Generating Look-alike Names For Security Challenges

Shuchu Han[1], Yifan Hu[2], Steven Skiena[3], Baris Coskun[4], Meizhu Liu[2], Hong Qin[3], Jaime Perez[2]
[1]NEC Labs America, [2]Yahoo! Research, [3]Stony Brook University, [4]Amazon AI
shuchu@nec-labs.com,{yifanhu,meizhu,jaimeperez}@oath.com,{skiena,qin}@cs.stonybrook.edu,
barisco@amazon.com

ABSTRACT

Motivated by the need to automatically generate behavior-based security challenges to improve user authentication for web services, we consider the problem of large-scale construction of realistic-looking names to serve as aliases for real individuals. We aim to use these names to construct security challenges, where users are asked to identify their real contacts among a presented pool of names. We seek these look-alike names to preserve name characteristics like gender, ethnicity, and popularity, while being unlinkable back to the source individual, thereby making the real contacts not easily guessable by attackers.

To achive this, we introduce the technique of *distributed name embeddings*, representing names in a high-dimensional space such that distance between name components reflects the degree of cultural similarity between these strings. We present different approaches to construct name embeddings from contact lists observed at a large web-mail provider, and evaluate their cultural coherence. We demonstrate that name embeddings strongly encode gender and ethnicity, as well as name popularity. We applied this algorithm to generate imitation names in email contact list challenge. Our controlled user study verified that the proposed technique reduced the attacker's success rate to 26.08%, indistinguishable from random guessing, compared to a success rate of 62.16% from previous name generation algorithms.

Finally, we use these embeddings to produce an open synthetic name resource of 1 million names for security applications, constructed to respect both cultural coherence and U.S. census name frequencies.

CCS CONCEPTS

• **Security and privacy** → **Authentication**; • **Information systems** → *Data extraction and integration*; Email;

KEYWORDS

user authentication, security challenges, name embeddings

1 INTRODUCTION

User authentication is crucial for most modern web services to function properly. Despite various attempts to replace it, password-based user authentication is still the de facto standard across industry [4]. One important problem with password-based authentication is that user passwords are under constant threat of being compromised due to leaked or stolen databases, password guessing attacks and phishing. Therefore, web service providers often adopt additional security measures to identify suspicious login attempts to prevent attackers from logging in with stolen passwords. These additional measures are largely based on checking whether the current login attempt of a user matches with her previous login activity profile [10]. If a login attempt is deemed to be suspicious, then additional challenges are shown to further authenticate the user, even if she provides the correct password.

In this paper, we focus on automatically generating such challenges for email services. One powerful approach is second-factor authentication, such as sending one-time-password to users' mobile devices. However, second-factor authentication can only cover opt-in users, who could be only a small fraction all users, depending on the application. For the rest, security questions set at registration time are typically used as additional challenges. However studies show that such security questions prove to be relatively ineffective, since they are either very easy to guess by attackers or hard to remember by users [3] [7]. Clearly there remains a strong need to generate reliable security challenges which are easy to solve by genuine account holder but harder to guess by attackers trying to gain access with stolen passwords.

Motivated by this, we propose to automatically generate contact list-based security challenges for email users from their recent activity. Our intuition is that the genuine account holder should be able to distinguish actual contacts they have recently corresponded with from a background of imitation names, whereas a mass attacker who has no personal connection to the user should not.

Constructing the correct answer for this type of challenge from the real contact list is trivial. However, the task of generating background names is more subtle than may appear at first. This challenge is only effective when the background names are culturally indistinguishable from the contacts, a property which does not hold under naive name generation methods. Otherwise, the correct answer will stand out and hence be easily guessed by an attacker. For example, consider an example challenge question asked to a hypothetical user, *wendy_wong@*, given in Table 1. When the background names are generated naively without preserving ethnic properties (right), guessing the correct answer becomes much easier, because the real contact "Charles Wan" has the same ethnicity as the email owner and stands out from the list of randomly generated imitation names. But when the generated names preserve ethnic and cultural properties of the real contact (middle), the guessing task for attacker becomes difficult.

Inspired by recent research advances in distributed word embeddings [2], we propose to generate ethnically coherent imitation names using name embeddings. Our key insight is that people tend to communicate more with people of similar cultural background and gender. Therefore, if we embed names in the vector space so that the distance between name parts reflects their co-occurrence frequency in users' contact lists, this embedding should capture aspects of culture and gender.

The major contributions of our work are:

This research was conducted when Shuchu Han was an intern at Yahoo! Research and Baris Coskun was with Yahoo! Research.

Table 1: A security challenge question: "pick someone you contacted among the following". Left: the contact list of a hypothetical user *wendy_wong@*. Middle: a replacement list generated using the technique proposed in this paper (retaining one real contact *Charles Wan*). Right: a naively generated random replacement list, where the target stands out clearly from the background.

Real Contacts	Proposed Challenge	Naive Challenge
Charles Wan	Fred Wong	John Sander
	Gerald Pang	Steve Pignootti
	Charles Wan	*Charles Wan*
	Eric Yik	Jeff Guibeaux
	Donald Wun	Sam Khilkevich
	Maurice Lau	Mary Lopez

- *Generating realistic replacement names through name embeddings* – We propose a new technique of representing the semantics of first/last names through high-dimensional *distributed name embeddings.* By training on millions of email contact lists, our embeddings establish cultural locality among first names, last names, and the linkages between them, as illustrated by examples in Figure 2. Through nearest neighbor analysis in embeddings space, we can construct replacement aliases for any given name that preserves this cultural locality.
- *Gender, racial, and frequency preservation through name embeddings* – Through computational experiments involving ground truth data from the U.S. Census and Social Security Administration, we show that our name embeddings preserve such properties as gender and racial demographics for popular names and industrial sector for corporate contacts.
- *Establishment of ethnic/gender homophily in email correspondence patterns* – Through large-scale analysis of contact lists, we establish that there is greater than expected concentration of names of the same gender and race for all major groupings under study. We also establish that longer contact lists contain smaller concentrations of men, suggesting than women have larger correspondence circles than men.
- *User study to demonstrate challenge effectiveness* – We conducted an Amazon Mechanical Turk test to compare randomly generated contact list challenges against those constructed using our name embedding approach. The results show that our proposed technique reduced the attacker's success rate to 26.08%, indistinguishable from random guessing, and a substantial improvement over the success rate of 62.16% resulting from previous name generation algorithms.A second user study on 1120 real email users establishes that 88% were able to successfully identify their contact against an even larger background (7 names vs. 3). These high success rates validates this approach to user challenge/verification.

The outline of this paper is as follows. Section 2 reviews related work. Section 3 presents our approach to constructing name embeddings, including an evaluation of different optimization criteria. Section 4 establishes that name embeddings preserve information concerning gender, ethnicity, and even frequency. Section 5 builds on this to study the gender and ethnicity properties of email contact lists. Section 6 conducts a controlled user study to verify the effectiveness and feasibility of the proposed contact list challenges. In Section 7, we investigate synthetic name generation without template names, generating an open resource of 1,000,000 synthetic names. We conclude in Section 8 with discussions on remaining challenges.

2 RELATED WORK

User Challenge/Security Issues. Reliable and secure challenges are essential in verifying the real identity of a login attempt (e.g., when the sign-on comes from a geo-location the user was never observed at before), and when the user tries to recover an account from which she could not remember the password.

Personal knowledge questions (also called "secret questions" or "challenge questions") have long been used as backup mechanism to reclaim lost accounts [14]. However, recent study [3] shows that such questions have a poor level of security and can be unreliable. They are insecure because the low entropy of the answers (e.g., the answer to "favorite food" can be guessed at 19.7% success rate). They are unreliable because users often forgot the answer, partly due to the fact that a significant fraction of users provided fake answers when setting up the security questions in an attempt to make them "harder to guess". On aggregate this behavior had the opposite effect as people "harden" their answers in a predictable way. This is why SMS and backup Email are the preferred challenge and recovery mechanism.

