Name-Matching Techniques for Disambiguating Interaction Data

INTRODUCTION

Research Background: Our research group is conducting a large-scale Social Network Analysis to analyze the relationships between academic success and social practices among students. To identify these networks, we ask students to report their interactions with other students through open-response name generator surveys. As a result, many of the survey responses contain ambiguous names (lacking a last name, simple name misspellings, etc.). To compile the interaction data, all response names need to be resolved to the correct network entities.

Problem: To disambiguate these responses, we qualitatively considered the process for manual disambiguation during our pilot study. The results of this effort provided an overall process for disambiguating each response (Figure 1). Further, these results provided an overall structure for using computing power to automate most of the disambiguation process. This presentation describes our current work to automate name-matching techniques for disambiguating student network data.

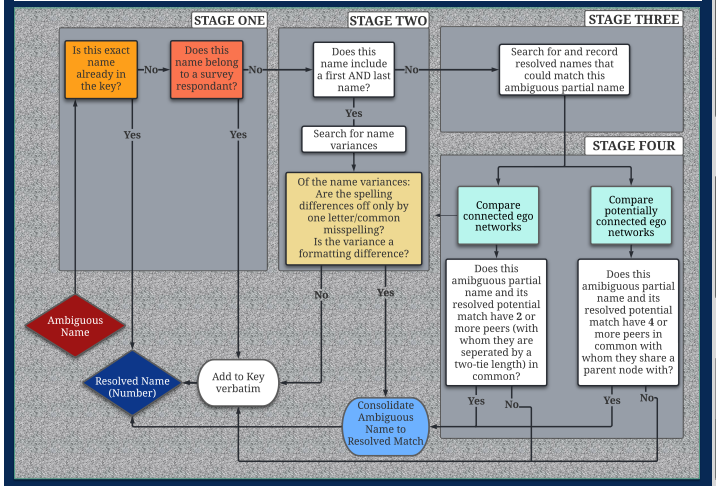


Figure 1. The overall process of resolving an ambiguous name (bottom left, red) into a real-world entity (bottom left, dark blue).

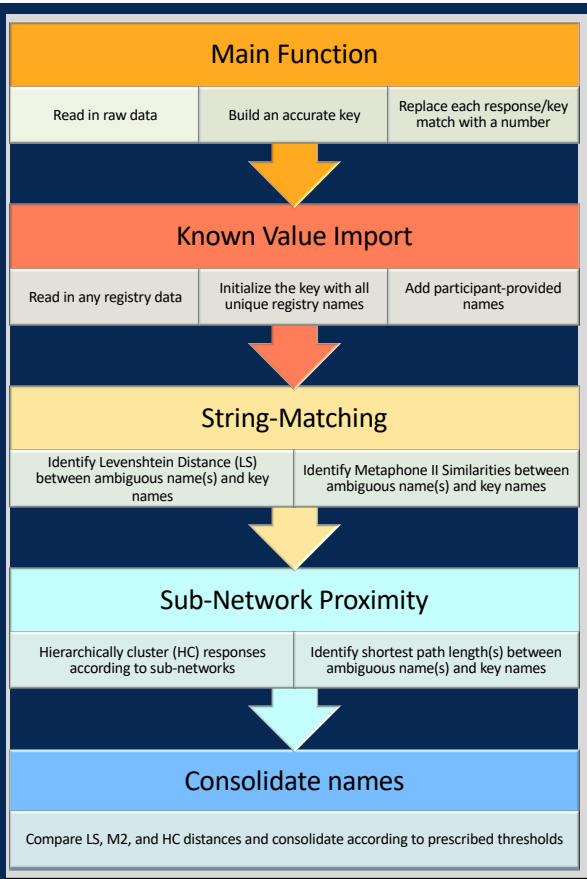


Figure 2. The main function structure for importing, comparing, and consolidating names according to string and sub-network similarity.

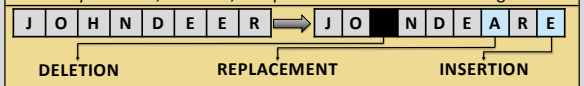
The key helps us record name variances which correspond to each name.
To initialize the key, we use registry data of invited participants (left) and participants' own names.

Name
John Deer
Bob Survey
Earl Excel
Gerry Network

	Original Names		Variant Names			
Number	First Name	Last Name	First Name	Last Name	First Name	Last Name
1	John	Deer	Jon	Deere	John	Deare
2	Bob	Survey	Bobby	Survey	Bobbie	Servey
3	Earl	Excel	Earl	Exel		
4	Gerry	Network	Jerry	Knetwork		

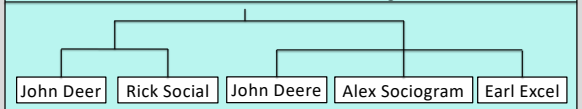
Metaphone algorithm will help us find names that sound similar.
The *Metaphone algorithm* computes a value for the pronunciation of a string.
[John Deer] and [John Deare]
both have the Metaphone keys: JNTR, ANTR

Levenshtein distance helps us find misspelled names.
The *Levenshtein Distance (LS)* determines literal string similarity by quantifying how many additions, deletions, or replacements will make two strings match.



Therefore, [John Deer] & [Jon Deare] have a Levenshtein Distance of 3

Clustering processes help us find if names belong to students that have similar friend groups.
The hierarchical clustering algorithm creates a tree diagram representing the hierarchy of smaller and smaller sub-networks in the overall network as demonstrated in the dendrogram below:



Combining the LS, M2, and HC help us identify overall name similarity.
Using thresholds identified through a manually disambiguated network, we will consolidate names of sufficient LS, M2, and HC similarity

METHODS

To automate these steps, we are writing the scripts in Python through PyCharm, allowing easy access to open-source libraries. To begin, we developed a main function which read the raw data and wraps each stage from Figure 1 as demonstrated by. Our strategy combines Levenshtein distance, Metaphone II, and hierarchical clustering to create a similarity score between an ambiguous name and a resolved name (Figure 2).

RESULTS

To date, we have a script that iterates through registry and interaction data and adds eligible survey participants to the key (Stage 1). Currently, we are working on finding remaining names in the interaction data and compiling name matching values for these names to names in the key. Since we already completed the disambiguation process manually, we will know our script is complete when the results match our previous results.

CONCLUSIONS

Using both Levenshtein distance and Metaphone II algorithms, in addition to agglomerative hierarchical clustering, researchers can resolve ambiguous names into the correct entity more efficiently.

More efficient and accurate name-matching methods will pave the way for study of more holistic social networks.

This material is based upon work supported by the mentor Jack Elliott's National Science Foundation Graduate Research Fellowship under Grant No. DGE1745048. Any opinion, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

Utah State University

Adam Weaver
Utah State University
Department of Engineering Education
adamweaver2000@gmail.com