Large-scale author coreference via hierarchical entity representations

Michael Wick MWICK@CS.UMASS.EDU
University of Massachusetts School of Computer Science, 140 Governors Drive, Amherst, MA, 01002

Ari Kobren AKOBREN@CS.UMASS.EDU
University of Massachusetts School of Computer Science, 140 Governors Drive, Amherst, MA, 01002

Andrew McCallum MCCALLUM@CS.UMASS.EDU
University of Massachusetts School of Computer Science, 140 Governors Drive, Amherst, MA, 01002

Abstract

Large-scale author coreference, the problem of ascribing research papers to real-world authors in bibliographic databases, is critical for mining the scientific community. However, traditional pairwise approaches, which measure coreference similarity between pairs of author mentions, scale poorly to large databases; and streaming approaches, which lack the ability to retroactively correct errors, can suffer from chronically low accuracy. In this paper we present a hierarchical model for solving author coreference that overcomes these issues. First, our model enables scalability over rich entity representations by compactly organizing the mentions of each author into trees. Second, we employ Markov chain Monte Carlo (MCMC) inference which is able to retroactively correct existing coreference errors when processing new mentions. We validate these two properties empirically, and demonstrate further scalability through asynchronous parallel MCMC (allowing us to scale to all 150,000,000 author mentions in Web of Science).

1. Introduction

Bibliometric reasoning about academic research, the people who contribute to it, and the organizations (e.g., journals, institutions, grants) that foster its growth, is a current area of high interest because analysis of such data has the potential to revolutionize the way in which scientific research is conducted. For example, if we could predict the next hot research area, or identify researchers in different fields with complementary research directions who should collaborate, or facilitate the hiring process by pairing potential faculty candidates with academic departments, then we could rapidly accelerate and strengthen scientific research. A first step towards making this possible is building a large bibliographic database by extracting *mentions* of papers, authors, journals, and institutions, and perform massive-scale cross document coreference to identify the real-world entities and their relations.

In this paper, we focus on author coreference—the problem of determining which author records (termed *mentions*) in a bibliographic database refer to the same real-world person (termed *entity*). Author coreference is important because it ascribes scientific papers to authors enabling direct bibliometric analysis of the people in the scientific community; however the problem is difficult to solve due to misspellings, alternative abbreviations and common first-initial last name combinations. For example, is the *W. Li* who authored the paper "Reduction of Fe chelated with citrate in an NOx scrubber solution", the same person as the *W. Li* who authored "Supercritical carbon dioxide extraction of essential oil from Cinnamomum migao?" Context in the author records such as the list of co-authors, keywords in the paper title, and publication venue provide crucial evidence for resolving such ambiguity.

Traditionally, this contextual information is exploited via a *pairwise* similarity function, which inputs a pair of mentions and outputs how likely that pair refers to to the same entity (Bagga & Baldwin, 1999; Soon et al., 2001; McCallum & Wellner, 2005; Singla & Domingos, 2005; Bengston & Roth, 2008). For example, in author coreference, the pairwise similarity function might convert each author mention into a

Proceedings of the 30^{th} International Conference on Machine Learning, Atlanta, Georgia, USA, 2013. JMLR: W&CP volume 28.

bag-of-words words representation, and then measure the coreference similarity between mentions as a cosine similarity between those mentions' bags. Although pairwise models of coreference have been the dominant approach for years, they suffer two serious drawbacks. First, pairwise models lack scalability because there are a quadratic number of similarity functions. Second, these models lack representational power because they only model properties of mention-pairs and not the entities themselves.

We present a discriminative model that addresses these issues by (1) reducing the number of similarity functions via a more efficient model structure which compresses mentions and (2) explicitly modeling the entities as first-class citizens (e.g., with a full-set of attributes) in the model. In particular our model organizes each author entity into a tree (Wick et al., 2012). Nodes in an entity tree contain attributes relevant to that entity: non-leaf node attributes are derived from their children, and leaf-node attributes are extracted from the entity's mentions. Intermediate nodes in the graph may thus compress entire sub-trees of mentions (for example, a node may summarize 100 contextually similar *L. Rabiner* mentions). Instead of measuring coreference compatibility between mention-pairs (McCallum & Wellner, 2003; Singh et al., 2011), our model measures coreference compatibility between a node and its parent. We employ temperature regulated Markov chain Monte Carl (MCMC) to solve hierarchical coreference by proposing changes to the trees with the goal of maximizing these compatibility scores.

