# Social Media Text Processing and Semantic Analysis for Smart Cities

## MSc Dissertation Viva

João Pereira

**Supervision:**

Rosaldo Rossetti

Pedro Saleiro

U.PORTO

FEUP FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO

LIACC

July 14, 2017

# Agenda

1. Scope
2. Problem Statement
3. Goals
4. State of the Art
5. Framework
6. Exploratory Data Analysis
7. Text Analytics Experiments
8. Conclusions

# Scope

- Instant connectivity 24/7
- Sharing of events, opinions and activities
- Many research areas have tried to exploit social media data
- Smart Cities and Intelligent Transportation Systems are also obvious candidates
- Information derived from such exploration may bring benefits to the cities' governance, traffic-flow management, etc.

# Problem Statement

Mining Twitter data is a laborious and time-consuming process.

a) Social media platforms have its own specificities

b) The volume of data retrieved is overwhelming

c) Social Media content with several restrictions

# Goals

1. Design and development of a framework for continuous collection of tweets from multiple bounding-boxes

2. Discovering of latent topics in the Twitter data

3. Travel-related classification of tweets using supervised learning

4. Aggregation and visualization of results

# State of the Art

1. Lansley et al. [4] depicted topic modelling over geo-located tweets in London and cross the results with geographic maps to identify land-use patterns

2. Maghrebi et al. [5] classified travel-mode geo-located tweets using term-based search in Melbourne.

3. Kuflik et al. [6] performed multi-class transport classification over geo-located tweets from Liverpool.

# Framework

The framework may contemplate the following requirements:

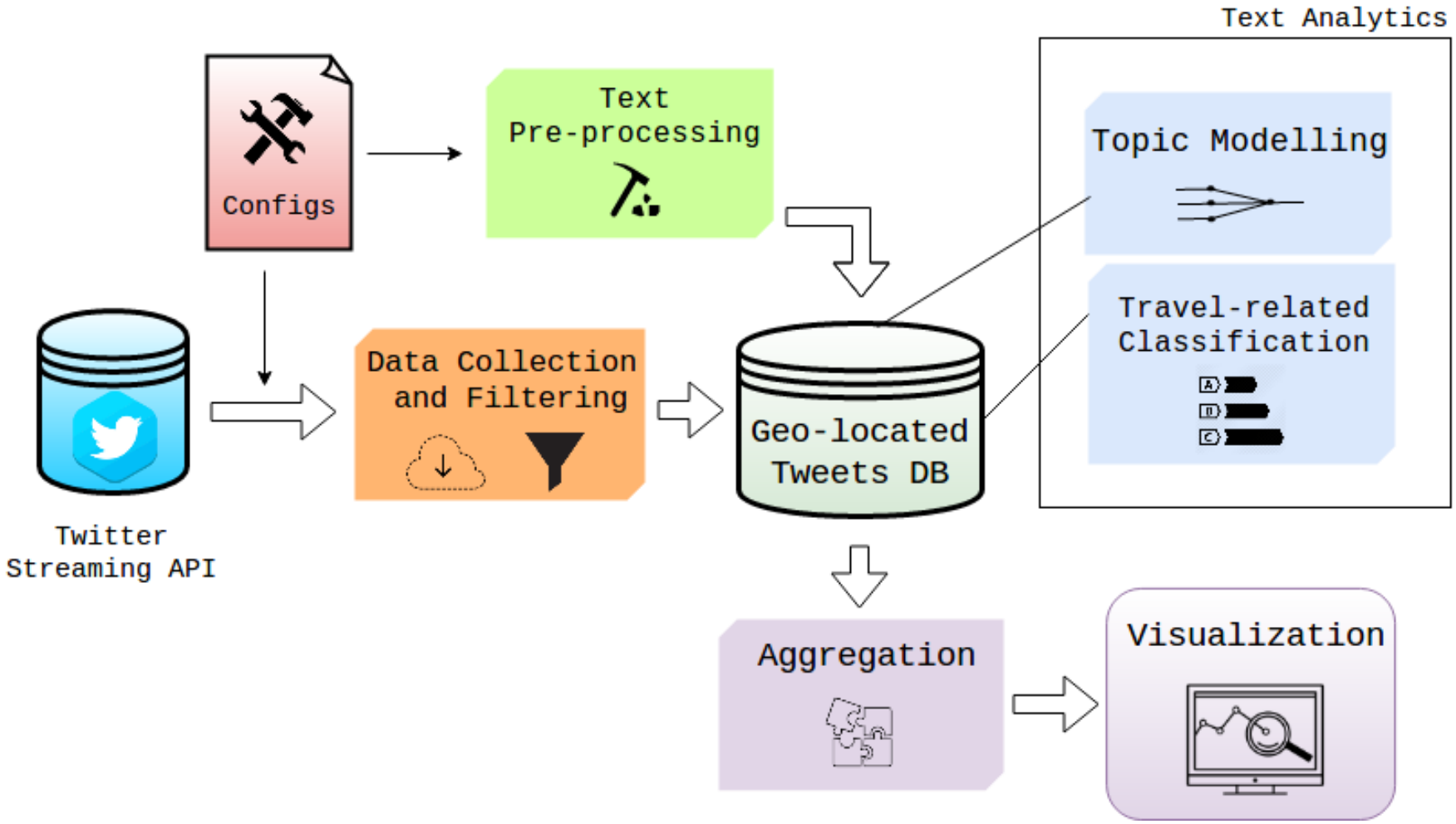
a) Collection of multiple bounding-boxes (cities, regions or countries)

