

# Demonym gazetteer

Alex Pardo and David Sanchez

29<sup>th</sup>, May 2014

# Contents

0.1	Introduction . . . . .	2
0.2	Attached files . . . . .	2
0.3	Description of the problem . . . . .	2
0.4	Approach . . . . .	3
0.5	Evaluation . . . . .	4
0.6	Discussion . . . . .	4
0.7	Future work . . . . .	4

## 0.1 Introduction

Demonym or gentilic, is a term for the residents of a locality. How to generate the demonym from a country or city is not an easy task, and in order to automatize this process we are going to build a system which it is capable to do this automatically. The next document will show the description of the problem, our system approach, the evaluation and the discussions of the results. Finally, we will explain some possible improvements in our work as a future work.

## 0.2 Attached files

Identifiers of the attached files (classes, functions and script .py files and data files when needed)

## 0.3 Description of the problem

As the document mentions in the introduction, the generation of the demonym from a location like country or city it is not a easy task. Exist a lot of possible rules and different cases.

Some examples of this process are:

- Africa – African
- Croatia – Croatian (also "Croat")
- North / South Korea – North / South Korean
- Hanoi (Vietnam) – Hanoian
- Iran – Iranian (also "Irani" or "Persian")
- Florence – Florentine (also Latin "Florentia")
- Ann Arbor – Ann Arborite
- Israel – Israelite (also "Israeli")
- Netherlands – Netherlander

For instance, as the examples of the above show, some demonyms are simple a process in which is needed to add ending letters in the original country or city. However, the irregular cases difficult the process to build the system because others constructions are based in delete some letters in the original word and add some ending new letters. Also, other problem to face is that some countries and cities have more than one demonym and this fact difficult the process to build a robust system.



An example of this two types of rules could be the Africa - African. From this example the system will built these rules:

- "a" - "an"
- " " - "n"

The system construct all these types of rules for all the examples extracted from Wikipedia and build a graphical representation with histograms [2]. The histogram filter the rules which they occurs rarely and it only shows the most popular rules.

The second phase is the test phase. First, the system download new countries and cities names (without demonym). The idea is based on generate automatically for each country or city its demonym. For do this, the system uses the rules knowledge (generated in the first step) and uses the most probable combination of the two different types of rules.

## 0.5 Evaluation

The evaluation performed

## 0.6 Discussion

A discussion of the results

## 0.7 Future work

A proposal of possible improvements