Methods Matter: Improving USPTO Inventor Disambiguation Algorithms with Classification Models and Labeled Inventor Records

Samuel L. Ventura[1]
Rebecca Nugent[1]
Erica R.H. Fuchs[2]

2012 Workshop on Disambiguation

1 Department of Statistics, Carnegie Mellon University
2 Department of Engineering & Public Policy, Carnegie Mellon University

June 12, 2012

Record Linkage and Disambiguation

Disambiguation is a subset of the broader field, "Record Linkage"

- Match records of unique individuals across two data sources (bipartite record linkage)
- Match records of unique individuals within a single database (disambiguation)

Disambiguation and record linkage have been applied to:

- Linking US Census records to other sources (Jaro 1989)
- Determining which MEDLINE bibliometric records correspond to which unique authors (Torvik and Smalheiser 2009)
- Linking information on crimes in Colombia across three criminal records databases (Sadinle 2010)
- Linking records of assignees in the USPTO database to other data sources, such as Compustat (Zucker et al 2011)

Methods for Record Linkage and Disambiguation

Existing statistical record linkage methods:

- The first mathematical model for bipartite record linkage, including a theorem for obtaining the optimal linkage rule (Fellegi & Sunter, 1969)
- Improved calculation of weight parameters in the Fellegi & Sunter model using the expectation maximization algorithm (Winkler 1988; Jaro 1989)
- Extension of the Fellegi & Sunter model to applications with 3 or more data sources (Sadinle and Fienberg, 2011)

Existing statistical disambiguation methods:

- Torvik and Smalheiser designed an algorithm to disambiguate author names in the MEDLINE database (2009)
  - Bayesian approach to choose record-pairs that are highly likely to match or not match
  - Agglomerative approach to find clusters of author records

These methods do not use labeled records during disambiguation

Record Linkage and Disambiguation in Technology and Innovation Entrepreneurship

Assignee Disambiguation and Record Linkage

- Hall, Jaffe, and Trajtenberg disambiguate USPTO assignees (2002)
- Zucker, Darby, and Fong disambiguate USPTO assignees and link these results to Compustat (2011)

Inventor Disambiguation: Pioneered by Lee Fleming

- Fleming 2007: Simple exact string matching and if-else decision making on the comparison fields (Fleming et al 2007)
- Fleming 2009: Linear weighting scheme of similarity scores for each comparison field (Lai et al 2009); results posted
- Fleming 2011: Implementation of Torvik & Smalheiser algorithm (Lai et al 2011; Torvik & Smalheiser 2009)

  Results compared against a small set of hand-disambiguated records corresponding to 95 US-based academic inventors

These methods do not use labeled records during disambiguation

Solution: Classification Models for Disambiguation

Classification models use labeled records to build statistical models that can predict a categorical feature of unlabeled records (e.g. match?)

- Labeled Records: Records for which the true ID is known
- Why use classification models for inventor disambiguation?
  - Adaptable: Do not rely on pre-defined weights and thresholds
  - Labeled records give insights into which features are important in determining which record-pairs are matches or non-matches
  - Resulting classifier can be used to predict whether or not unlabeled record-pairs match
- Examples of classification models (Hastie et al 2009):
  Logistic Regression, Linear / Quadratic Discriminant Analysis, Classification Trees, Random Forests

Application: Predict whether or not unlabeled record-pairs match

- Input: Similarity scores for each comparison field of each record-pair
  Last, first, middle names; city, state, country; assignee name; etc
- Output: Predicted match or non-match

Evaluate Methodology with Labeled Inventor Records

Evaluate and compare existing disambiguation algorithms and classification models using labeled inventor records

- How many false positive and false negative errors in results?
- Do any of the algorithms favor a particular type of error? Balance both types of errors?
- How could these errors affect research based on these results?