Nevertheless, challenge questions as alternatives to, or in conjunction with SMS/email are still valuable. They serve either as additional signals, or as a last resort when the user has no longer access to the phone/backup Email. One such question that is relatively easy for the account owner to answer is to distinguish genuine contacts that they corresponded with, from a background of imitation names. Such challenges have been used by applications including Facebook and WeChat. For example WeChat asks the user to pick photos and names of friends from a list of background photos and names. This kind of challenge questions are only effective when the contacts do not standout from the background ones, in terms of ethnicity, gender, and name frequency. This paper studies contact list challenges consisting of real and imitation contacts. We propose a data-driven approach for generating look-alike names that are appropriate as imitation contacts for these challenges.

Word and Graph Embeddings. Neural word embedding techniques, exemplified by the popular word2vec [17, 21], are now known to be effective in capturing syntactic and semantic relationships. Levy and Goldberg [15] found that the skip-gram based word2vec embedding can be considered a matrix factorization technique, with the matrix to be factored containing the word-word point-wise mutual information. With this broad understanding of word2vec in mind, the technique is applicable to tasks beyond those of traditional natural language processing. It can be employed whenever there is a large amount of data consist of entities and their co-occurrence patterns. Our work on name embedding is

Male names	1th NN	2nd NN	3rd NN	4th NN	5th NN	Female names	1th NN	2nd NN	3rd NN	4th NN	5th NN
Andy	Andrew	Ben	Chris	Brian	Steve	Adrienne	Allison	Aimee	Amber	Debra	Amy
Dario	Pablo	Santiago	Federico	Hernan	Diego	Aisha	Aliyah	Nadiyah	Khadijah	Akil	Aliya
Elijah	Isaiah	Joshua	Jeremiah	Bryant	Brandon	Brianna	Brittany	Briana	Samantha	Jessica	Christina
Felipe	Rodrigo	Rafael	Eduardo	Fernando	Ricardo	Candy	Connie	Becky	Angie	Cindy	Christy
Heath	Brent	Chad	Brad	Brett	Clint	Chan	Wong	Poon	Ho	Wai	Yip
Hilton	Xooma	Eccie	Erau	Plexus	Gapbuster	Cheyenne	Destiny	Madison	Brittany	Taylor	Kayla
Isaac	Samuel	Israel	Eli	Esther	Benjamin	Dominque	Renarda	Lakenya	Lakia	Lashawna	Shatara
Jamal	Jameel	Kareem	Anmar	Khalifa	Nadiyah	Ebonie	Lakeshia	Tomeka	Ebony	Latasha	Shelonda
Lamar	Terrell	Derrick	Eboni	Tyree	Willie	Florida	Fairfield	Integrity	Beacon	Southside	Missouri
Mohammad	Shahed	Mohmmad	Ahmad	Rifaat	Farishta	Gabriella	Daniella	Vanessa	Marilisa	Isabella	Elisa
Moshe	Yisroel	Avraham	Gitty	Rivky	Zahava	Giovanna	Giovanni	Elisa	Paola	Giuliana	Mariangela
Rocco	Vito	Salvatore	Vincenza	Pasquale	Nunzio	Han	Jin	Yong	Sung	Huan	Teng
Salvatore	Pasquale	Nunzio	Gennaro	Vito	Tommaso	Kazuko	Keisuke	Junko	Yumi	Yuka	Tomoko
Thanh	Minh	Thuy	Thao	Ngoc	Khanh	Keren	Ranit	Galit	Haim	Zeev	Rochel

Table 2: The five nearest neighbors(NN) of representative male and female names in embedding space, showing how they preserve associations among Asian (Chinese, Korean, Japanese, Vietnamese), British, European (Spanish, Italian), Middle Eastern (Arabic, Hebrew), North American (African-American, Native American, Contemporary), and Corporate/Entity.

such an example. Another example is DeepWalk [22], a novel approach for learning latent representations of vertices in a graph by constructing “sentences" via random walk on the graph.

Ethnicity Analysis. Identification of ethnicity from names has applications in many areas, including biomedical research, demographic studies, and targeted advertising. The starting point of such identification is a taxonomy of ethnic groups designed to capture the multiple facets of ethnicity, including language, religion, geographical region, and culture. Mateos, Webber and Longley [16] presented an ontology of ethnicity that classifies the UK population into 15 groups (e.g., African, European, Hispanic, Jewish, Muslim, South/East Asian, etc).In this paper, we adopt the definition by the US Census Bureau and use a broader taxonomy of ethnic groups (see Table 3).

The problem of ethnicity identification from names has been studies by many researchers. Ambekar et al. [1] proposed a name classifier based on hidden Markov models (HMMs) and decision trees, which operate on the components of a name. They classified names into 13 cultural/ethnic groups drived from [16], with ground truth data extracted from the Wikipedia. Treeratpituk and Giles [23] also used the Wikipedia data, and built a logistic regression classifier with four types of features: nonASCII chacters, n-grams, Double Metaphone n-grams and Soundex. A list of earlier works can be found in [1].

While previous studies like [1] and [23] are oriented around building a classifier using natural language processing (NLP) techniques, there has also been work that consider the population as a whole and infer the ethnic composition of the group. Chang et al. [9] proposed a generative model to determine the ethnic breakdown of a population based solely on names and Census data. They used this model to study the ethnic composition of the Facebook users over time. In addition, they found that different ethnic groups relate to one another in an assertive manner, a finding confirmed by our work. Haris [12] proposed another method to infer the ethnic composition of a population, using Bayes rule and the Expectation-Maximization technique.

3 BUILDING NAME EMBEDDINGS

3.1 Methodology

We seek to construct a list of background names that looks plausibly real, even though they are machine generated. One way to create such a list is to start from real contact names, and replace each one with a look-alike names of the same gender, ethnicity and name frequency. However this approach requires multiple complex components, including reliable ethnicity and gender classifiers. Instead we propose to do away with these complex steps by deriving the signals directly from a large amount of data, through name embeddings.

In our approach, each name, first or last, is embedded in high dimensional space as a high dimensional vector using word2vec [18]. Our observation (verified in Section 5) is that people have a tendency to contact people of the same ethnicity and gender. Note that we don’t claim that this is true for each and every user, but rather there is a slight bias within the global population. Consequently, when using the contact lists of millions of users as a text corpus, the resulting embedding of names would capture this tendency by placing names of the same gender and ethnicity close-by in the high-dimensional space. For each real name in the contact list, we can then choose a name near it in the high dimensional space. The resulting replacements should have gender and ethnicity similarity to the original names (Table 1 (middle)). Furthermore, in aggregate, the name frequency of the look-alike names should also resemble that of the real name distribution.

3.2 Data Sources and Preparation

In this section, we introduce the datasets that are used in this study, as well as a detailed description about our data preparation process. Here, we would like to emphasize that the collected data is stored and utilized in full compliance with related data governance policies.

Data Sources. Datasets employed in our work are:

- *Contact Lists* – This set of data, here after referred to as the *contact lists* is a proprietary sample of contact lists from 2 million distinct email users of a major Internet company.