Empirically, we demonstrate that our model is much more efficient than the pairwise model, achieving orders of magnitude speed-ups at the same-level of accuracy. We further show that our approach can scale to large datasets such as DBLP (5 million authors) using just a single processor with one core, and even larger datasets such as PubMed (60,000,000 million authors), and Web of Science (150,000,000 authors) through asynchronous parallelization across multiple machines.

2. Author Coreference

In order to better understand why pairwise approaches are not adequate for solving large scale author coreference, it is useful to express them as a graphical model (e.g., factor graph). In this representation, author mentions are observed variables, the decisions as to whether or not two mentions are coreferent are encoded in binary prediction variables, and the pairwise similarity functions are log factors that input two mentions with a setting to their binary decision variable, and output their similarity score (e.g., cosine between bags) if the setting to the decision variable is 1 and 0 otherwise.[1] We depict a pairwise model instantiated on six author mentions in Figure 1 (black boxes are factors, open circles are the binary decision variables, and shaded circles are the mentions).

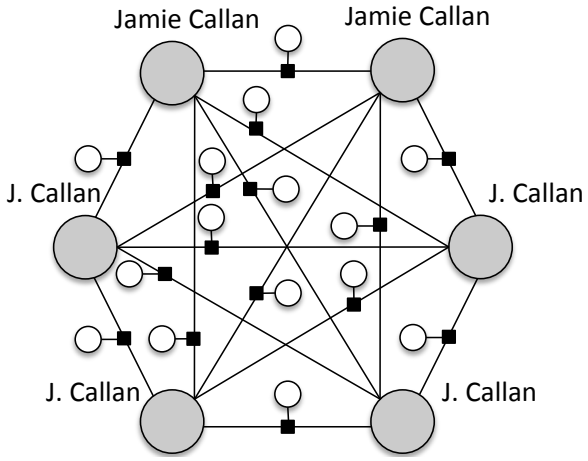


Figure 1: **Pairwise model on six mentions:** Open circles are the binary coreference decision variables, shaded circles are the observed mentions, and the black boxes are the f actors of the graphical model that encode the pairwise compatibility functions.

Coreference is solved by searching for a setting (to the quadratic number) of decision variables that has the highest probability (usually subject to a transitivity constraint that ensures a consistent setting of these variables). While in theory the solution to this problem is NP-hard, in practice, local search techniques such as temperature-regulated MCMC have proven to be effective. Unfortunately, even MCMC becomes intractable as the clusters get larger. For example, the MCMC step of moving a mention from one entity to another requires evaluating a linear number of factor functions (linear in the number of mentions referring to the two entities). For large datasets in which the average size of each entity is likely to be large, the MCMC computational costs are expensive.

2.1. Hierarchical coreference

In contrast to the pairwise model in which every pair of mentions is connected by a factor, our hierarchical model is structured as a tree with mentions at the leaves. For our model, we introduce random variables $e_i \in E$ that directly represent the attributes of

[1] In general, the factor could output a different (nonzero) score if the setting to the variable is 0.

