     | SG      | 0.928±0.001 | 0.574±0.011 | 0.923±0.000 | 0.574±0.007 | 0.925±0.001 | 0.673±0.004 |
| MC      | SG      | 0.928±0.002 | 0.554±0.007 | 0.924±0.001 | 0.590±0.003 | 0.924±0.001 | 0.685±0.005 |
| CFL     | G       | 0.950±0.002 | 0.627±0.013 | 0.950±0.000 | 0.678±0.003 | 0.938±0.001 | 0.751±0.009 |
| HiCFL   | G       | **0.965±0.003** | **0.645±0.014** | **0.957±0.001** | **0.691±0.003** | **0.942±0.002** | **0.769±0.009** |
| LR      | G       | 0.860±0.000 | 0.206±0.000 | 0.788±0.000 | 0.280±0.000 | 0.833±0.000 | 0.429±0.000 |
| MLP     | S       | 0.804±0.003 | 0.144±0.003 | 0.767±0.009 | 0.260±0.005 | 0.804±0.006 | 0.421±0.010 |
| MLP     | G       | 0.836±0.005 | 0.234±0.016 | 0.813±0.006 | 0.295±0.011 | 0.842±0.003 | 0.467±0.011 |
| MLP     | SG      | 0.844±0.003 | 0.230±0.015 | 0.796±0.008 | 0.291±0.011 | 0.839±0.006 | 0.463±0.013 |
| MC      | SG      | 0.852±0.006 | 0.251±0.019 | 0.795±0.014 | 0.303±0.017 | 0.861±0.004 | 0.547±0.006 |
| CFL     | G       | 0.918±0.001 | 0.441±0.009 | **0.897±0.002** | **0.408±0.004** | **0.887±0.002** | **0.563±0.018** |
| HiCFL   | G       | **0.940±0.008** | **0.508±0.026** | **0.897±0.004** | **0.421±0.014** | **0.915±0.002** | **0.648±0.009** |

Hierarchical Learning is helpful!
Comparison to Human Performance

|         | ML-AI    | ML-CS    | AI-CS    |
|---------|----------|----------|----------|
| Human   | 0.698±0.087 | 0.846±0.074 | 0.716±0.115 |
| HiCFL   | **0.854±0.017** | **0.932±0.007** | **0.768±0.023** |

Let humans (5 senior students majoring in CS) and machines judge which term in a query pair is more relevant to ML.

*HiCFL far outperforms human performance!*
The depth of the background color indicates the domain relevance. The darker the color, the higher the domain relevance (annotated by the authors); * indicates the term is a core term, otherwise it is a fringe term.

### Machine Learning (HiCFL)

*Important concepts such as supervised learning, deep learning are ranked very high*

*Terms ranked before 1010th are all good domain-relevant terms*
**Deep Learning (HiCFL, PU Learning)**

Unlabeled positive terms like artificial neural network, generative adversarial network are ranked very high

Terms ranked 101\textsuperscript{st} to 110\textsuperscript{th} are all highly relevant to DL; Terms ranked 1001\textsuperscript{st} to 1010\textsuperscript{th} are related to ML
Conclusion

• We propose to measure *fine-grained domain relevance* — the degree that a term is relevant to a given domain (broad or narrow)

• To handle long-tail terms, we design a novel *core-anchored semantic graph* to bridge domain relevance of terms

• To leverage the graph and domain hierarchy, we propose *hierarchical core-fringe learning*

• To reduce human efforts, we employ *automatic annotation* and *hierarchical positive-unlabeled learning*

• Extensive experiments demonstrate that our methods outperform strong baselines and even surpass professional human performance
Email: jeffhj@illinois.edu
Code and data: https://github.com/jeffhj/domain-relevance

Thanks!