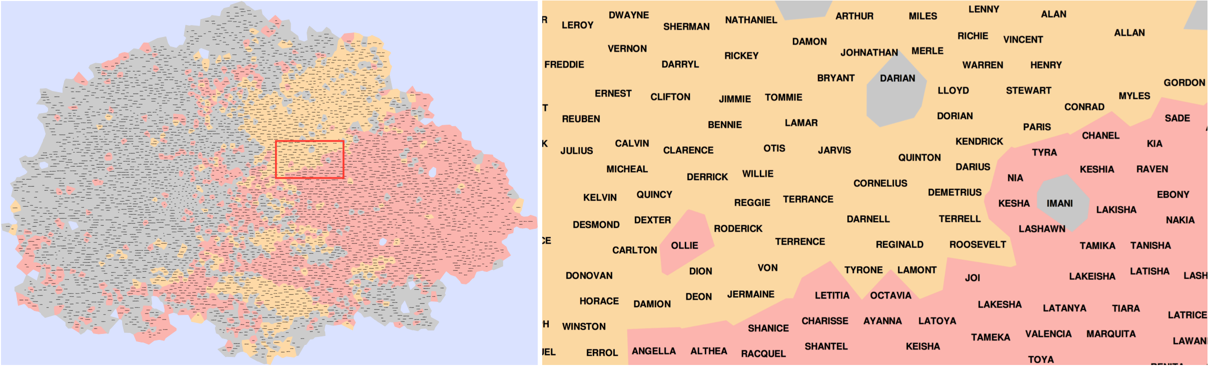


Figure 1: Visualization of the name embedding for the top 5,000 first names from email contact data, showings a 2D projection view of name embedding (left). The pink color represents male names while orange denotes female names. Gray names have unknown gender. The right figure is a close view along the male-female border, centered around African-American names.

	Census 2000		*Contact lists*	
Races	Count	Percentage	Count	Percentage
White	115,167	0.8593	12,837,406	0.7428
Black	5,262	0.0393	544,983	0.0315
API	6,100	0.0455	1,323,888	0.0766
AIAN	268	0.002	19,272	0.0011
2PRace	131	0.001	7,934	0.0005
Hispanics	7,089	0.0529	2,548,329	0.1475
Total	134,017		17,281,812	

Table 3: Ethnicity distribution of *Census 2000* data and that of intersected names between the *contact list* and *Census 2000* data. The label information comes from *Census 2000.*

Names in each contact list are ordered by contacting frequency/recency. The length of each contact list varies from 1 to 21, because longer lists have been truncated to remove less frequent/older contacts. Each entry of a list is a full name but not necessarily a human name. To preserve the privacy of users, the owner associated with each contact list was not available in the data. This complicates certain analysis, such as correlating the gender/ethnicity of account holders with their contacts (Section 5). We note that the contact lists are always used in aggregate. No use is made of an individual contact list. Furthermore, first/last names that appear infrequently have been filtered out. This study/paper was reviewed and approved by the appropriate institutional review board (IRB).

- *Census 1990* [5]– The Census 1990 dataset is a public dataset from US Census website. It records the frequently occurring surnames from US Census 1990. This dataset contains 4,725 popular female names and 1,219 popular male names.
- *Census 2000* [6] – The Census 2000 dataset is another public dataset from US Census website. It contains the frequently occurring 151,672 surnames from US Census 2000. Associated with each name is a distribution over six categories of races. The races are: White, Black, Asian/Pacific Islander (API), American Indian/Alaskan Native (AIAN), Two or more races (2PRACE), and Hispanics. In this paper we refer to the races and ethnicity interchangeably.

The ethnicity distributions of *Census 2000* and *Contact lists* data are given in Table 3.

Data Preparation. The *contact lists* data records the social interaction of email users. These contact lists include substantial noise in the name fields. Artifacts include omitted names [11], sometimes marked by an arbitrary string like "zzzzzzzz", or names that are meaningful but clearly not human, like "Microsoft", "Facebook", and "GEICO". Moreover, many names in contact lists data are partial, with only the first name or the last name present.

To improve the quality and integrity of the *contact list* data, we apply the following data cleaning processes to the original data following the guidance of US Census 2000 demographic report [25].

1. Remove non-English characters.
2. Remove known special appellations, such as "Dr", "Mr", "MD", "JR", "I", "II", "III".
3. Remove middle names. First/last name is the first/last part of a full name. For example, for name "Margarita M. Alvarez", only "Margarita" and "Alvarez" are kept.

After data cleaning and removing lists containing no names, 92% of the contact lists remain.

3.3 Word2vec Embeddings

Word2vec [20] is an efficient tool to learn the distributed representation of words for large text corpus. It embeds words in high dimensional space so that words that tend to occur in the same context are close-by in the embedding space. More specifically, each word is represented by a normalized embedding vector, and the similarity between two words is the dot product of the corresponding vectors. It comes with two models: the Continuous Bag-of-Words model (CBOW) and the Skip-Gram (SG) model. The CBOW model predicts the current word based on the context while the Skip-Gram model does the inverse and maximizes classification of a context word based on the current word [17]. Word2vec can be considered a matrix factorization technique [15], with the matrix to be factored containing the word-word point-wise mutual information. With this broad understanding of word2vec in mind, the technique is applicable whenever there is a large amount of data consist of entities and where the entity co-occurrence patterns are of interest.

We start our analysis by using the cleaned *contact lists* and the word2vec software [20]. Each contact list is treated as a sentence, and together they form a text corpus. Unless otherwise stated, all results in the paper are based on the CBOW model with the default word2vec parameter settings (see Section 3.4 for comparison of different models and ways of constructing the corpus). The output of word2vec is a dense matrix with dimensions 517, 539×100, where each unique name is represented as a row of the matrix.

Embedding Visualization. To understand the landscape of the name embeddings, we visualize the names as a 2D map. We used the stochastic neighborhood embedding [24] to reduce the original 100-dimensional embedding to 2D. We assign each name to a cluster using gender/ethnicity ground truth, and created the maps using gvmap [13].

Figure 1 (left) illustrates the landscape of first names. This visualization establishes that the embedding places names of the same gender close-by. Using Census data, we color male names orange, female names pink, and names with unknown gender gray. Overall names of the same gender form mostly contiguous regions, indicating that the embedding correctly capture gender information by placing names of the same gender close-by. Figure 1 (right) is an inset showing a region along the male/female border. We can see that “Ollie", which is considered a predominantly female name per Census data (2:1 ratio of female/male instances), is placed in the male region, close to the male/female border. Per [8], we found that “Ollie" is more often a male name, and used as a nickname for “Oliver". Hence our embedding is correct in placing it near the border. The embedding also correctly placed “Imani" and “Darian", two names not labelled by the Census data, near the border, but in the female/male regions, respectively. Per [8], “Imani" is a African name of Arabic origin, and can be both female and male, mainly female; “Darian" can also be female and male, but mainly male, and is a variant of “Daren" and “Darien", among others.

Fig. 2 (left) presents a map of the top 5000 last-names. We color a name according to the dominant racial classification from the Census data. The top 5000 names contain four races: White (pink), African-American (orange), Hispanic (yellow), and Asian (green). Names without a dominant race are colored gray. The three cutouts in Fig. 2 highlight the homogeneity of regions by cultural group. The embedding clearly places White, Hispanic and Asian in large contiguous regions. African-American names are more dispersed. Interestingly, there are two distinct Asian regions in the map. Fig. 3 presents insets for these two regions, revealing that one cluster consists of Chinese names and the other Indian names. Overall, Fig. 1 and Fig. 2 show that our name embeddings capture gender and ethnicity information well.

3.4 Evaluation of Different Word2vec Embeddings

The embedding from word2vec is influenced by two factors: the input text, and the word2vec parameter settings. So far we have been using the *contact lists* unchanged as the input to word2vec. However there are many possible variants. Instead of putting both first and last names together in one embedding space, one variant might generate two embeddings, one using only the first names, and another using the last names. In addition to the input text, the second factor that influences the embedding is the different settings of models and parameters, for example, the selection between CBOW model and SG model, the size of sampling window and the size of negative samples.