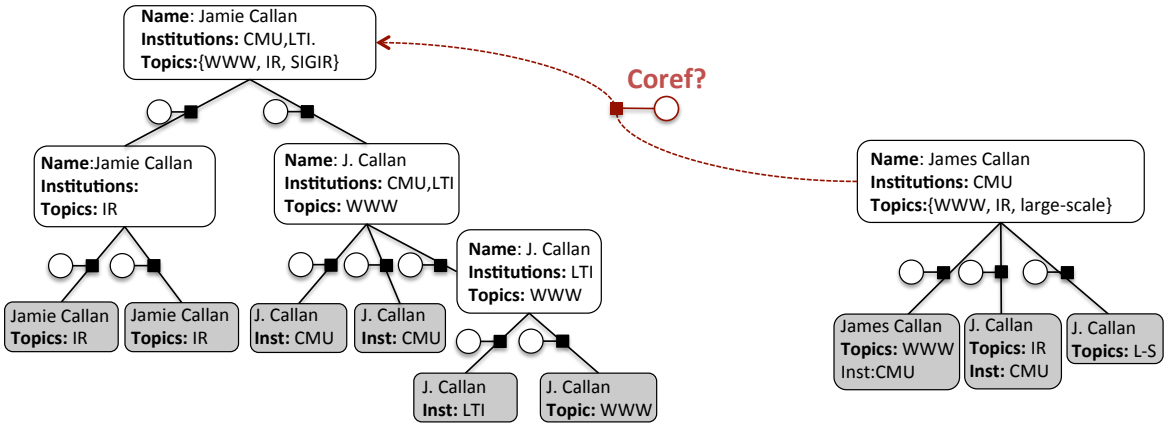


Figure 2: **Discriminative hierarchical factor graph for coreference** instantiated on two entities. The entity on the left consists of the same six mentions for which the pairwise model in Figure 1 is instantiated, and the entity on the right consists of three additional mentions. Note how deciding whether or not to merge these two entities requires only evaluating a single factor (shown in red).

entities, decision variables y_{ij} that encode whether a mention $m_j \in M$ (a "child") is coreferent to entity e_i (a "parent"), and factors that measure the coreference compatibility between a parent entity, e_i and its child mention, m_j. Despite having the same set of attribute types as the mention variables (e.g., first name, last name, bags of co-authors, bags of topics), the attributes of an entity are not observed, but rather, are inferred from the attributes of their children. Our model structure is recursive in the sense that entity variables may be the parents of other entity variables (and the same types of factors exist to measure their coreference compatibility). We also include factors over each node in the tree measuring the compatibility of that node's attributes (recall that those attributes have been inferred from that node's descendants) and can be thought of as a prior over nodes. Even with these extra nodes and factors, our model still requires fewer factor evaluations than the pairwise model and thus enables fast MCMC inference. We describe the specific factors of our model in more detail in Section 2.3.

The structure of our model yields the following interpretation: the root of each tree is the canonical "entity," intermediate nodes are "subentities," and the observed mentions are leaves. Singleton mentions (which are not coreferent with any other mentions) are represented as trivial trees of size one. Our model requires that each node (either mention, subentity or entity) has either no children or two or more children (to avoid linear chains). In constructing the tree, we infer a parent's attributes to be an aggregate of the attributes of its children.

2.2. Inference

We employ low-temperature Markov chain Monte Carlo (MCMC) to search for a set of entity trees that have high probability under the hierarchical coreference model. MCMC explores the space of coreference trees by suggesting local modification to an existing coreference hypothesis. For example, MCMC might propose to (1) infer the existence of a new entity, (2) delete an entity, or (3) move a subtree of mentions from one entity to another. Such proposals then have a chance of being accepted by the model. The probability with which the model accepts one of these proposals is equal to the likelihood ratio of the proposed hypothesis to the current hypothesis. The variant of MCMC which we employ is described in more detail in earlier work (Wick et al., 2012).

2.3. Hierarchical Author Coreference Features

In author coreference, the author mentions are often extractions from the headers and bibliography sections of research papers. Thus, each mention is associated with a paper title, a list of co-authors, a publication venue, a date, a list of email addresses, a list of institutions, and sometimes even domain-specific keywords. In our model, these fields are represented using four bags- of-words. In particular, the co-author bag contains co-author first-initial-last names. The venue bag the tokens in the publication field, and the keywords bag contains a combination of keywords, institutions, and email addresses. The bag of topics contains a list of topics inferred by running latent Dirichlet allocation (Blei et al., 2003) on the paper titles and venues from DBLP; only topics with probability greater than 0.1

are included in the bag.

The factors in our model measure the compatibility of these bags. Our model includes child-parent factors that encourage measure a cosine similarity between a child and parent's bag-of-words. Intuitively, these factors encourages children to have similar topics/co-authors/venues as their parent. Further, we penalize the entropy of each bag-of-words. Intuitively, these factors encourage the model to discover authors who research small numbers of topics, or collaborate with small numbers of co-authors. Finally, our model includes priors on the structure of the tree (to encourage or discourage the existence of entities or subentities, and control the depth and bushiness of the trees). Our comprehensive set of coreference factors are in Table 2, and are instantiations of the generic hierarchical factor templates provided in Table 1.

