Cleaning the Spurious Links in Data

Mong Li Lee, Wynne Hsu, and Vijay Kothari, *National University of Singapore*

Data quality problems can arise from abbreviations, data entry mistakes, duplicate records, missing fields, and many other sources. These problems proliferate when you integrate multiple data sources in data warehousing, federated databases, and global information systems. Data warehouses load and frequently update large amounts of data from heterogeneous sources, further increasing the likelihood of introducing errors. When mining these data warehouses for decision-making information, making logical and well-informed decisions becomes problematic if the data's quality is doubtful.

Comparing context information between data records can help solve the data quality problem of spurious links—that is, multiple links between data entries and real-world entities.

Data cleaning refers to processes for detecting and removing errors and inconsistencies from data. Research in this area has included outlier detection,[1] noise handling for classification,[2] and duplicate elimination.[3,4] Given the "garbage in, garbage out" principle, clean data is crucial for database integration, data warehousing, and data mining, thus leading to data cleaning's strong association with these research areas.

A newly discovered class of erroneous data is *spurious links*, where a real-world entity has multiple links that might not be properly associated with it. Standard duplicate elimination techniques such as the sorted-neighborhood method[3] or the priority-queue algorithm[5] can't find this anomaly. The existence of such spurious links often leads to confusion and misrepresentation in the data records representing the entity. Consider the Digital Bibliography & Library Project database (http://dblp.uni-trier.de), a large collection of computer science bibliographical records. Each DBLP record captures a publication's list of authors, title, conference or journal name, and page numbers. Although the data set is well known for its high-quality bibliographic information, collecting and maintaining the data from diverse sources requires enormous effort. Errors, including spurious links, are inevitable.

To solve this problem, we use context information to identify spurious links. First, we identify data records that contain potential spurious links. We then determine the set of attributes that constitute each record's context. For example, the coauthors and conference/journal fields constitute a publication record's context and offer good hints for identifying potential spurious links. Experiments with three real-world databases have demonstrated that our approach can accurately identify spurious links.

A context-based solution

Figure 1 gives an overview of our approach. After finding records that might contain spurious links, this approach uses selected fields that capture the records' context to compute their context similarity. These context fields might have access to additional domain knowledge that provides a more comprehensive picture of the records' context. We developed a method to determine context similarity between two record sets, and we use their overall degree of similarity to subsequently identify any spurious links.

Table 1 shows a pair of records from the DBLP database. At first glance, Record 2 seems to contain a typographical error in the author name "Tshiharu Hasegawa," compared with "Toshiharu Hasegawa" in Record 1. The possibility that "Tshiharu Hasegawa" and "Toshiharu Hasegawa" refer to the same real-world person increases if we have this context information:

- They have some common coauthors, such as "Tetsuya Takine" and "Yutaka Takahashi."
- Their publication titles show similar research areas. For example, both the publication titles in Records

1094-7167/04/$20.00 © 2004 IEEE
Published by the IEEE Computer Society

Authorized licensed use limited to: Sterne Kessler Goldstein Fox. Downloaded on June 04,2024 at 05:04:36 UTC from IEEE Xplore. Restrictions apply.

Roku EX1016
IPR2024-01053
U.S. Patent No. 8,296,791

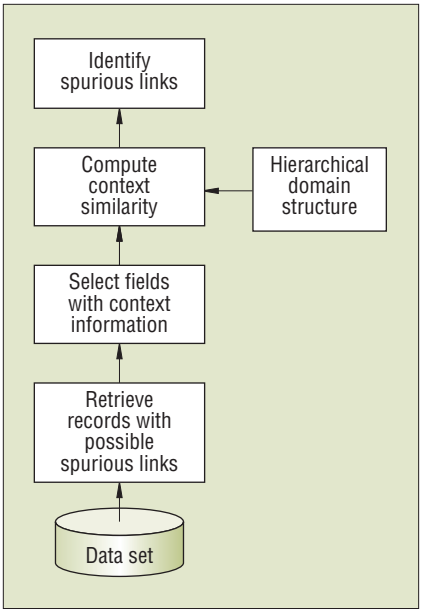

Figure 1. Context-based cleaning of spurious links.