To understand how these two factors influence the embedding, in particular with regard to the quality of the resulting look-alike names, we evaluate the following variants of the word2vec embeddings:

- Set the word2vec model to be CBOW or SG.
- Generating joint embeddings of first names and last names using the contact lists as they are (“CBOW joint" or “SG joint").
- Generating embedding for first names and last names separately by including only first/last names in the contact lists (“CBOW sep" or “SG sep").

Metrics. To evaluate the quality of the embeddings with regard to look-alike names, we propose two metrics to measure popularity, gender and ethnicity similarities between real and look-alike names.

First, we seek the overall frequency of a name in the real contact lists to be similar to that in the look-alike lists. For example, if “Mark" is a more popular name than “Barnabas" in the real contact lists, we would also expect that “Mark" appears more often as a replacement name than “Barnabas". We define two types of frequencies. The *real name frequency* is the frequency of names in the *Contact list*. The *replacement usage frequency* is the frequency of a name in the replacement name population. To measure the popularity dis-similarity, we sample 10k names randomly from the name list, with sampling probability proportional to *real name frequency*. We record ten nearest neighbors (NN) for each of them. Now the popularity dis-similarity is computed by *Jensen-Shannon Divergence* using the *real name frequency* of the sampled names and *replacement usage frequency* of the nearest neighbor names. Second, we measure gender similarity by precision at k, defined as the percentage of the k-nearest neighbors of a name having the same gender as the name itself. Finally, we measure ethnicity similarity by precision at 1. For example, the precision for White names is defined as $P(W|W) = P(\text{1st NN is White}|\text{real name is White})$.

Results. We present the evaluation results in Table 4. The CBOW model generally outperformed the SG model for the majority of the nearest neighbor tests. Therefore we use “CBOW joint" as the embedding throughout this paper. Note that while $P(B|B)$ (35%-59%) is generally much lower than $P(W|W)$ (92%-94%), considering that a randomly picked name from the contact list has a probability of 74% of being White but only a probability of 3% of being Black, $P(B|B)$ is actually significantly above the odds a random name is black.

4 PROPERTIES OF NAME EMBEDDINGS

After word2vec, each name part is represented by a vector in the high dimensional space. Earlier, in Fig. 1 and Fig. 2, we have provided visual evidence that this embedding is coherent, in the sense that it clusters names of similar gender and ethnicity.

4.1 Gender Coherence and Analysis

We first examine the gender coherence of first names and their ten nearest neighbors. The subset of first names we consider is the intersection between *contact lists* and *Census 1990*. It contains 1,146 unique male first names and 4,009 unique female first names. All

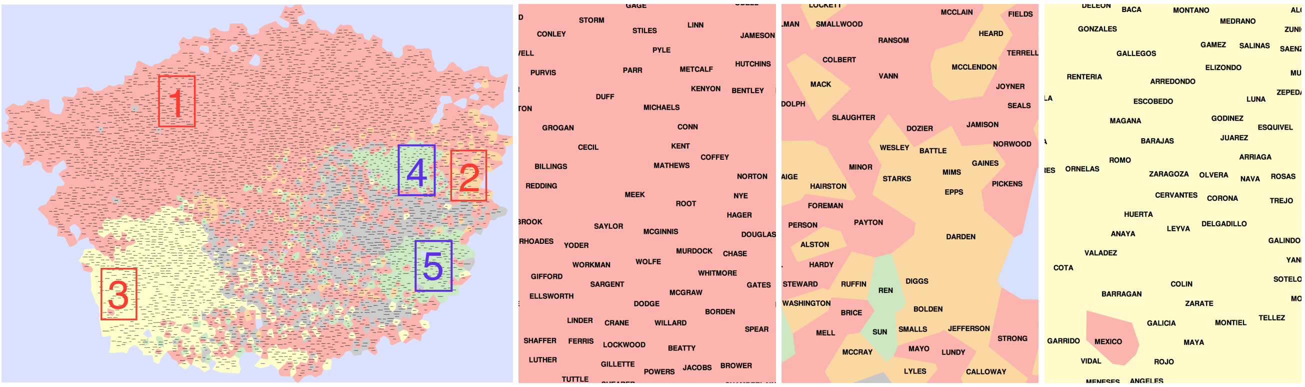


Figure 2: Visualization of the name embedding for the top 5000 last names, showings a 2D projection view of the embedding (left). Insets (left to right) highlight British 1, African-American 2 and Hispanic 3 names.

Variation	Popularity	Gender		Ethnicity (NN(1))			
		NN(1)	NN(10)	$P(W\|W)$	$P(B\|B)$	$P(A\|A)$	$P(H\|H)$
CBOW joint	0.6434(0.0007)	0.9092	**0.9360**	0.9362	**0.5939**	**0.7626**	**0.7543**
SG joint	0.6747(0.0002)	0.8844	0.9274	**0.9461**	0.4561	0.7208	**0.7543**
CBOW sep	0.6675(0.0003)	**0.9162**	0.9350	0.9299	0.4437	0.7167	0.6710
SG sep	**0.5776(0.0001)**	0.8844	0.9205	0.9217	0.3451	0.6797	0.6971

Table 4: Evaluation of different embedding variants, *CBOW*: continues bag-of-words model. *SG*: skip gram model. The suffix *joint* means first names and last names are used together as the input for word2vec, while *sep* means separately. *Popularity* is measured by Jensen-Shannon Divergence of frequency distribution, while all other values are precision at $k = 1$ or $k = 10$.

names in the subset are ranked by their popularity as measured in the *Census 1990* data. Table 5 presents our gender coherence results, measured by precision at k, as a function of the population of the names, and k, the number of nearest neighbors. For example, the cell at $\{\leq 20\%, 2\}$ of Table 5 (left) reads 97. This means that for the top 20% most popular names, 97% of their nearest 2-neighbors have the same gender as them. To save space, we only report the first two significant digits of each precision (e.g., 0.9742 becomes 97). In addition we color the cells of the tables based on the values. Within each table, we use warm colors for high values and cold color for low values. This gives us heat-maps through which it is easier to see the trend of how the precision varies with popularity of the first name, and the number of neighbors.

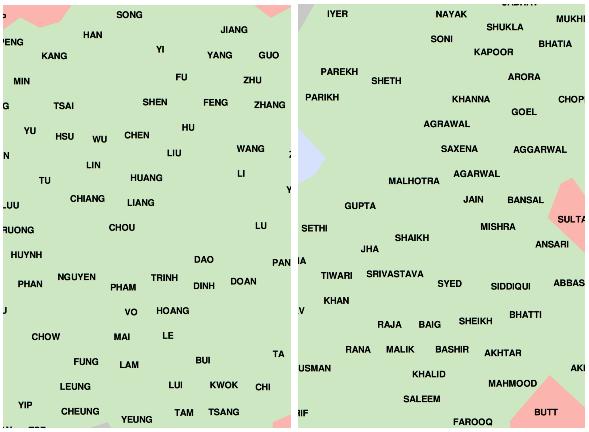


Figure 3: The two distinct Asian clusters. Left: Chinese/South Asian names (4 in Fig. 2). Right: Indian names (5 Fig. 2).

Top %	1	2	3	4	5	6	7	8	9	10	Top %	1	2	3	4	5	6	7	8	9	10
≤ 10%	100	99	98	98	98	98	98	98	98	98	≤ 10%	97	97	97	96	95	95	95	95	95	95
≤ 20%	99	97	96	96	95	95	95	95	95	95	≤ 20%	91	91	91	90	89	89	89	89	88	88
≤ 30%	96	95	94	93	93	92	93	92	92	91	≤ 30%	85	84	84	84	83	83	82	82	82	81
≤ 40%	93	93	91	90	90	89	89	89	88	88	≤ 40%	80	79	79	78	78	77	77	77	76	76
≤ 50%	89	89	86	85	85	84	84	84	83	83	≤ 50%	75	74	74	73	73	72	72	72	71	71
≤ 60%	86	86	84	83	82	82	82	81	81	80	≤ 60%	69	69	68	68	67	67	66	66	66	66
≤ 70%	82	81	79	79	78	77	77	76	76	76	≤ 70%	66	65	66	65	64	64	63	63	63	63
≤ 80%	79	78	76	75	74	74	74	73	73	72	≤ 80%	62	61	61	61	60	60	59	59	59	59
≤ 90%	76	75	73	73	72	71	71	71	70	70	≤ 90%	59	59	59	58	58	57	57	57	57	57
All	73	72	70	69	69	68	68	67	67	67	All	57	56	56	56	55	55	55	54	54	54

Table 5: Gender coherence of the name embedding for males (left) and females (right), as measure by the percentage of k-neighbors being male (left) and female (right).