2.4. Parallelization

In order to parallelize author coreference, we employ "blocking" (Hernández & Stolfo, 1995; McCallum et al., 2000) which partitions the author mentions into disjoint sets. We assume that mentions in different blocks do not refer to the same entity thereby allowing for asynchronous parallelization. Our system blocks mentions based on a normalized concatenation of their first-initial last name. For example, the entities "Francisco C. Pereira," "F.C.N. Pereira," and "Fernando Pereira" would all be in the same block.

Our asynchronous parallelization algorithm operates as follows. First, we store all the mentions in a database (indexed by their block). Then, we assign each block to an inference worker and add that worker to a worker queue. We maintain the following property during inference: if there are fewer than k active workers, we pop a worker off the queue and run it in parallel. The worker (1) reads its mentions from the DB using an assigned block key, (2) performs coreference on the block, (3) writes the results back to the DB, and (4) sets its status to inactive.

3. Experiments

3.1. Hierarchical vs pairwise coreference models

In this section we compare our hierarchical coreference model to the pairwise coreference model in order to assess our model's scalability. We use a publicly available collection of 4,394 BibTeX files containing 817,193 entries[2] from which we extract 1,322,985 author mentions. In addition we include 2,833 labeled mentions from REXA[3] for evaluation.

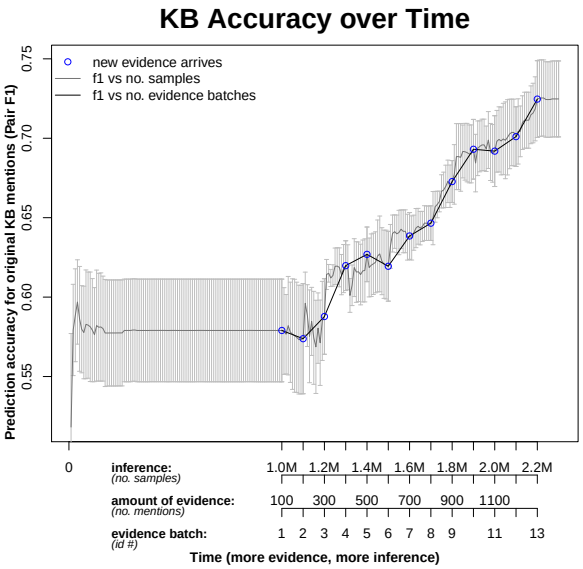


Figure 4: Incorporation of additional author mentions improves coreference accuracy of old author mentions.

In Figure 3a we plot the number of samples (MCMC proposals) versus time. Notice that the pairwise model becomes increasingly slower with time. This is due to the fact that we initialize both models to singleton clusters, and as the size of each cluster grows, the cost of evaluating each sample in the pairwise model increases. In contrast, the sampling rate of the hierarchical model remains relatively high, even as the size of the clusters increase. In figure 3b, we plot the accuracy versus time. We can see that because of the higher sampling rate that the hierarchical model is able to achieve higher accuracy levels faster than the pairwise model.

3.2. Hierarchical MCMC vs streaming inference

One way of achieving scalability to large numbers of author mentions is to employ a *streaming* inference algorithm which is only able to visit each mention once. Thus, streaming coreference algorithms inherently lack the ability to reconsider previous inference decisions when new data arrives. As a result, these methods are unable to retroactively rectify coreference errors when mentions containing new evidence become available. In this experiment, we study the impact of new data on the accuracy of coreference and show that there is

[2] http://www.iesl.cs.umass.edu/data/bibtex

[3] http://www2.selu.edu/Academics/Faculty/aculotta/data/rexa.html

factor type	input (variables)	parameters	output (log score)
BoW cosine similarity	parent bag (p), child bag (c)	w, t	$w \log(\|c\|_1 + 2) \left(\frac{(p-c) \cdot c}{\|p-c\|_2 \|c\|_2} + t \right)$
entity existence penalty	node (e)	w	$-w \mathbb{1}\{\text{isRoot}(e)\}$
subentity existence penalty	node (e)	w	$-w \mathbb{1}\{\neg\text{isRoot}(e) \wedge \neg\text{isLeaf}(e)\}$
BoW norm. entropy penalty	node's bag-of-words (b)	w	$-w \frac{H(b)}{\log \|b\|_0}$
BoW complexity penalty	node's bag-of-words (b)	w	$-w \frac{\|b\|_0}{\|b\|_1}$
names penalty	node's bag-of-names (b)	w	$-\min\left(w(\|b\|_0 - 1)^2, -16\right)$

