Damegender: Writing and Comparing Gender Detection Tools

David Arroyo Menéndez

June 4, 2019

Abstract

Nowadays there are various APIs to detect gender from a name. In these slides, we offer a tool to use and compare these apis and a method to classify male and female applying machine learning and using a free license. The gender detection from a name is useful to make gender studies from social networks, mailing lists, software repositories, articles, etc.

Download source and article to a make a good tracing

- git clone `https://github.com/davidam/damegender.git`

Social Need (I)

Traditional approaches for inferring the gender given a name are based on the use of census data and specific APIs that tend to use also census data. I propose to use ML techniques to complement them, so that it can be applied to nicknames, new names, diminutives, etc. that usually do not appear in census data. The underlying assumption that lead me to this approach is that we humans tend to have a certain intuition, that commonly works, to infer the gender of a name even if it is the first time we see it

Social Need (II)

In this moment there are a gender gap between males and females in computer science and science in general (STEMM: Science, Technology, Engineering, Mathematics and Medicine). Create free tools and improve the current state of art allows measure and later create policies with facts to fix the situation.

Underlying Technologies

- Scikit
- NLTK
- Numpy
- Matplotlib
- Perceval

APIs with Datasets

```
$ python3 api2gender.py Laura --surname="Cornejo" --api=namsor
female
scale: 1.0
```

- Poor explicative power
- Peer review only with black box tests
- You need pay if you want compute many data (more expensive)
- You can compute from local massive data (fast)

Census Open Data

```
$ python3 main.py David --total="ine"
David gender is male
363559  males for David from INE.es
0 females for David from INE.es
```

- Precise explanation from a Statistical Institute
- You have the possibility to make peer review in a specific geographical area

Selecting components with PCA

first_letter	last_letter	last_letter_a	first_letter_vocal	last_letter_vocal	last_letter_consonant	target component
-0.2080025204	-0.3208958517	0.2352509625	0.2113242731	*0.6095269139*	*-0.6095269139*	-0.1035071139
-0.6037951881	*0.5174873789*	-0.4252467151	0.4278794455	0.0388287435	-0.0388287435	-0.0265942125
0.1049343046	0.1158117877	-0.2867605971	-0.3473950734	0.0901034539	-0.0901034539	-0.8697264971
0.2026467275	0.3142402839	*0.630802294*	*0.5325769702*	-0.1291229841	0.1291229841	-0.3811720011

In this analysis, we can observe 4 components.

The first component is about if the last letter is vocal or consonant. If the last letter is vocal we can find a female and if the last letter is a consonant we can find a male.

The second component is about the first letter. The last letter is determing females and the first letter is determing males.

The third component is not giving relevant information.

The fourth component is giving tha $last_{lettera}$ and the $first_{lettervocal}$ is for females.

Give me informative features

```
$ python3 infofeatures.py
Females with last letter a: 0.4705246078961601
Males with last letter a: 0.048672566371681415
Females with last letter consonant: 0.2735841767750908
Males with last letter consonant: 0.6355328972681801
Females with last letter vocal: 0.7262612995441552
Males with last letter vocal: 0.3640823393612928
```

- A female distinguish feature is the last letter a.
- A male distinguish feature is the last letter consonant.

Some accuracies

Way to guess a string	Accuracy
Namsor	0.753957037874505
Genderize	0.71537591859807
Support Vector Machines	0.704918032786885
Gender Guesser	0.690220463538722
NLTK Bayes	0.667750141322781
Gaussian Naive Bayes	0.596099491237987
Multinomial Naive Bayes	0.587620124364047
Stochastic Gradient Descendent	0.587337478801582
Bernoulli Naive Bayes	0.596240814019219

With Machine Learning we can guess nicknames, new names, or diminutives

Proof of Concept in Repositories

```
$ python3 git2gender.py https://github.com/chaoss/grimoirelab-p
The number of males sending commits is 15
The number of females sending commits is 7
```

Proof of Concept in Mailing Lists

```
# Count gender from a mailing list
$ cd files/mbox
$ wget -c http://mail-archives.apache.org/mod_mbox/httpd-annour
$ cd ..
$ python3 mail2gender.py http://mail-archives.apache.org/mod_mb
The number of males sending mails is 6
The number of females sending mails is 0
```

Future Work

Damegender is a tool to research in gender gap. So, the future work is to understand the massive gender gap with an empirical approach.
The public mailing list and software repositories is a big public data source in this sense.

Conclusions

The market of gender detection tools is dominated by companies based on payment services through APIs. This market could be changed thanks to free software tools and open data due to give more explicative results for the user. Although the machine learning techniques is not new in this field, it's an incentive for researchers in computer science create free software tools.

These advances in computer science could be giving support to study the gender gap in repositories and mailing lists.