1 and 2 are related to networking research.

- The conference proceedings or journals in which they publish have overlapping research areas. For example, the scope of "Performance" overlaps with that covered in "Performance Evaluation."

Records with potential spurious links

First, we need to identify the records with potential spurious links. We use existing string similarity matching algorithms to identify record pairs with a high degree of similarity in some attribute. For example, Table 1 shows a pair of records with high similarity for one of the coauthors, "Toshiharu Hasegawa" and "Tshiharu Hasegawa." On the basis of these high similarity values, we retrieve the associated records.

Tables 2 and 3 show extracts from the sets of records retrieved from the DBLP Computer Science Bibliography using "Toshiharu Hasegawa" and "Tshiharu Hasegawa," respectively.

Context attributes

Next, we identify the attributes that might provide clues about the existence of spurious links. A *spurious attribute* contains erroneous values that might result in spurious record links. A *context attribute* contains values strongly correlated with those in the spurious attribute.

To identify the context attributes, we apply association-rules mining to the database to discover all associations among the attribute values. That is, we try to determine if a value in one attribute frequently occurs with some value in another attribute. We're particularly interested in rules whose antecedent contains the spurious attribute. If we obtain many rules that associate values between the spurious attribute and some other set of attributes, we consider the latter to be candidate context attributes. For example, for the DBLP data set, we found many rules with high confidence and minimum support (a user-defined threshold to retain only the rules satisfied by a reasonable number of data tuples) involving authors and their coauthors. An attribute can be a context attribute even if we don't obtain high-confidence association rules. For instance, the values corresponding to the Proceedings attribute don't

Table 1. A possible pair of duplicates in the Digital Bibliography & Library Project data set.

Record	Author list	Title	Proceedings
1	Tetsuya Takine, Yutaka Takahashi, *Toshiharu Hasegawa*	Analysis of an Asymmetric Polling System with Single Buffers	Performance
2	A. Sugahara, Tetsuya Takine, Yutaka Takahashi, *Tshiharu Hasegawa*	Analysis of a Nonpreemptive Priority Queue with SPP Arrivals of High Class	Performance Evaluation

Table 2. Extracted records using "Toshiharu Hasegawa."

Record	Author list	Title	Proceedings
1	Shojiro Muro, Toshihide Ibaraki, Hidehiro Miyajima, Toshiharu Hasegawa	File Redundancy Issues in Distributed Database Systems	VLDB
2	Kazuhiro Ohtsuki, Yutaka Takahashi, Toshiharu Hasegawa	Analysis for Traverse Time in an Integrated Communication Network	ICC
3	Shojiro Muro, Toshihide Ibaraki, Hidehiro Miyajima, Toshiharu Hasegawa	Evaluation of File Redundancy in Distributed Database Systems	TSE
4	Tetsuya Takine, Yutaka Takahashi, Toshiharu Hasegawa	Analysis of Asymmetric Polling System with Single Buffers	Performance

Table 3. Extracted records using "Tshiharu Hasegawa."

Record	Author list	Title	Proceedings
1	Tetsuya Takine, Hideaki Takagi, Yutaka Takahashi, Tshiharu Hasegawa	Analysis of Asymmetric Single-Buffer Polling and Priority of Systems without Switchover Times	SIGMETRICS
2	Yoshiyuki Shiozawa, Tetsuya Takine, Yutaka Takahashi, Tshiharu Hasegawa	Analysis of a Polling System with Correlated Input	Computer Networks & ISDN Systems
3	Fumio Ishizaki, Tetsuya Takine, Tshiharu Hasegawa	Analysis of a Discrete-Time Queue with Gated Priority	Performance Evaluation
4	Shoji Kasahara, Yutaka Takahashi, Tshiharu Hasegawa	Analysis of Waiting Time of M/G/1/K System	Performance Evaluation

Authorized licensed use limited to: Sterne Kessler Goldstein Fox. Downloaded on June 04,2024 at 05:04:36 UTC from IEEE Xplore. Restrictions apply.