Table 1: Precise definitions for factors in the hierarchical coreference model. We assume a sparse-vector representations for a bag of words (b), $|b|_n$ is the l_n norm of bag b, $H(b)$ is the Shannon-entropy of bag b, $\mathbb{1}\{\text{formula}\}$ is an indicator function.

factor type	inputs	bag-of-words	weights (w,t)	dynamic range
BoW cosine similarity	parent bag, child bag	topics	$w = 8$, $t = -0.25$	$\log n[-2, 6]$
BoW cosine similarity	parent bag, child bag	co-authors	$w = 4$, $t = -0.125$	$\log n[-0.5, 3.5]$
BoW cosine similarity	parent bag, child bag	venues	$w = 4$, $t = -0.25$	$\log n[-1, 3]$
BoW cosine similarity	parent bag, child bag	keywords	$w = 2$, $t = 0$	$\log n[0, 2]$
entity existence penalty	root node (entity)	—	-1.0	$\{0, -1.0\}$
subentity existence penalty	interm. node (subentity)	—	-0.5	$\{0, -0.5\}$
BoW norm. entropy penalty	node	topics	0.5	$[-0.5, 0]$
BoW complexity penalty	node	co-authors	2	$[-2.0, 0)$
BoW complexity penalty	node	venues	1	$[-1.0, 0)$
names penalty	root node (entity) bag	first names	1	$\{-16, -9, -3, 0\}$
names penalty	root node (entity) bag	first initials	1	$\{-16, -9, -3, 0\}$
names penalty	root node (entity) bag	middle names	1	$\{-16, -9, -3, 0\}$
names penalty	root node (entity) bag	middle initials	1	$\{-16, -9, -3, 0\}$
names penalty	root node (entity) bag	last name	∞	$\{-\infty\}$

Table 2: The comprehensive set of factor templates in our hierarchical coreference model. See Table 1 for scoring functions.

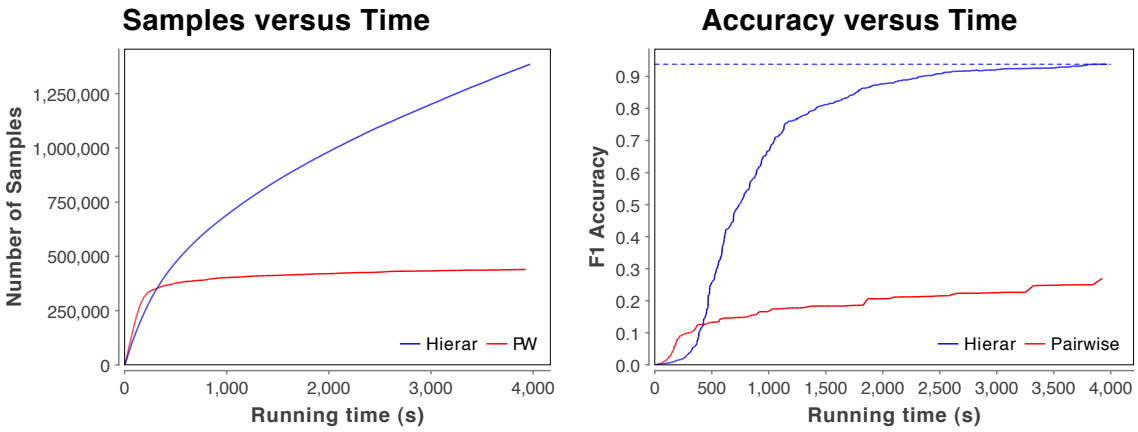
indeed a cost included in employing streaming algorithms for coreference.