b) Flexibility in each module

c) Scalability

d) Results must be shown on-time to the final user

# Architecture Overview

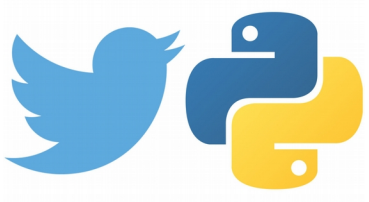


# Data Collection

Twitter Streaming API allows three different collection heuristics:

- Term-based search;
- Follow users activity;
- **Locations through bounding-box system**.

Tweepy, a open-source Python library to acess the Twitter APIs.



Preliminary test performed on Tweepy to verify limits of the API.

A module to support parallelization for collecting geo-located tweets in several bounding-boxes .

# Data Filtering

The data retrieved has two inconsisitencies:

- A large amount of tweets in different idioms comparative to the one spoken in the target city;
- Tweets from outside the searching bounding-box are retrieved by the API.

"The **black** square represents the searching bounding-box, while the **red** square represents a tweet with *place_id* "Nova Iguaçu" and the **green** one a tweet with place_id "Duque de Caxias". The tweet associated to the **red** square is filtered out of the final dataset."

# Database

- Tweets are retrieved in JSON-like format;


- MongoDB is a NoSQL software to store collections/schemas of this data format;
- The overwhelming amount of tweets makes conventional querying operations to perform poorly;
- On the contrary, MongoDB has high performance regarding the querying system.
- Nonetheless, this software allows easier and quickly deployment and scalability.