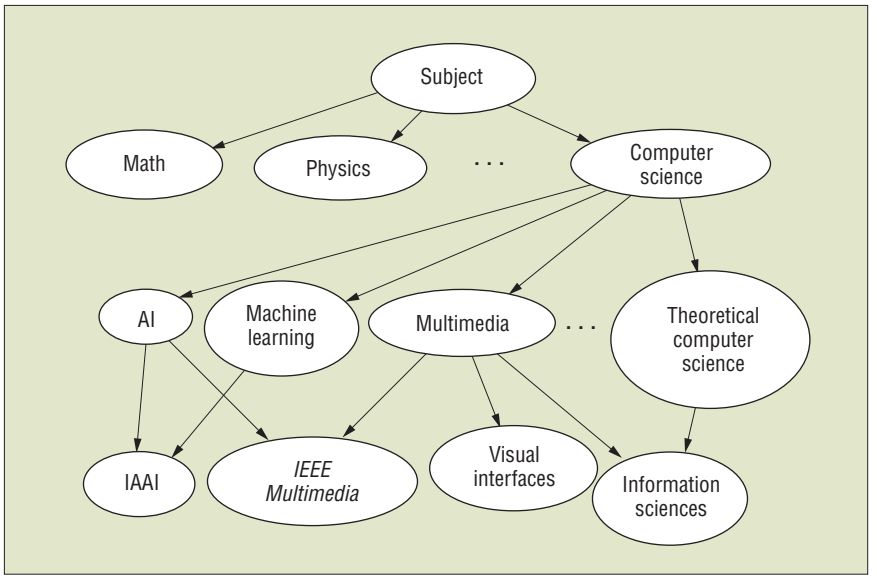

Figure 2. A concept hierarchy for research articles' conference/journal field.

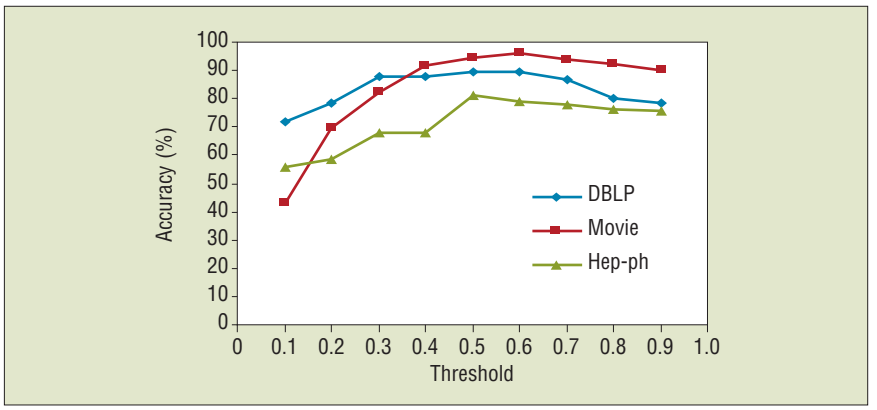

Figure 3. The threshold value's effect on accuracy.

appear in any high-confidence association rule satisfying the minimum support. However, if we generalize the Proceedings attribute values to the research area, we discover high-confidence rules involving Author and Research Area, and we can use the latter as a context attribute. We use the concept hierarchies to generalize attribute values[6] so that we don't miss such context attributes.

Figure 2 shows a concept hierarchy constructed for research articles' conference/journal field.