In Figure 4 we demonstrate adding more mentions to the bibliographic database actually improves the coreference accuracy of the original bibliographic database. For this experiment, we first randomly divide our labeled dataset in half—the first half we term the *initial data* and second half we term the *supplemental data*. Next, we initialize our database to contain only the initial data (every mention is a singleton), and run our MCMC-based author coreference algorithm. We record the accuracy of the initial data every 1000 samples. After one million samples, we begin adding additional mentions from the supplemental data (in the plot, every blue dot shows the addition of more mentions). Notice that the addition of each new batch of supplementary data improves the accuracy of the initial data. This is because our MCMC-based coreference algorithm is able to revisit previous coreference decisions and incorporate the new data; in contrast, a streaming algorithm would stagnate and suffer from poor accuracy.

For this experiment, we use a more ambiguous version of the REXA author coreference dataset (Culotta et al., 2007). This dataset contains 1,459 automatically extracted paper citations and 4,370 automatically extracted author mentions (1,459 of which are manually labeled with author coreference groundtruth). Each of the 1,459 labeled mentions belongs to one of eight common first-initial-last-name combinations: *D. Allen, A. Blum, S. Jones, L. Lee, J. McGuire, A. Moore, H. Robinson, S. Young.*

3.3. Massive-scale coreference

We have been able to scale our coreference system to up to 150 million author mentions from web of science. Table 3 summarizes some large databasets on which we have performed coreference. Note that we compiled the data for this table from disparate runs spanning the course of a year. Different computational resources were utilized in each run, and further



(a) Samples vs. time (1.3 million mentions)

(b) Accuracy vs. time (1.3 million mentions)

Figure 3: Hierarchical vs pairwise models of author coreference.

dataset	#mentions	#cpus	# cores	inference running time
DBLP	5 million	1	1	5 hours
REXA	20 million	3	48	1 day
PubMed	60 million	1	24	2 days
WoS	147 million	3	48	2 days

Table 3: Large scale author coreference.

the computational resources are shared; thus reported run times are subject to wide variation. Reported inference time does not include time spent reading and writing the data to the DB.

4. Related Work

Recent work has demonstrated that modeling entities hierarchically improves scalability of coreference. For example, fixed depth entity hierarchies with sub-entities and super-entities has been shown to greatly improve the efficiency of inference in models with pairwise factors between mentions (Singh et al., 2011). We are able to achieve further scalability by removing these pairwise factors, and increase the modeling power by allowing tree depth to be arbitrary.

Other work has achieved scalability by averaging all the mention feature vectors in each entity (Rao et al., 2010; Levin et al., 2012). The former combines this technique along with streaming algorithms to scale to millions of mentions, and the latter applies this technique in the context of agglomerative clustering to scale to 54 million author mentions in Web of Science. Our model also compresses the mentions in an entity; however, rather than simply averaging the mentions, we directly model the author attributes and compress them using a tree. As a result, we are able to provide both scalability (150 million author mentions) and increased representational power: we are able to provide the advantages of recently proposed entity-based coreference systems that are known to provide higher accuracy (Haghighi & Klein, 2007; Culotta et al., 2007; Yang et al., 2008; Wick et al., 2009; Haghighi & Klein, 2010).

5. Acknowledgements

This work was supported in part by the Center for Intelligent Information Retrieval and in part by the National Science Foundation under Grant No. S12100000021 and in part by IARPA via DoI/NBC contract #D11PC20152 in part by DARPA under agreement #FA8750-13-2-0020. The U.S. Government is authorized to reproduce and distribute reprint for Governmental purposes notwithstanding any copyright annotation thereon. Any opinions, findings and conclusions or recommendations expressed in this material are the authors and do not necessarily reflect those of the sponsor.

6. Conclusion

In this paper we presented a scalable author coreference system using a hierarchical model of coreference with asynchronous parallel MCMC for inference. We demonstrated that our model is more scalable than pairwise approaches while also being more accurate than streaming approaches. Finally, we demonstrated scalability to various datasets including 150 million authors on Web of Science. We hope our techniques will help enable the creation of large bibliographic knowledge bases and provide the foundation for rich entity-based bibliometrics.

References

Bagga, Amit and Baldwin, Breck. Cross-document event coreference: annotations, experiments, and observations. In *Proceedings of the Workshop on Coreference and its Applications*, CorefApp '99, pp. 1–8, Stroudsburg, PA, USA, 1999. Association for Computational Linguistics. URL http://dl.acm.org/citation.cfm?id=1608810.1608812.