# Text Pre-processing

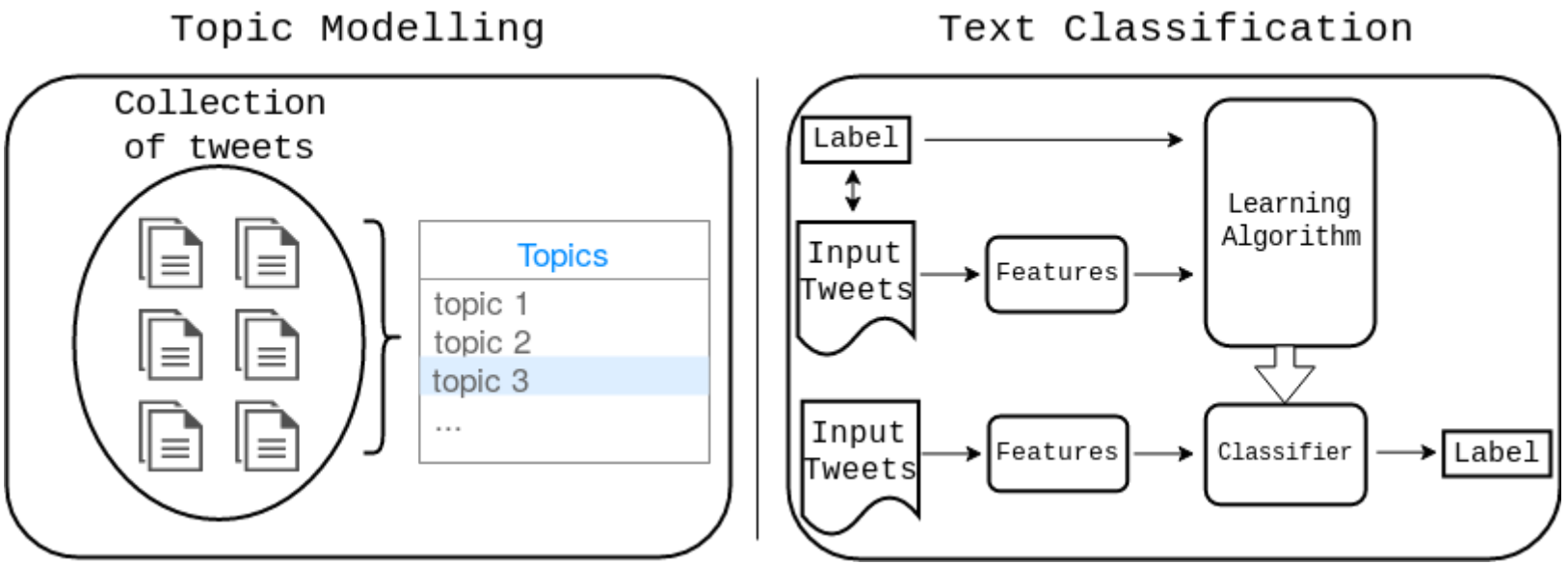
We apply a considerable group of text pre-processing operations to the messages.

- Lower casing;
- Lemmatization;
- Tokenization;
- Transformation of repeated characters;
- Punctuation removal;
- Cleaning of *metadata* and numerical symbols in the text;
- Stop and short words removal.

# Text Analytics

Two different approaches in the implementation of the text analytics modules:

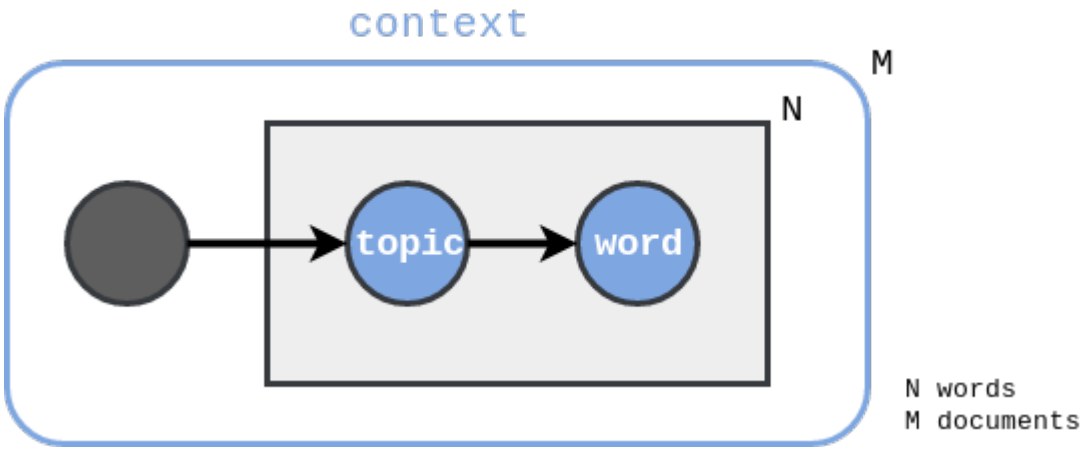
- Unsupervised approach – Topic Modelling;
- Supervised approach – Text Classification.