On the basis of this rationale, we apply this algorithm to determine the potential context attributes:

Algorithm FindContextAttr

1. If there exist attributes with a concept hierarchy Then generalize the attribute values.
2. Generate association rules for the database given user-specified minimum support and minimum confidence.
3. Identify the rules that contain spurious attributes in their antecedents.
4. Attributes in the consequent of these rules constitute the context attributes.

Record similarity

We now determine the retrieved records' similarity to find spurious links. Given two sets of records and a list of context attributes, we must decide if the two sets refer to the same real-world entity. We do this by determining how similar their context attributes are.

Our study indicates that for identifying spurious links, a simple, yet effective column-wise similarity measure achieves the best accuracy. Essentially, the column-wise similarity measure for a context attribute C would take the union of the values of attribute C in each set of records and find the degree of overlap according to the formula

$$ColSim_C(A_1, A_2) = |A_1 \cap A_2| / \min(|A_1|, |A_2|),$$

where A_1 is the set of unique values for the context attribute C in the first table of records, and A_2 is the set of unique values for C in the second table of records.

Referring to the author list attribute in Tables 2 and 3, the two sets of attribute values have a similarity value of 4/6 = 0.67.

Given a set of context attributes $C_1, C_2, \ldots, C_n$, we find the overall context similarity of the record sets R_1 and R_2 using

$$ContextSim(R_1, R_2) = \sum_{i=1}^{n} w_i * ColSim_{ci},$$

where w_i is the weight of context attribute C_i, and $w_1 + \ldots + w_n = 1$. If most context attributes exhibit similarity above some predetermined threshold, we can infer that the records likely refer to the same person in the real world. Therefore, a spurious link exists.

Optimal threshold values

In our experiments (which we describe in the next section), we determined the optimal threshold at which our approach achieves maximum accuracy in identifying spurious links. We defined the accuracy metric as the ratio of correctly identified duplicate pairs to the total number of potential duplicate pairs. Figure 3 shows how the threshold value affects data set accuracy when we assign equal weights to the context attributes.

Testing the approach

We implemented our algorithms in Java and ran the experiments on a Pentium 4 1.6-GHz system with 256 Mbytes of RAM running Windows XP Professional. We used IntelliClean[4] on the three data sets to find record sets with high similarity in the spurious attribute, and we used DM2-CBA[7] to generate association rules to determine the context attributes.

DBLP

The DBLP database consists of over 200,000 records, each stored in an XML file. We used a subset of the data to create a relational database of 12,258 records. Each record in the relational database contains information about the publication's first author, coauthors, title, and conference or journal (called

Authorized licensed use limited to: Sterne Kessler Goldstein Fox. Downloaded on June 04,2024 at 05:04:36 UTC from IEEE Xplore. Restrictions apply.

"proceedings" in the database). IntelliClean output 305 pairs of records with high similarity in their First Author attribute. Table 4 shows a sample of the records found and the degree of string similarities.

Our algorithm for finding context attributes in the data set outputs the coauthor and proceedings fields. First, using the concept hierarchy in Figure 2, we generalize the proceedings field to the research areas the publications belong to. We generate association rules for the database having a minimum confidence of 75 percent and a minimum support of at least two records. We consider only those rules that contain the First Author attribute values in their antecedent. We generalize these rules to the corresponding attributes and obtain two associations: First Author → Coauthor, and First Author → Proceedings. These rules indicate that strong associations exist between the values in the first author field and those in the coauthor and proceedings fields. So we use the Coauthor and Proceedings attributes as context attributes.

For each pair of duplicate records, we use the first authors' names to retrieve records with the same name from the database. If we've identified "name1" and "name2" as possible duplicates, we can then use these values to retrieve two record sets from the database. Set A consists of records whose first author is "name1," and Set B consists of records whose first author is "name2." We then apply a column-wise similarity method to find the similarity between the two sets.

Assigning equal weights to the context attributes and using the optimal threshold value of 0.5, we achieved an accurate detection rate of 89 percent for 50 pairs of duplicates. Table 5 shows a sample of the results for the DBLP data set. A high-context similarity value indicates that spurious links exist; a low-context similarity value indicates no spurious links.