Bengston, Eric and Roth, Dan. Understanding the value of features for coreference resolution. In *Empirical Methods in Natural Language Processing (EMNLP)*, 2008.

Blei, David M., Ng, Andrew Y., and Jordan, Michael I. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3: 993–1022, March 2003. ISSN 1532-4435. URL http://dl.acm.org/citation.cfm?id=944919.944937.

Culotta, Aron, Kanani, Pallika, Hall, Robert, Wick, Michael, and McCallum, Andrew. Author disambiguation using error-driven machine learning with a ranking loss function. In *Sixth International Workshop on Information Integration on the Web (IIWeb-07)*, Vancouver, Canada, 2007. URL http://www2.selu.edu/Academics/Faculty/aculotta/pubs/culotta07author.pdf.

Haghighi, Aria and Klein, Dan. Unsupervised coreference resolution in a nonparametric bayesian model. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics*, pp. 848–855, Prague, Czech Republic, June 2007. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/P07-1107.

Haghighi, Aria and Klein, Dan. Coreference resolution in a modular, entity-centered model. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, HLT '10, pp. 385–393, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics. ISBN 1-932432-65-5. URL http://dl.acm.org/citation.cfm?id=1857999.1858060.

Hernández, Mauricio A. and Stolfo, Salvatore J. The merge/purge problem for large databases. In *Proceedings of the 1995 ACM SIGMOD international conference on Management of data*, SIGMOD '95, pp. 127–138, New York, NY, USA, 1995. ACM. ISBN 0-89791-731-6. doi: http://doi.acm.org/10.1145/223784.223807. URL http://doi.acm.org/10.1145/223784.223807.

Levin, Michael, Krawczyk, Stefan, Bethard, Steven, and Jurafsky, Dan. Citation-based bootstrapping for large-scale author disambiguation. *JASIST*, 63 (5):1030–1047, 2012.

McCallum, A. and Wellner, B. Toward conditional models of identity uncertainty with application to proper noun coreference. In *IJCAI Workshop on Information Integration on the Web*, 2003.

McCallum, Andrew and Wellner, Ben. Conditional models of identity uncertainty with application to noun coreference. In Saul, Lawrence K., Weiss, Yair, and Bottou, Léon (eds.), *NIPS17*. MIT Press, Cambridge, MA, 2005.

McCallum, Andrew K., Nigam, Kamal, and Ungar, Lyle. Efficient clustering of high-dimensional data sets with application to reference matching. In *Proceedings of the Sixth International Conference On Knowledge Discovery and Data Mining (KDD-2000)*, Boston, MA, 2000.

Rao, Delip, McNamee, Paul, and Dredze, Mark. Streaming cross document entity coreference resolution. In *COLING (Posters)*, pp. 1050–1058, 2010.

Singh, Sameer, Subramanya, Amarnag, Pereira, Fernando C. N., and McCallum, Andrew. Large-scale cross-document coreference using distributed inference and hierarchical models. In *ACL*, pp. 793–803, 2011.

Singla, Parag and Domingos, Pedro. Discriminative training of Markov logic networks. In *AAAI*, Pittsburgh, PA, 2005.

Soon, Wee Meng, Ng, Hwee Tou, and Lim, Daniel Chung Yong. A machine learning approach to coreference resolution of noun phrases. *Comput. Linguist.*, 27(4):521–544, 2001. ISSN 0891-2017.

Wick, Michael, Culotta, Aron, Rohanimanesh, Khashayar, and McCallum, Andrew. An entity-based model for coreference resolution. In *SIAM International Conference on Data Mining*, 2009. URL http://www2.selu.edu/Academics/Faculty/aculotta/pubs/wick09entity.pdf.

Wick, Michael, Singh, Sameer, and McCallum, Andrew. A discriminative hierarchical model for fast coreference at large scale. In *Proc. ACL*, 2012.

Yang, Xiaofeng, Su, Jian, Lang, Jun, Tan, Chew Lim, Liu, Ting, and Li, Sheng. An entity-mention model for coreference resolution with inductive logic programming. In *Association for Computational Linguistics*, pp. 843–851, 2008.