Whois? Deep Author Name Disambiguation using Bibliographic Data

Zeyd Boukhers[1,2] Nagaraj Bahubali Asundi[1]
@ZBoukhers

[1]Institute for Web Science and Technologies (WeST)
University of Koblenz-Landau, Germany
[2] Fraunhofer Institute for Applied Information Technology (FIT)
Sankt Augustin, Germany

Padua, 21 September 2022

Outline

Table of Contents

Author Name Ambiguity Problem

- There are millions of authors[1] sharing a relatively finite set of names.

Why is it a problem?

- It does not allow an accurate calculation of author-level metrics,
- It prevents the continued integrity of bibliographic data in DLs,
- and many more.

[1] As of January 2019, DBLP indexes over 4.4 million publications, published by more than 2.2 million authors.

Author Name Disambiguation

- **Input:** a collection of publications.
- **Goal:** map every author name in each publication to its respective real-world author (using ORCID for example).

Daniel Micciancio, Hao Chen, Statistical Zero-Knowledge Proofs with Efficient Provers, CRYPTO 2003: 282-298

Author Name Disambiguation

Clustering

- The set of publications authored by the same name is clustered w.r.t real-world authors (the most common approach)

Graph-based

- Also unsupervised but based on the relationships between classes such as co-authorship (a trending approach)

Supervised

- Learn from the existing collection(s) to disambiguate the authors names in streaming records.

Table of Contents

Motivation

- Both authors and publishers are getting keener and keener to identify themselves/authors in their publications (using ORCID for example, but
- In EXCITE, we found that the sources of around 60% of the extracted references are missing.
- The author names of the cited publications (i.e. reference section) are still ambiguous.

Tang, J., Qu, M., Wang, M., Zhang, M., Yan, J., Mei, Q.: Line: Large-scale information network embedding. In: Proceedings of the 24th international conference on world wide web. pp. 1067–1077 (2015)

Challenges

- **Homonymy:** authors sharing the same names
  - Hao Chen, Associate Professor from California
  - Hao Chen, Associate Professor from Memphis
- Names substituted by their initials to save space
  - Hao Chen as H. Chen
- Erroneous names due to wrong manual editing
  - Hao Chen as Hoa Chen

Table of Contents

Formulation

- Given $\mathcal{D}$, a collection of N evidence-based bibliographic records, each of which consists of *title* , *source*, $\omega\times$(*real-world author* and the respective *author name*).
- Let Δ be a set of M unique author names shared by $\mathcal{A}$, a set of L unique authors, where $L >> M$
- **Whois's Goal:** given a new record $d^* \notin \mathcal{D}$, link each author name $\in \Delta$ that occurs in d^* to one of the appropriate L authors using *title**, *source**, $\omega^*\times$(*real-world author* and the respective *author name*).

Note

- Each *author name* might refer to one or more *authors* in $\mathcal{A}$
- Each *real-world author* might be referred to by one or two *author names* in Δ

  e.g., Rachid Deriche as Rachid Deriche and R. Deriche

Table of Contents

Whois 1/3

For each author name $\delta_i^* \in \omega^*$:

1. Find the number of real-world authors in $\mathcal{A}$ that might correspond to δ_i^*[2]:
   - $= 0 \Rightarrow \delta_i^*$ refers to a new author $\notin \mathcal{A}$. There is no ambiguity.
   - $= 1 \Rightarrow \delta_i^*$ refers to only one author $\in \mathcal{A}$. There is no ambiguity.
     - It can happen that the author $\notin \mathcal{A}$. Whois does not handle.
   - $> 1 \Rightarrow \delta_i^*$ refers to more than one author. *Whois* comes into play.
     - It can happen that the author $\notin \mathcal{A}$. Whois does not handle.