Table 5 shown that our proposed name embedding scheme has strong gender coherence, especially for popular names. As we can see from the tables, the percentage of neighbors that have same gender as the original first name is very high for the top 30% most popular names, compared to a randomly assigned name (50%). This percentage decreases when unpopular names are included, and similarly decreases the number of neighbors increases.

4.2 Ethnicity Coherence and Analysis

We evaluate the ethnicity coherence by examining the ethnicity of a last name and its 10 nearest neighbors. The evaluation is based on the intersected last names between *Census 2000* and the *contact list*. The coherence values are computed by the percentage of nearest neighbors that have same ethnicity as a query name itself. To better understand the coherence trend, we use the same strategy as with gender coherence analysis, and examine the precision as a function of the popularity of the names, and the size of nearest neighbors. The results are presented in Table 6. In general, the top neighbors of a popular name tend to have a high probability of being in the same ethnicity group. The coherence for an ethnic group correlates positively with the popularity of the group in the *contact lists*. The coherence for AIAN and 2PRACE are poor, because they only account for 0.1% and 0.05% of the last names in the contact lists.

4.3 Name Popularity Analysis

To measure the popularity preserving properties of word embeddings, we calculate the *real name frequency* of a name (R), the average *real name frequency* of its replacement names (ten nearest neighbors) (A), and its *replacement usage frequency* (U). Two measurements, Pearson's correlation coefficient (PCC) and Spearman's rank correlation coefficient (SCC), are used to measure how well the popularity is preserved. The results are shown in Table 7. Overall we can see that the correlation between the *real name frequency* R and the *replacement usage frequency* U is higher than that for the *real name frequency* R and its neighbors' *real name frequency* A. This indicates that a popular name is very likely to appear in among the nearest neighbors of other names, even though its nearest neighbors are not necessarily popular names.

5 COHERENCE PROPERTIES OF EMAIL CONTACT LISTS

Visualization in Fig. 1 and Fig. 2, and quantitative analysis in the previous section have confirmed that our name embedding is able to capture both gender and ethnicity information. Since the embedding is generated in a completely *unsupervised* manner by applying word2vec to the contact lists of millions of people, it is curious that the embedding can capture gender and ethnicity so well. The principal of homophily suggests that in aggregate, *users exhibit a preference to communicate with people of the same ethnicity and gender.* In this section we substantiate this observation through statistical analysis.

5.1 Coherence in Gender Distribution

One important aspect of a name is its associated gender. To identify a name's gender, the first name is always preferred than last name in demographic studies [19]. Here, we follow this popular rule and use the first name to identify the gender of a given full name. The gender of a name could be male, female or unknown. The "unknown" names could be human names with no known ground truth gender, or non-human names, for example, "Microsoft" or "Amazon". To avoid the bias of any specific machine learning classifier, we rely on dictionary look-up method to identify the gender of a name using the *Census 1990* data.

To start with, we look at the gender distribution of the *contact lists* as a function of the length of the contact list. Fig. 4 shows the average percentage of males as a function of the length of the contact lists (red), as well as as a function of the length of gender-identifiable names in the lists. It is seen that the longer the list, the smaller percentage of males it contains.

Top %	1	2	3	4	5	6	7	8	9	10
≤10%	96	96	96	96	96	96	96	96	96	96
≤20%	96	96	96	96	96	96	96	96	96	96
≤30%	96	96	96	96	96	96	96	96	96	96
≤40%	96	96	96	96	96	96	96	96	96	96
≤50%	95	95	95	95	95	95	95	95	95	95
≤60%	95	95	95	95	95	95	95	95	95	95
≤70%	95	95	95	95	95	95	95	95	95	95
≤80%	94	94	94	94	94	94	94	94	94	94
≤90%	94	94	94	94	94	94	94	94	94	94
All	94	94	94	94	94	94	94	94	94	94

White

Top %	1	2	3	4	5	6	7	8	9	10
≤10%	62	59	57	55	54	53	52	52	51	50
≤20%	65	63	60	59	58	58	57	57	56	56
≤30%	63	62	61	60	59	59	58	58	58	57
≤40%	63	62	61	60	59	59	59	59	59	58
≤50%	62	61	60	60	59	59	58	58	58	58
≤60%	61	61	60	60	59	59	58	58	58	58
≤70%	61	61	60	59	59	59	58	58	58	58
≤80%	60	60	59	59	58	58	58	58	57	57
≤90%	60	59	59	59	58	58	58	57	57	57
All	59	59	59	59	58	58	58	57	57	57

Black

Top %	1	2	3	4	5	6	7	8	9	10
≤10%	90	91	91	91	91	91	91	91	91	91
≤20%	89	89	89	89	89	89	89	89	89	89
≤30%	86	85	86	86	86	86	86	86	86	86
≤40%	83	83	84	84	84	84	84	84	84	84
≤50%	81	81	81	81	82	82	82	82	81	81
≤60%	80	80	80	80	80	80	80	80	80	80
≤70%	79	79	79	79	79	79	79	79	79	79
≤80%	78	78	78	78	78	78	78	78	78	78
≤90%	77	77	77	77	77	77	77	77	77	77
All	76	76	76	76	76	76	76	76	76	76

API

Top %	1	2	3	4	5	6	7	8	9	10
≤10%	19	19	22	22	22	21	20	20	19	18
≤20%	17	19	20	19	18	17	16	16	15	14
≤30%	15	18	18	17	16	15	14	14	13	13
≤40%	13	15	15	14	14	13	12	12	11	10
≤50%	12	13	13	11	11	11	10	10	9	9
≤60%	11	12	11	10	10	10	9	9	8	8
≤70%	10	11	10	10	9	9	8	8	7	7
≤80%	9	10	9	9	8	8	7	7	6	6
≤90%	9	9	9	8	8	7	7	7	6	6
All	8	9	8	8	7	7	6	6	6	6

AIAN

Top %	1	2	3	4	5	6	7	8	9	10
≤10%	54	46	44	44	45	46	47	47	45	45
≤20%	54	50	49	49	49	49	51	51	50	50
≤30%	56	53	52	51	51	51	52	52	50	50
≤40%	54	51	52	52	52	52	52	52	50	51
≤50%	58	52	54	55	54	54	53	54	52	53
≤60%	58	51	50	53	52	51	51	51	50	51
≤70%	57	52	53	54	53	52	52	52	51	52
≤80%	56	51	51	52	51	50	51	51	51	51
≤90%	53	49	50	50	50	49	49	50	49	49
All	51	48	49	49	48	47	48	48	48	48

2PRACE

Top %	1	2	3	4	5	6	7	8	9	10
≤10%	97	97	97	96	96	96	96	96	95	95
≤20%	95	94	94	94	94	94	93	93	93	93
≤30%	91	91	91	90	91	90	90	90	90	90
≤40%	89	89	88	88	88	88	88	88	88	88
≤50%	86	86	86	86	86	86	86	86	85	85
≤60%	83	83	83	83	83	83	83	82	82	82
≤70%	81	80	81	81	81	81	81	80	80	80
≤80%	79	79	79	79	79	79	79	79	79	79
≤90%	77	77	77	77	77	77	77	77	77	77
All	75	76	76	76	76	76	76	76	75	75

Hispanic

Table 6: Percentage of k-nearest ($k = 1, 2, \ldots, 10$) neighbors of a name that has the same ethnicity as itself, when restricting the name in the top p-percent ($p = 10, 20, \ldots, 90, All$) of names. API: Asian/Pacific Islander. AIAN: American Indian/Alaska Native. 2PRace: two or more races.