Case Study: We focus on inventor disambiguation within the USPTO patent database (over 8 million patents)

- Data: All inventor records from USPTO patents belonging to the field of optoelectronics (453,973 inventor records)
- Labeled Records: Obtained from CVs collected for a study on economic downturns and technology trajectories in optoelectronics (Akinsanmi, Reagans, and Fuchs 2012)

Our Labeled Inventor Records

Source: Inventors' curricula vitae (CVs) and lists of their patents (Akinsamni, Reagans, Fuchs 2012)

- 281 CV inventors
- 47,125 labeled inventor records
- "Labels" are IDs corresponding to each unique inventor

Inventors come from three groups:

- Top 1.5% of inventors by patent total through 1999 (N = 194)
- Top 1.5% of inventors by patenting rate through 1999 (N = 62)
- Random samples of inventors with patents in different technology classes (N = 25)

The only dataset of labeled records similar to ours in both size and structure is the UCI Machine Learning Repository's "Record Linkage Comparisons Patterns Data Set (2012)

- N = 100,000 labeled epidemiological records

Pairwise Comparisons of Labeled Inventor Records

Labeled Inventor Records:

- List name, location, assignee, etc for each record
- Give IDs indicating the true identification of the individual

Pairwise Comparisons of Labeled Inventor Records:

- Calculate similarity scores for name, location, assignee, etc for each pair of labeled records
- Compare the IDs of a pair of records to see if they match

last	first	mid	suffix	city	state	country	ass	class	subclass	coinv	TRUE	index1	index2
1	0	0.6667	0	0.4889	0	1	0.0000	0	0	0	0	9769	9758
1	0	0.6667	0	0.4222	0	1	0.0000	0	0	0	0	9769	9759
1	0.55	0.6667	0	0.4651	0	1	0.5544	1	0	0	0	9769	9760
1	0.7	0.6667	0	0.6000	0	1	0.5259	0	0	0	0	9769	9761
1	1	1.0000	0	0.4651	1	1	0.5465	1	0	1	1	9769	9762
1	1	1.0000	0	0.4651	1	1	0.9231	1	0	1	1	9769	9763
1	1	1.0000	0	0.4651	1	1	0.9231	1	0	1	1	9769	9764
1	1	1.0000	0	1.0000	1	1	0.9231	0	0	1	1	9769	9765

- Build classification models using this information

Evaluation Metrics: Splitting and Lumping

Lai et al (i.e. Fleming 2011) use Torvik & Smalheiser's interpretation of error metrics "splitting" and "lumping" (2009) to evaluate their results

- Their version focuses only on the largest cluster of records corresponding to each unique individual
- We choose to evaluate all pairwise comparisons the algorithm makes

Splitting: A single unique inventor is "split" into multiple IDs

$$Splitting = \frac{\#\ of\ comparisons\ incorrectly\ labeled\ as\ non\text{-}matches}{Total\ \#\ of\ pairwise\ true\ matches}$$

$$= \textbf{Rate of false negative matches}$$

Lumping: Multiple unique inventors are "lumped" into one ID

$$Lumping = \frac{\#\ of\ comparisons\ incorrectly\ labeled\ as\ matches}{Total\ \#\ of\ pairwise\ true\ matches}$$

$$= \textbf{Rate of false positive matches}$$

Performance of Existing Algorithms on Labeled Records

Existing Algorithm	Splitting (%)	Lumping (%)
Fleming 2007	8.06	0.10
Fleming 2009	0.40	4.77

High splitting % (Fleming 2007):

- Unique inventors don't get credit for all of their patents!
- List of most prolific inventors is incomplete / incorrect
- Inventor mobility is underestimated