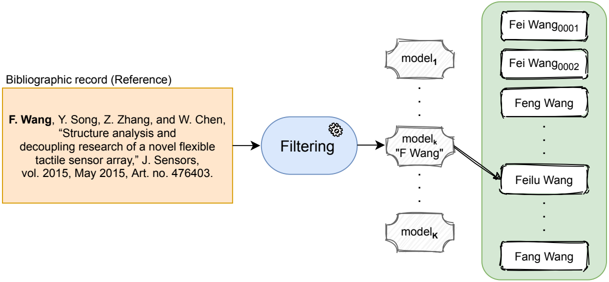
[2] Blocking

Whois 2/3

2. Extract the atomic name variate (ANV) $\overline{\delta_i^*}$ from the author name δ_i^*
   - e.g. *Albert Einstein* $\rightarrow$ *A. Einstein*
3. Let $\overline{\delta_i^*}$ corresponds to $\overline{\delta_k}$ which denotes the *kth* atomic name variate among K possible name variates $\in \mathcal{A}$
4. Pick model $\theta_k \in \Theta = \{\theta'_k\}_{k'=1}^{K}$ to distinguish between all authors $\mathcal{A}_k$ who share the same name variate $\overline{\delta_k}$

Whois 3/3

Figure: An illustration for the task of linking a name mentioned in the reference string with the corresponding DBLP author entity



Author Name Representation

Characteristics

- Author names are specific sequences of characters
- They do not hold any specific semantic nature
- So, encode *author names* based on the order and distribution of characters

Char2Vec

- Uses a fixed list of characters for word vectorization
- Captures the non - vocabulary words and places words with similar spelling closer in the vector space
- Hence, useful when the text consists of abbreviations, typos, etc.

Source and Title Representation

Characteristics

- Title is a meaningful sentence that embeds the specific topic
- Source (e.g. journal names and book titles) can provide a hint about the area of research
- So, capture the context of the sequences of words forming the title and source

BERT

- Provides semantic-based embedding of words

Architecture 1/2

Input

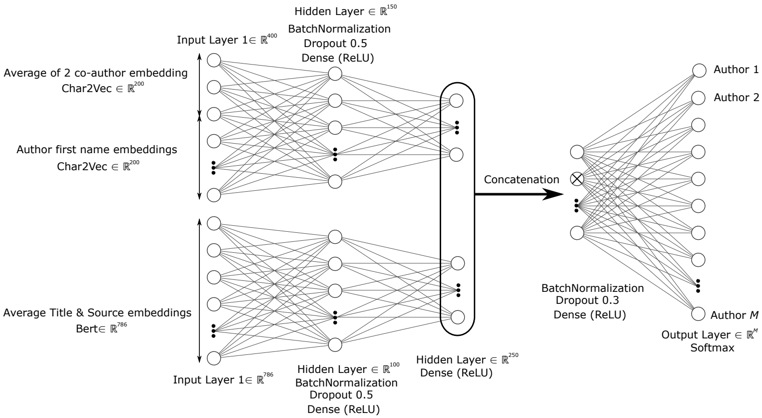
- $\mathrm{x}_1 = \mathrm{char2vec}(\delta_u^{*\text{first-name}}) \bigoplus \frac{1}{2}\left(\mathrm{char2vec}(\delta_p^*) + \mathrm{char2vec}(\delta_j^*)\right)$
- $\mathrm{char2vec}(\mathrm{w}) \rightarrow$ vector of length 200, generated using *Char2Vec* [1]
- $\mathrm{x}_2 = \frac{1}{2}\left(\mathrm{bert}(t^*) + \mathrm{bert}(s^*)\right)$
- $\mathrm{bert}(\mathrm{w}) \rightarrow$ vector of length 786, generated using BERT [2]

Output

- Softmax classifier representing each *author* class

Architecture 2/2

Figure: The architecture of *WhoIs* model



Model Training

For each of the K ANVs $\overline{\delta_k}$

- Given $\mathcal{D}_k \subset \mathcal{D} \rightarrow$ records authored by authors having the ANV $\overline{\delta_k}$
- Generate U_k training samples $\langle \delta_{u_k}, \delta_{u_k,p}, \delta_{u_k,j}, t_{u_{k\mu}}, s_{u_k} \rangle_{u_k=1}^{U_k}$ where $\delta_{u_k,j} \rightarrow$ random co-author name of d_{u_k} or same author name as $\delta_{u_k,p}$
- Convert the sample into $\langle \overline{\delta_{u_k}}, \overline{\delta_{u_k,p}}, \overline{\delta_{u_k,j}}, t_{u_k}, s_{u_k} \rangle$
- So, each bibliographic record is fed into the model $P(\omega, 2)$ times. ω : the number of co-authors $\in d$.
- θ_k is trained on U_k