	PCC		SCC	
	RA	*RU*	*RA*	*RU*
First names	0.5813	0.7795	0.5170	0.5402
Last names	0.2260	0.4454	0.3444	0.3916

Table 7: Correlation of real names and replacement names.

Second, we want to know the difference between the observed *contact lists* and randomly generated contact lists. Our observation is that users' contact lists exhibit a bias towards either male domination, or female domination. To substantiate this observation, we look at the frequency distribution of percentage of males in

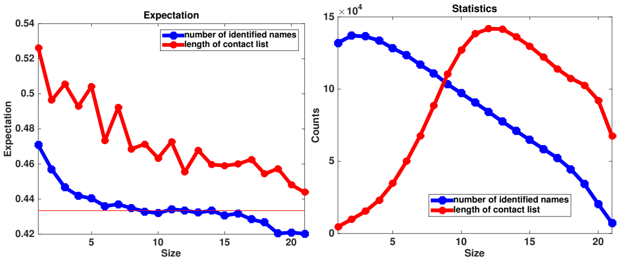


Figure 4: Left: the expectation of male names frequency as a function of the size of identified names and contact list lengths. Right: the number of contact lists for different size of identified names and contact list lengths.

our mailing list, and compare with the null hypothesis. In Fig. 5 (left), we divide contact lists by a threshold based on the minimum number T of identifiable genders in the list. E.g., $T = 5$ means those contact lists with at least five gender-identifiable names. The distributions of the ratio of identifiable males in the contact lists with $T = 5$ and 10 are seen as the two lower curves. Clearly, the majority of the contact list has around 50% males. However, looking at these distribution along would not tell us whether the distributions have any bias. For this purpose, we need to compare them with the null hypothesis.

We generate the null distribution by assigning the gender of a name randomly following the gender distribution of the *contact list*. As a result, for a contact list with the number of identified names s equals to i and a probability of male of p_m, the probability that this list has j males is the binomial:

$$p(m = j|s = i) = C_i^j p_m^j (1 - p_m)^{i-j}.$$

Since the number of identified names varies for different contact lists, the probability of having a ratio of $x \in [0, 1]$ male in the contact lists is:

$$p(x) = \frac{\sum_{i=1}^{21} \sum_{k=i*x \text{ } is \text{ } an \text{ } integer} p(s = i)p(m = k|s = i)}{\sum_{i=1}^{21} \sum_{j=1}^{i} p(s = i)p(m = j|s = i)}. \quad (1)$$

Here $p(s = i)$ is the percentage of contact lists having exactly i gender identifiable names. Fig. 5 (left) shows that the distributions based on the null hypothesis (the two higher curves) are spikier, with around 30% of the contact lists having 50% of males, compared with the observed 15%. Fig. 5 (right) shows the deviation of the observed distribution from the null hypothesis. It shows a clear bimodal pattern, confirming our observation that contact lists on average exhibit a bias towards either male domination, or female domination, especially the latter.

To further verify the gender bias in observed contact lists, we model the observed number of males in all contact lists as a Binomial mixture model. Basically we assume that number of males in a contact list that we observe is generated by one of two separate Binomial distributions with different parameters, one representing female users and the other representing male users. We run Expectation-Maximization algorithm to find the best set of model parameters that explains the observed data most accurately. Here we only consider contact lists with more than 5 identifiable genders. After the EM algorithm converges, we generate synthetic data from the model and plot it alongside with the observed data in Figure 5. We observe that model fits the observed data quite well. Also the parameters of the fitted model suggest a strong gender-bias in contact lists, such that the probability of a contact in a male user's contact lists being male is 0.61, whereas the probability of a contact in a female user's contact lists being male is 0.27. The results also suggest that 47% of the observed contact lists belong to male users and 53% belongs to female users.

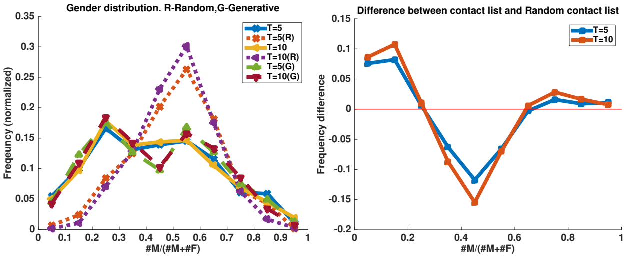


Figure 5: Left: the distribution of user's gender in contact lists data. Distributions with legend "R" are from binomial distribution with probability 0.5. Distributions with legend "G" are from binomial mixture model with parameters inferred using EM algorithm. Others are from observation in the *contact lists*. Right: deviation from the null hypothesis.

5.2 Ethnicity Distribution

While first names often reveal gender, last names give a strong signal about ethnicity. In this subsection, we study the ethnicity distribution of the *contact lists*. We use *Census 2000* data set as ground truth and perform a similar look-up classification as we did for the gender analysis.

Just like the case for gender, we observe that users' contact lists on average exhibit a bias towards one ethnicity. To substantiate this observation, we look at the frequency distribution of percentage of a particular ethnicity in our mailing list, and compare with the null hypothesis. The null hypothesis is constructed just like in the case for genders. Take the Hispanic ethnicity as an example. They constitute 14.75% of the names in the *contact lists*, so we set $p_m = 0.1475$ in (1). This allow us to plot the distribution for the null hypothesis, and compare with the observed Hispanic distribution. Fig. 6 shows the deviation of the observed distribution from the null hypothesis for "Black", "API" and "Hispanics" ethnic groups. It confirms that the *contact lists* have a tendency of containing lists that have higher than expected percentage of one of these ethnic groups, even though the bias is not quite as pronounced as in the case of genders.

6 IDENTITY CHALLENGE USER STUDY

Our work is motivated by the need to generate realistic looking replacement names for contact list challenges. Our running assumption has been that by generating background names in the contact list challenges using name embedding, we preserve the cultural similarity of the real and imitation contact names, and make it harder for would be hackers to identify the real contact name. To test this hypothesis, we conducted a controlled user experiment using the Amazon Mechanical Turk service.

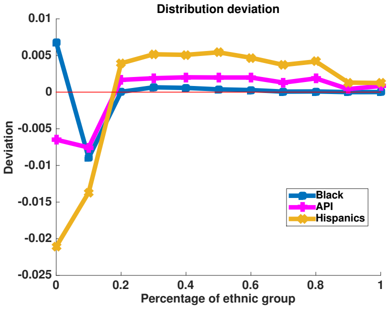


Figure 6: Deviation between observed distribution of percentage of names in ethnic groups “Black", “API" and “Hispanics", and the distribution from the null hypothesis, showing a bimodal pattern.

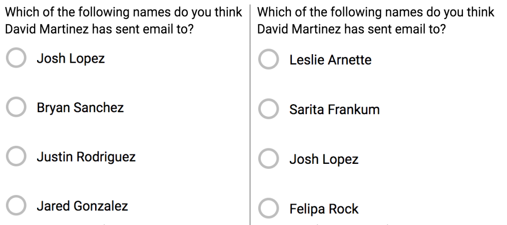
6.1 Data and Tasks

For our controlled user experiment, we need to construct contact list containing the email account owner, a real contact person of the account owner, and a few imitation contacts. To limit cognative load of the test takers and allow them to finish a set of questions in a reasonable amount of time, we limit the number of total contacts to four, with the contact names randomly ordered. Note that in real contact list challenge, to further reduce the probability of success by random guess, one could use more than four contacts.