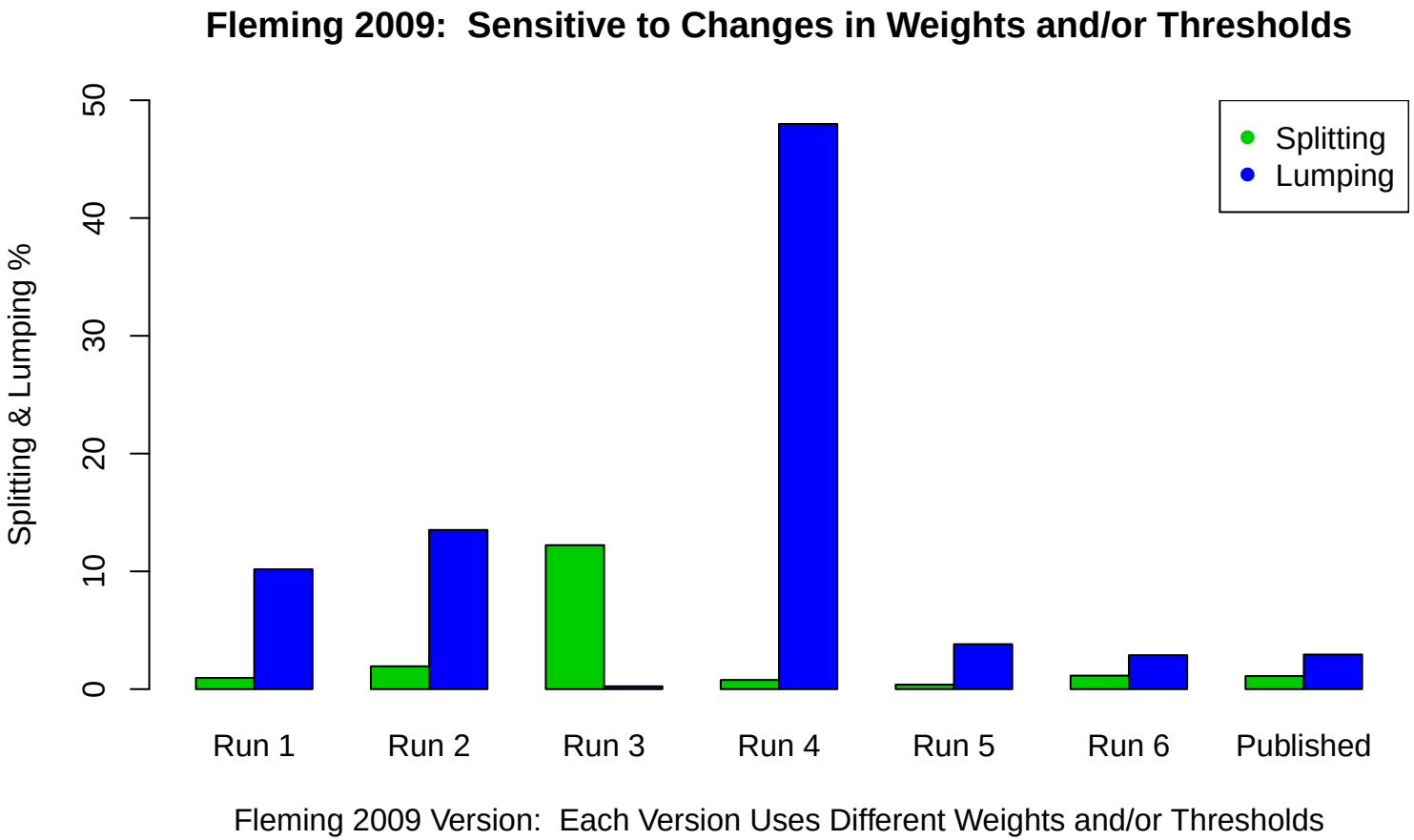
High lumping % (Fleming 2009):

- Unique inventors get credit for additional patents!
- List of most prolific inventors has inventors who don't belong
- Inventor mobility is overestimated

Fleming 2009 Sensitivity Analysis

Fleming 2009: Linear weighting scheme of similarity scores

- Results can change substantially when weights and thresholds are changed slightly



- Results may also change when applied to a new set of inventor records

Performance of Classification Methods on Labeled Records

Disambiguation Method	Splitting (%)	Lumping (%)
Fleming 2007	8.06	0.10
Fleming 2009	0.40	4.77
Linear Discriminant Analysis	6.01	0.51
Quadratic Discriminant Analysis	3.55	0.34
Classification Trees	0.93	1.06
Logistic Regression	0.52	1.32
Random Forests	0.13	0.38

Some classification methods yield improved results:

- Balance low splitting and low lumping
- Random Forests decreases splitting 67.5% over Fleming 2009
- Random Forests decreases lumping 92.0% over Fleming 2009

Our New Classification Approach

Conditional Forest of Random Forests (FoRF): Train random forest classifiers on conditional subsets of labeled pairwise comparisons