Table 4. Sample duplicates from the DBLP data set.

Name 1	Name 2	Similarity
Toshiharu Hasegawa	Tshiharu Hasegawa	0.972
Patricia A. Jacobson	Patricia A. Jacobs	0.979
Kenny Wong	Ken Wong	0.950

Table 5. Sample results for the DBLP data set.

Name 1	Name 2	Context similarity value	Actual match?
Steven Minton	Steve Minton	0.775	Yes
Sangjin Lee	Sang-Jin Lee	0.710	Yes
Yin-Feng Xu	Yinfeng Xu	0.590	Yes
David Thaler	David Hartley	0.250	No
Changjie Tang	Chang-Jie Tang	0	No
Honghua Yang	Zhonghua Yang	0	No

Movie

We performed similar experiments on a Movie data set, which consists of 11,453 records. Each record in the Movie database (www-db.ics.uci.edu/pages/flamingo/Dataset.htm), created by the University of California, Irvine database group, contains information such as movie ID (a unique numerical identifier), title, year of release, director, producers, studios, category, and awards.

We first extracted a set of 200 potential duplicate pairs from the director field. From the association rules generated, we found Producers, Studios, Category, and Location to be context attributes for director. Assigning equal weights to the context attributes and using the optimal threshold value of 0.6 (see Figure 3), we achieved a maximum accuracy of 96 percent in identifying spurious links. Table 6 shows a sample of the results.

Hep-ph

This experiment sought out spurious links in the KDD Cup 2003 Hep-ph data set (http://arxiv.org/archive/hep-ph), an archive of high-energy physics and particle phenomenology publications. We extracted information from all papers printed over six years and created a data set containing 28,204 records. Each record contains title, author, proceedings, year, and page number information.

We identified 585 record pairs containing potential duplicates of the first author. Again, context attributes include the Coauthor and Proceedings attributes. Assigning equal weights to the context attributes, we obtained a maximum accuracy of 82 percent with a threshold of 0.5.

This data set's lower accuracy stems from the existence of many potential spurious links that had insufficient context information. For example, we found that many publications didn't include coauthor information. Table 7 shows a sample of the results.

Table 6. Sample results for the Movie data set.

Name 1	Name 2	Context similarity value	Actual match?
DeMille	DeMile	0.750	Yes
Hitchcock	Hitchcok	0.636	Yes
Conway	Convay	0.602	Yes
Mulligan	Milligan	0.380	No
Francis	Francisci	0.142	No

Table 7. Sample results for the Hep-ph (high-energy physics and particle phenomenology publications) data set.

Name 1	Name 2	Context similarity value	Actual match?
A. Hoecker	A. Hocker	1.000	Yes
M.E. Carrington	M. Carrington	0.700	Yes
M. Goeckeler	M. Gockeler	0.666	Yes
J. Hashiba	J. Hashida	0.400	No
R. Holman	R. Hofmann	0.200	No
E. Gabrielli	A. Gabrieli	0	No

Authorized licensed use limited to: Sterne Kessler Goldstein Fox. Downloaded on June 04,2024 at 05:04:36 UTC from IEEE Xplore. Restrictions apply.