To preserve user privacy, we do not use real email account owner names. Instead, we do the following. We randomly choose 200 email account owners. For each user, we find the two most frequent contacts, C_1 and C_2. We generate four similar names of C_2, denoted as D_1, D_2, D_3 and D_4, using name embedding. We construct two types of contact list challenges, and both treat C_1 as the account owner, D_1 as the real contact. The first type generates imitation names using name embedding, while the other type using randomly generated names.

- **Embedding based contact list (method A):** we treat D_2, D_3 and D_4 as background contacts.
- **Random contact list (method B):** we randomly generate three names using http://onerandomname.com as imitation contacts.

Table 7 shows a sample pair of test questions using methods A/B. To guide the user, we start the test by asking “In the following, the email account owner sent emails to only one of the four persons. Do your best to guess which one is that person.”.



Which of the following names do you think David Martinez has sent email to?

○ Josh Lopez

○ Bryan Sanchez

○ Justin Rodriguez

○ Jared Gonzalez

Which of the following names do you think David Martinez has sent email to?

○ Leslie Arnette

○ Sarita Frankum

○ Josh Lopez

○ Felipa Rock

Figure 7: Sample test questions. The real contact is “Josh Lopez”. Left: embedding based-imitation contacts. Right: randomly generated-imitation contacts.

Table 8: Percentage of success in the MTurks tests. Four different forms are used. Each form has 50 questions.

Form	#MTurks	#Questions	Embedding	Random
1	25	625	25.44%	62.56%
2	50	1250	24.48%	53.12%
3	49	1225	26.37%	66.04%
4	26	650	29.23%	71.85%

Table 9: Success rate among different age groups

age	21-30	31-40	41-50	51-60	other ages
#MTurks	44	59	22	16	10
embedding	26.1%	26.6%	27.1%	27.2%	24.0%
random	48.7%	69.8%	66.5%	64.5%	63.4%

6.2 Experiment Design

To maximize the utilization of each subject and reduce variation due to different subjects, we applied a within-subject design in the tests using methods A/B. Each subject is given 50 contact list challenge questions, of which 25 are generated using our method (A), and 25 using the random method (B). The 50 questions are randomly ordered. To reduce variation due to a specific group of 50 questions, and to make sure each MTurk only take one side of a pair of A/B tests, we generated 4 sets of 50 questions. We call each set a form. In addition to the test questions, we ask some trivial questions to make sure the MTurks are actually reading the questions and attempt to answer the questions responsiblly.

For each form, we attempted to recruit 50 MTurks, with rewards between \$0.75 and \$1 per form. After posting these four forms on Amazon for 2 weeks, we ended up getting 25, 50, 49, 26 MTurks for each of the forms, respectively. All MTurks answered the human verification questions correctly, therefore we treat all answered forms as valid.

Table 8 gives the results of the MTurk tests. The success rate by random guess should be 25%. We test the null hypothesis that the method under consideration behaves similar to random guesses. As we can see, the success rates of the MTurks on randomly generated contact list challenge are very high, with a weighted average success rate of 62.16%. This gives a p value of 1.11×10^{-16}, indicating that MTurks can identify the real contact with probability much better than explained by random guess. On the other hand, the weighted average success rate of our method is 26.08%, giving a $p = 0.062$ (binomial test), not invalidating the null hypothesis. We conclude that the rate of success for our method is close to random guesses, and therefore embedding based contact list challenges are much more secure than randomly generated contact list challenges.

We asked test takers for their gender, age and ethnicity. We found that 60% of the MTurks are male. 80% are of white ethnicity. The 21-30 age group appears to be less successful in identifying the real contacts among the randomly generated imitation contacts (Table 9). No significant difference is seen on contact list challenges generated with the proposed name embedding method.

6.3 Real User Test

We conducted a user study on 1120 random email users from a major Internet company to establish whether actual users can pass this test on their own accounts. For each email user, we showed a contact list consists of 1 real contact and 7 imitation contacts generated using our method. The purpose of the study is to see whether *genuine* users can pick out their real contact from among 7 imitation contacts. The experiment was reviewed and approved by the appropriate institutional review board (IRB). During our experiment, 983 users (88%) were able to pick out their real contacts correctly. We conclude that these validation rate is high enough for practical use, particularly consider the organization of our study. According to the Internet company's data, a randomly challenged user only have 35% chance pass the other types of challenges. Given this background, we believe that the observed success rate establishes that the proposed contact list challenge is a viable tool for security challenge. Furthermore, when using contact list challenge for real applications, one can combine multiple contact list challenges, and require the user to successfully pass a majority of them. This reduces the chance of success by random guessing, denying instant feedback to the hacker while allowing the real user to gain access even if she fails one particular challenge.

7 DE NOVO NAME GENERATION

Our primary interest in this paper concerns *replacement name generation*, where given a particular name (f, l) we seek to construct a replacement name (f', l') with similar properties and veracity. However, a related class of applications concerns generating large sets of plausible names without any starting templates, to serve as demonstration identities in information processing systems.

listofrandomnames.com		Embedding-based de novo generation	
Male	Female	Male	Female
Keith Albro	Sibyl Bjork	Reginald Bouldin	**Ethel Agnew**
Sonny Bordner	**Amie Corrao**	**Max Bowling**	**Mabel Beaudoin**
Stanley Brummond	**Joselyn Custard**	**Dale Depriest**	Jolanda Boring
Reuben Carlucci	Marvella Deese	**Richard Diefenderfer**	**Lori Butz**
Darrell Chatmon	Holly Delman	**Michael Doutt**	**Diana Chao**
Jeffry Egnor	Kayleigh Derr	**Randall Drain**	**Cynthia Clay**
Russel Foye	Eugenia Fahnestock	**Anthony Hattabaugh**	Karin Combes
Hank Fries	Clemmie Formica	**Henry Humbert**	**Krista Emmons**
Patrick Gazaway	Gigi Fredericksen	**Jeremy Jacobsen**	**Rebecca Gagnon**
Roy Gilman	Marylyn Gersten	**Jeffrey Jimenez**	**Betty Grant**
Federico Gulley	**Elisabeth Harkness**	**Brian Kerns**	**Ruth Griffin**
Adalberto Hakes	**Almeda Ivy**	**Ronald King**	**Nancy Lantz**
Sylvester Kammer	Dot Klingbeil	**Elton Kolling**	**Joann Larsen**
Tanner Lundblad	Shay Krom	**Robert Kuhls**	**Deborah Lovell**
Jarod Man	Tessie Kush	**Fred Lawyer**	**Carla Mccourt**
Lee Mcclintock	Providencia Laughter	**Raymond Middleton**	**Caroline Mclaney**
Elvin Mcwhirt	Merlyn Lovings	**Andres Morales**	Denise Murders
Harry Nino	Milda Marcos	**John Morales**	**Mary Navarro**
Preston Pickle	Sierra Olivieri	**Alvin Morrison**	**Margarita Reyes**
Edgar Ramer	Pennie Pasquale	**Patrick Mulvey**	**Brenda Rock**
Rafael Rasheed	**Mallory Peralta**	**Victor Rahn**	**Selina Rubin**
Earnest Robert	Manda Stetz	**Nick Shick**	**Opal Sinkfield**
Ryan Seiber	Lissette Torrey	**Howard Siegel**	**Denise Stephens**
Kraig Tullos	Zelda Vanderburg	**Daniel Spady**	**Doretha Thurmond**
Howard Welk	Hee Weast	**Patricia Vargas**	Serina Webb

Table 10: Comparison of our de novo generated synthetic names and random names from website http://listofrandomnames.com. We count the number of returned results by using Google search engine with exact name string as input. Bold names have more than 100 matches, while red colored names don't have any matches.