1. Split the pairwise comparisons into multiple groups based on known features of the inventor records or similarity scores (e.g. different missingness categories)
2. Train random forests on each group of pairwise comparisons
3. When predicting if pairs of unlabeled records match, use only the appropriate random forests classifier

Example: Three different categories of missingness in middle name

- Both middle names are missing
- One middle name is missing
- Neither middle name is missing

Conditional Forest of Random Forests on Labeled Records

Disambiguation Method	Splitting (%)	Lumping (%)
Fleming 2007	8.06	0.10
Fleming 2009	0.40	4.77
Random Forests	0.13	0.38
Conditional FoRF	0.10	0.08

Conditional Forest of Random Forests (FoRF) further improves inventor disambiguation accuracy

- Conditions on features of the records / comparisons (e.g. US vs. Foreign) and models these different subsets
- Balances low splitting and low lumping
- Reduces splitting by 75.0% over Fleming 2009
- Reduces lumping by 98.3% over Fleming 2009

Why is Conditioning Effective in Inventor Disambiguation?

Motivation: Specific subsets of records or comparisons may have different sets of important features, allowing us to improve model accuracy within subgroups

Conditional FoRF Approach:

1. Condition on a feature of the records or comparisons
2. Build a (random forests) classifier for each conditional subset of pairwise comparisons

Result: The importance of each variable changes across forests

Forest / Variable	Last	First	Middle
Both Missing	271	2088	0
One Missing	402	1520	0
None Missing	2188	8719	3488

(Across rows, relatively higher values indicate more importance in determining which record-pairs are matches or non-matches)

Conclusions

Disambiguation methods within and outside statistics do not build models or (in some cases) validate results using labeled records

Current disambiguation methods in technology and innovation entrepreneurship not only suffer from the above challenges, but also fail to leverage recent model developments in statistics

In the case of optoelectronic inventor disambiguation in the USPTO, we find that Fleming 2009 suffers from systematic errors and tends to overestimate the number of patents belonging to each prolific inventor

Classification models yield improved disambiguation results

- Use labeled records to build models that predict the match vs. non-match response of unlabeled comparisons
- Reduce and balance false positive and negative error rates
- Conditional FoRF yields best results
  - Splitting reduced by 75.0% over Fleming 2009
  - Lumping reduced by 98.3% over Fleming 2009

Acknowledgements

NSF RTG Grant 25951-1-1121631

NSF Science of Science and Innovation Policy Grant,
"Quantifying the Resilience of Industry Innovation Ecosystems"
(Award 0830354)

NSF Science of Science and Innovation Policy Grant,
"CAREER: Rethinking National Innovation Systems Economic
Downturns, Offshoring, and the Global Evolution of Technology"
(Award 1056955)

Next Steps: Summary

1. Balancing the Set of Labeled Inventor Records
2. Implement and Evaluate Fleming 2011
3. Algorithm Dependency on Size
4. Scaling Up to Larger Databases

Next Steps: Balancing the Set of Labeled Inventor Records

Current sample of CVs is potentially biased towards prolific inventors:

- Top 1.5% of inventors by patent total through 1999 (N = 194)
- Top 1.5% of inventors by patenting rate through 1999 (N = 62)
- Random samples of inventors (N = 25)

Incorporating additional labeled records from new inventor CVs will:

- Enhance our classification and evaluation methods
- Allow us to assess and alleviate any potential biases present in our current set of labeled records

Next Steps: Implement and Evaluate Fleming 2011

Implementation: We will run Fleming 2011 on only our set of labeled inventor records

Evaluation:

- The authors reported 3.19% "splitting" and 1.50% "lumping" based on their own small set of labeled inventor records
- We will evaluate our own implementation of Fleming 2011
- We expect Fleming 2011 to slightly outperform Fleming 2009

Algorithm Dependency on Size

Are the results of disambiguation algorithms (past and present) dependent on the size and/or feature distribution of the dataset?

Next Steps: Scaling Up to Larger Databases

Several steps of our algorithm are computationally intensive:

- Similarity score calculation
- Random forests classifier calculation
- Prediction of unlabeled record-pairs using classifier

Explore and analyze the effects that different blocking schemes have on the disambiguation results

Use parallelization methods, which yield a 65% decrease in computational time based on initial tests