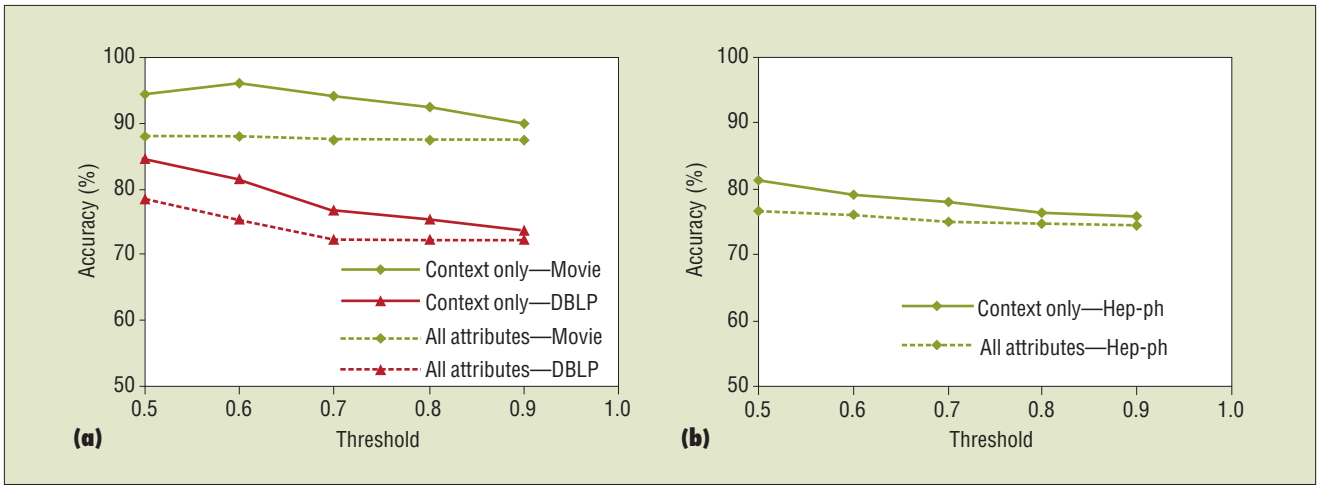

Figure 4. How context attributes affect accuracy in the (a) DBLP and Movie data sets and (b) Hep-ph data set.

Sensitivity experiments

The next set of experiments explored how context attributes and choice of similarity methods affected accuracy in detecting spurious links in the three data sets. Figure 4 shows that for the same threshold values, using context attributes yields higher accuracy than using all attributes.

For example, in the Movie data set, the 96 percent accuracy for context attributes was 8 percent higher than the accuracy for all attributes. This is because using irrelevant attributes tends to decrease the records' overall similarity, thereby leading to the wrong conclusions. So, using only context information is more efficient and accurate in identifying the spurious links.

Next, we evaluated the performance of the column-wise and cosine similarity methods. The cosine similarity measure defines the similarity between two vectors as the cosine of the angle between them. As the cosine value approaches 1, the two vectors become coincident, implying that they refer to the same concept. To determine the similarity between two record sets, we first calculated each attribute's cosine similarity measure and took the average to obtain the overall similarity. We computed the cosine similarity between the sets of records for context

Related Work

The problem of dirty data emerged during 1980s census work by the US Internal Revenue Service.[1] Since then, a steady stream of data-cleaning research has focused on preprocessing of dirty data,[2] noise handling for classification,[3] and duplicate detection and elimination in databases.[4,5] One classification of data-cleaning problems distinguishes between those arising from single and multiple data sources.[5]

Two systems in particular provide a systematic, comprehensive solution to data cleaning. AJAX proposes a declarative framework that extends SQL to allow the specification of data transformation, duplicate elimination, and matching of multiple tables.[6] Potter's Wheel is an interactive framework that provides a graphical user interface for specifying data transformation.[7] Both AJAX and Potter's Wheel are domain-independent approaches.

IntelliClean, a knowledge-based system, follows three main stages.[2] The first preprocesses data to remove abbreviations and standardize the data types and formats. The second stage uses the knowledge base rules to identify and remove approximate duplicate records. The third (or postprocessing) stage involves human intervention to verify and validate the list of duplicates produced.

Clearly, current research and systems have concentrated on eliminating duplicates. Spurious links, however, might persist in data even after the elimination of duplicates.

References

1. *Record Linkage Techniques: Proc. Workshop Exact Matching Methodologies*, B. Kilss and W. Alvery, eds., Statistics of Income Division, US Internal Revenue Service, 1985.