Indeed, several de novo name generation looks are available on the web, like http://listofrandomnames.com, which randomly combine pairs of first and last names. These systems may or may not respect component frequencies in the population. However they generally do not employ linkage information between given and family names, resulting in implausible combinations.

A synthetic name generation algorithm should have the following properties:

- *Scale* – The algorithm should be able to generate an arbitrarily large number of names, without high levels of repetition.
- *Respect population-level frequency-of-use statistics* – First name and last name tokens should be generated as per names in the target population.
- *Culturally-appropriate first/last-name linkage* – Name token usage is not independent, but conditionally linked.
- *Privacy preservation* – No linkage between real and synthetic identities is permitted.

Simultaneously satisfying all these properties is non-trivial. Linking first names to the nearest last names in embedding space violate the requirement for scale. Random generation from component lexicons violates the second requirement without auxiliary information. Even with proper sampling, random generation fails to satisfy the cultural linkage requirement. Drawing full names from reference sources like telephone directories or the fake-name replacement strategies run afoul of privacy preservation.

We propose the following approach. We construct a batch of m names simultaneously, where $m = 100$ is an appropriate value. We randomly sample m first and last name components as per the population distribution, here generated according to the U.S. Census distribution. We use the embedding-similarity between name components to weigh a complete $m \times m$ bipartite graph. By computing a maximum weight bipartite matching, we get m synthetic names with linkage informed by the geometry of the name embedding.

Table 10 compares the first 25 synthetic men and women names produced by our methods with http://listofrandomnames.com. We conducted a study by searching for each of the full names in Google and checking how many results are returned. Our rationale is that a plausible name should appear more often on the web than an implausible one. In the table, we marked in bold names that has at least 100 matches in Google search. In addition we use red color to show names that have no matchs at all. Clearly our name generator performs much better, with 47 bold names to 18 for http://listofrandomnames.com.

8 CONCLUSION

Motivated by the need to create imitation names in contact list based security challenges, we propose a new technique for generating look-alike names through distributed name embeddings. By training on millions of email contact lists, our embeddings establish gender and cultural locality among names. The embeddings make possible construction of replacement aliases for any given name that preserve gender and cultural identity. Through large-scale analysis of contact lists, we established that there is a greater than expected concentration of names of the same gender and race for all major groupings under study. We conduct a controlled user study via Amazon Mechanical Turk, comparing randomly generated contact list challenges with challenges constructed using name embedding. The study demonstrated the effectiveness of the latter.

Using the techniques developed in this paper, we have constructed a collection of synthetic names, which will be released as an open resource upon the publication of this manuscript.

For future work, we plan to incorporate the proposed algorithm for generating contact list challenges as part of the security challenges for users who, for one reason or another, are not suitable subjects for the traditional two factor authentication.

9 ACKNOWLEDGMENTS

We thank the anonymous reviewers for their valuable feedback and suggestions on our work. This work is partially supported by DBI-1355990 and NFS Grants IIS-1546113.

REFERENCES

[1] Anurag Ambekar, Charles Ward, Jahangir Mohammed, Swapna Male, and Steven Skiena. 2009. Name-ethnicity classification from open sources. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge Discovery and Data Mining*. ACM, 49–58.

[2] Yoshua Bengio, Aaron Courville, and Pierre Vincent. 2013. Representation learning: A review and new perspectives. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 35, 8 (2013), 1798–1828.

[3] Joseph Bonneau, Elie Bursztein, Ilan Caron, Rob Jackson, and Mike Williamson. 2015. Secrets, Lies, and Account Recovery: Lessons from the Use of Personal Knowledge Questions at Google. In *Proceedings of the 24th International Conference on World Wide Web (WWW '15)*. ACM, New York, NY, USA, 141–150. https://doi.org/10.1145/2736277.2741691

[4] Joseph Bonneau, Cormac Herley, Paul C. van Oorschot, and Frank Stajano. 2015. Passwords and the Evolution of Imperfect Authentication. *Commun. ACM* 58, 7 (June 2015), 78–87. https://doi.org/10.1145/2699390

[5] Census Bureau. 1990. https://www.census.gov/topics/population/genealogy/data/1990_census/1990_census_namefiles.html. (1990).

[6] Census Bureau. 2000. https://www.census.gov/topics/population/genealogy/data/2000_surnames.html. (2000).

[7] Elie Bursztein and Ilan Caron. 2015. https://security.googleblog.com/2015/05/new-research-some-tough-questions-for.html. (2015).

[8] Mike Campbell. 1996. http://www.behindthename.com. (1996).

[9] J. Chang, I. Rosenn, L. Backstrom, and C. Marlow. 2010. ePluribus: Ethnicity on social networks. In *Proceedings of the International Conference in Weblogs and Social Media (ICWSM)*. 18âĂŞ25.

[10] David Freeman, Sakshi Jain, Markus Dürmuth, Battista Biggio, and Giorgio Giacinto. 2016. Who Are You? A Statistical Approach to Measuring User Authenticity. In *23nd Annual Network and Distributed System Security Symposium, NDSS 2016, San Diego, California, USA, February 21-24, 2016*. The Internet Society, 1–15. http://www.internetsociety.org/sites/default/files/blogs-media/who-are-you-statistical-approach-measuring-user-authenticity.pdf

[11] Ralph Gross and Alessandro Acquisti. 2005. Information revelation and privacy in online social networks. In *Proceedings of the 2005 ACM workshop on Privacy in the electronic society*. ACM, 71–80.

[12] J. Andrew Harris. 2015. What's in a Name? A Method for Extracting Information about Ethnicity from Names. *Political Analysis* 23, 2 (2015), 212–224.

[13] Yifan Hu, Emden Gansner, and Stephen Kobourov. 2010. Visualizing graphs and clusters as maps. *IEEE Computer Graphics and Applications* 30 (2010), 54–66.

[14] Mike Just. 2004. Designing and Evaluating Challenge-Question Systems. *IEEE Security & Privacy* 2, 5 (2004), 32–39. https://doi.org/10.1109/MSP.2004.80

[15] Omer Levy and Yoav Goldberg. 2014. Neural Word Embedding as Implicit Matrix Factorization. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*. 2177–2185.

[16] P. Mateos, R. Webber, and P. Longley. 2007. *The cultural, ethnic and linguistic classification of populations and neighbourhoods using personal names*. Technical Report CASA Working Papers 116. Centre for Advanced Spatial Analysis University College London.

[17] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. In *Proceedings of Workshop at ICLR*.

[18] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*. 3111–3119.

[19] Alan Mislove, Sune Lehmann, Yong-Yeol Ahn, Jukka-Pekka Onnela, and J Niels Rosenquist. 2011. Understanding the Demographics of Twitter Users. *ICWSM* 11 (2011), 5th.

[20] open source project. 2013. https://code.google.com/archive/p/word2vec/. (2013).

[21] Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. *Proceedings of the Empiricial Methods in Natural Language Processing (EMNLP 2014)* 12 (2014).

[22] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. 2014. Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 701–710.

[23] Pucktada Treeratpituk and C. Lee Giles. 2012. Name-Ethnicity Classification and Ethnicity-Sensitive Name Matching. In *Proceedings of AAAI Conference on Artificial Intelligence*.

[24] Laurens Van Der Maaten. 2014. Accelerating t-sne using tree-based algorithms. *The Journal of Machine Learning Research* 15, 1 (2014), 3221–3245.

[25] David L Word, Charles D Coleman, Robert Nunziata, and Robert Kominski. 2008. Demographic aspects of surnames from census 2000. *Unpublished manuscript, Retrieved from http://citeseerx. ist. psu. edu/viewdoc/download* (2008).