Prediction

- Let $d^* = \{t^*, s^*, \langle \delta_u^* \rangle_{u=1}^{\omega^*}\}$ be new record
- Generate Y samples $(S_{y=1}^{Y})$ with all pairs of co-author names $\langle \delta_{\text{target}}^*, \delta_p^*, \delta_j^*, t^*, s^* \rangle_{p=1,j=1}^{\omega^*,\omega^*}$ where $Y = P(\omega^*, 2)$
- Feed all samples to the corresponding model θ_μ

  $$a_{\text{target}} = \underset{1\cdots L_\mu}{\operatorname{argmax}} \left(\theta_\mu(S_1) + \theta_\mu(S_2) + \cdots + \theta_\mu(S_Y)\right)$$

Table of Contents

Dataset

DBLP[3]

- 4.4M records as of July 2020
- *Theses* and *Books* are authored by a single author and do not contain a source name → Excluded.
- So, only publications of *Journals* and *Proceedings* are collected
- Statistical details of the used dataset
  - # of records – 5.258.623
  - # of unique authors – 2.665.634
  - # of unique author names – 2.613.577
  - # of unique atomic name variates – 1.555.517

[3] https://dblp.uni-trier.de/xml/

Results

Table: Comparison between *WhoIs* and other baseline methods on CiteSeerX dataset in terms of Macro F1 score as reported in [3]. **ANV** denotes that only atomic name variates were used for all target authors and all their co-authors.

	Macro ALL/ANV	Micro ALL/ANV
WhoIs	0.713 / 0.702	0.873 / 0.861
NDAG [3] (Unsup.)	0.367	N/A
GF [4] (Unsup.)	0.439	N/A
DeepWalk [5] (Unsup.)	0.118	N/A
LINE [6] (Unsup.)	0.193	N/A
Node2Vec [7] (Unsup.)	0.058	N/A
PTE [8] (Semi-Sup.)	0.199	N/A
GL4 [9] (Sup.)	0.385	N/A
Rand [3] (Unsup.)	0.069	N/A
AuthorList [3] (Unsup.)	0.325	N/A
AuthorList-NNMF [3] (Unsup.)	0.355	N/A

Table of Contents

Conclusion & Future Work

Conclusion

- Leveraging co-authorship and domain of expertise using DNN is beneficial for AND task.
- Do we really need to tackle AND as a clustering task while we have -relatively- free ambiguity corpora/indices?

Future Work

- We are introducing *Ambiguity Risk Score* by leveraging the author ethnicity.
- We are capturing the research evolution of the author over time.
- We are using a completely probabilistic approach (Metropolis-Hasting) to disambiguate author names embedded in a Graph.

Thank you!

Questions?

[1] K. Cao and M. Rei, "A joint model for word embedding and word morphology," *arXiv preprint arXiv:1606.02601*, 2016.

[2] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[3] B. Zhang and M. Al Hasan, "Name disambiguation in anonymized graphs using network embedding," in *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, 2017, pp. 1239–1248.

[4] D. Kuang, C. Ding, and H. Park, "Symmetric nonnegative matrix factorization for graph clustering," in *Proceedings of the 2012 SIAM international conference on data mining*, SIAM, 2012, pp. 106–117.

References II

[5] B. Perozzi, R. Al-Rfou, and S. Skiena, "Deepwalk: Online learning of social representations," in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2014, pp. 701–710.

[6] J. Tang, M. Qu, M. Wang, M. Zhang, J. Yan, and Q. Mei, "Line: Large-scale information network embedding," in *Proceedings of the 24th international conference on world wide web*, 2015, pp. 1067–1077.

[7] A. Grover and J. Leskovec, "Node2vec: Scalable feature learning for networks," in *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, 2016, pp. 855–864.

References III

[8] J. Tang, M. Qu, and Q. Mei, "Pte: Predictive text embedding through large-scale heterogeneous text networks," in *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, 2015, pp. 1165–1174.

[9] L. Hermansson, T. Kerola, F. Johansson, V. Jethava, and D. Dubhashi, "Entity disambiguation in anonymized graphs using graph kernels," in *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*, 2013, pp. 1037–1046.