2. W.L. Low, M.L. Lee, and T.W. Ling, "A Knowledge-Based Approach for Duplicate Elimination in Data Cleaning," *Information Systems*, vol. 26, no. 8, Dec. 2001, pp. 585–606.

3. X. Zhu, X. Wu, and Q. Chen, "Eliminating Class Noise in Large Datasets," *Proc. 20th Int'l Conf. Machine Learning* (ICML 03), AAAI Press, 2003, pp. 920–927.

4. M.A. Hernandez and S.J. Stolfo, "The Merge/Purge Problem for Large Databases," *Proc.1995 ACM SIGMOD Conf. Management of Data* (SIGMOD 95), ACM Press, 1995, pp. 127–138.

5. E. Rahm and H.H. Do, "Data Cleaning: Problems and Current Approaches," *IEEE Data Eng. Bull.*, vol. 23, no. 4, Dec. 2000, pp. 3–13.

6. H. Galhardas et al., "AJAX: An Extensible Data Cleaning Tool," *Proc. 2000 ACM SIGMOD Conf. Management of Data* (SIGMOD 00), ACM Press, 2000, p. 590.

7. V. Raman and J.M. Hellerstein, "Potter's Wheel: An Interactive Data Cleaning System," *Proc. 27th Int'l Conf. Very Large Databases* (VLDB 01), Morgan Kaufmann, 2001, pp. 381–390.

Authorized licensed use limited to: Sterne Kessler Goldstein Fox. Downloaded on June 04,2024 at 05:04:36 UTC from IEEE Xplore. Restrictions apply.

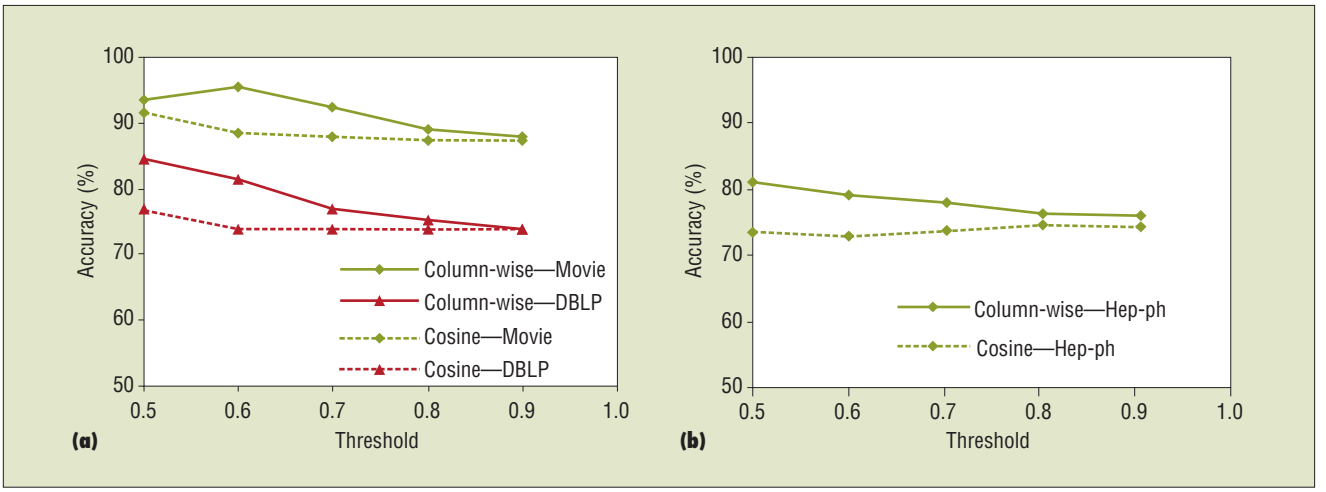

Figure 5. How similarity methods affect accuracy in the (a) DBLP and Movie databases and (b) Hep-ph database.

attribute $c1$ using the formula

$$Sim_{c1}(d_1, d_2) = \frac{d1 \bullet d2}{|d1| \bullet |d2|}.$$

Figure 5 shows the accuracies for different threshold values for the three data sets. We consider only the context attributes here. The column-wise similarity method clearly performed better than the cosine similarity method.

Our method could also help solve a variant of the spurious link problem, where data records for different real-world entities are grouped together as belonging to one real-world entity. For example, in a bibliography database, publications retrieved for an author might not belong to a single person if two authors have the same name. Context information such as coauthors and research area could help us solve this problem. Also, although we have focused on bibliographic data, the method could easily be extended to solve spurious link problems in different data types, such as biomedical and genomic. ■

References

1. S. Ramaswamy, R. Rastogi, and K. Shim, "Efficient Algorithms for Mining Outliers from Large Data Sets," *Proc. 2000 ACM SIGMOD Conf. Management of Data* (SIGMOD 00), ACM Press, 2000, pp. 427–438.
2. X. Zhu, X. Wu, and Q. Chen, "Eliminating Class Noise in Large Datasets," *Proc. 20th Int'l Conf. Machine Learning* (ICML 03), AAAI Press, 2003, pp. 920–927.
3. M.A. Hernandez and S.J. Stolfo, "The Merge/Purge Problem for Large Databases," *Proc. 1995 ACM SIGMOD Conf. Management of Data* (SIGMOD 95), ACM Press, 1995, pp. 127–138.
4. W.L. Low, M.L. Lee, and T.W. Ling, "A Knowledge-Based Approach for Duplicate Elimination in Data Cleaning," *Information Systems*, vol. 26, no. 8, Dec. 2001, pp. 585–606.
5. A.E. Monge and C.P. Elkan, "An Efficient Domain-Independent Algorithm for Detecting Approximately Duplicate Database Records," *Proc. ACM SIGMOD Workshop Research Issues on Knowledge Discovery and Data Mining* (DMKD 97), 1997; www.informatik.uni-trier.de/~ley/db/conf/dmkd/dmkd97.html#MongeE97.
6. H.J. Hamilton, R.J. Hilderman, and N. Cercone, "Attribute-Oriented Induction Using Domain Generalization Graphs," *Proc. 8th Int'l Conf. Tools with Artificial Intelligence* (ICTAI 96), IEEE CS Press, 1996, pp. 246–253.
7. B. Liu, W. Hsu, and Y. Ma, "Integrating Classification and Association Rule Mining," *Proc. 4th Int'l Conf. Knowledge Discovery and Data Mining* (KDD 98), ACM Press, 1998, pp. 80–86.

The Authors

Mong Li Lee is an assistant professor at the National University of Singapore's School of Computing. Her research interests include data cleaning, data integration of heterogeneous and semistructured data, and performance database issues in dynamic environments. She received her PhD in computer science from the National University of Singapore. Contact her at the School of Computing, Nat'l Univ. of Singapore, 3 Science Dr. 2, Singapore 117543; leeml@comp.nus.edu.sg.

Wynne Hsu is an associate professor of computer science at the National University of Singapore's School of Computing. Her research interests include knowledge discovery in databases with an emphasis on data mining algorithms in relational databases, XML databases, image databases, and spatiotemporal databases. She received her PhD in electrical engineering from Purdue University. She is a member of the ACM. Contact her at the School of Computing, Nat'l Univ. of Singapore, 3 Science Dr. 2, Singapore 117543; whsu@comp.nus.edu.sg.

Vijay Kothari is working in India. His research interests include data cleaning and data mining. He received his MSc in computer science from the National University of Singapore. Contact him at No. 22, Murugappa St., Purasawalkam, Chennai, Tamil Nadu, India 600007; vijukothari@yahoo.com.

Authorized licensed use limited to: Sterne Kessler Goldstein Fox. Downloaded on June 04,2024 at 05:04:36 UTC from IEEE Xplore. Restrictions apply.