# Topic Modelling

**Latent Dirichlet Allocation (LDA) - Blei et al. [1]**

- Generative probabilistic model;
- Aims to find the latent topics present in a collection of documents;
- Two distinct distributions:
  - Distribution of words over topics;
  - Distribution of topics over documents.
- We use the Python's LDA library to implement this text analytics module.



# Topic Modelling - Example

**Topic 1:**

**20 most frequent words**
paulo, vai, hoje, dia, jogo, ser, melhor, time, vamo, brazil, todo, santo, brasil, gol, cara, aqui, agora, corinthiam, ano, palmeiro, vem

Documents:
382,479 tweets in Rio de Janeiro

Topic 1 has the label “Sports and Games”

**Topic 2:**

**20 most frequent words**
paulo, brazil, sao, santo, vila, just, parque, posted, photo, shopping, paulista, centro, bernardo, jardim, cidade, avenido, praia, santa, campo, academia

Documents:
86,519 tweets in São Paulo

Topic 2 has the label “Tourism and Places”

# Travel-related Classification

What is a travel-related tweet?

- “estou ficando acostumada ver o nascer do sol de **dentro de um ônibus** ou **estação de trem**”
- “**train** is quicker they said , get the **train** they said”

We opt for using a supervised learning approach.

This approach required two important tasks:

- Features extraction;
- Construction of gold-standard datasets due to the lack of it in open-source repositories.

# Travel-related Classification

**Bag-of-words (BoW)**

Majority of studies focus on conventional techniques

Frequency-term based text representation

Dictionary size and words belonging to documents can be limited

Messages

(1) I was in an uber yesterday.

(2) You like donuts.

Bag-of-words Representation

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| (1) | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 |
| (2) | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |

Dictionary

```
[
      "I",
    "was",
     "in",
     "an",
   "uber",
 "yesterday",
    "You",
   "like",
  "donuts"
]
```

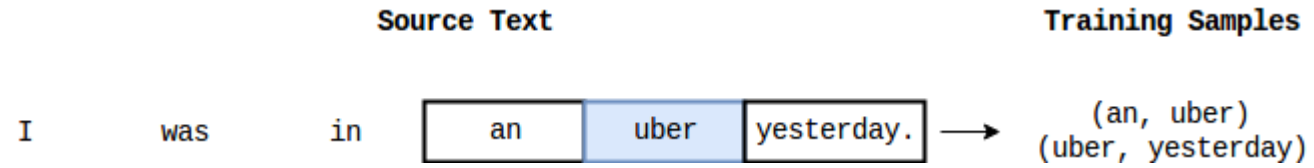
# Travel-related Classification

**Word Embeddings**

- Continuous representation of text into multi-dimensional vectors
  - *Words or phrases* are mapped to vectors of real numbers;
- Model uses:
  - Neural network
  - Context in which the word appears (surronding words, i.e. words behind and ahead)

# Travel-related Classification

## Word Embeddings - The Fake Task




Learning of statistics from the number of times each pairing shows up.

# Travel-related Classification

- ***Word2vec* – Mikolov et al. [2]**
  - Input of a collection of pre-processed tweets
  - Training of word embeddings is automatic
  - The model outputs the matrix of weights for each word in the dictionary
- ***Paragraph2vec*, a.k.a. *doc2vec* – Mikolov et al. [3]**
  - Similar to *word2vec*
  - Each document has a label passed into the embeddings matrix
  - Labels -> tweets *ids*
- **Gensim is a Python library which has both embeddings' models**

# Aggregation and Visualization

## Aggregation

- MongoDB provides inner frameworks that allow aggregation of data;
- These aggregation operations are easier and quickly because they are performed using a map-reduce paradigm
- Aggregation of results are made periodically.

## Visualization

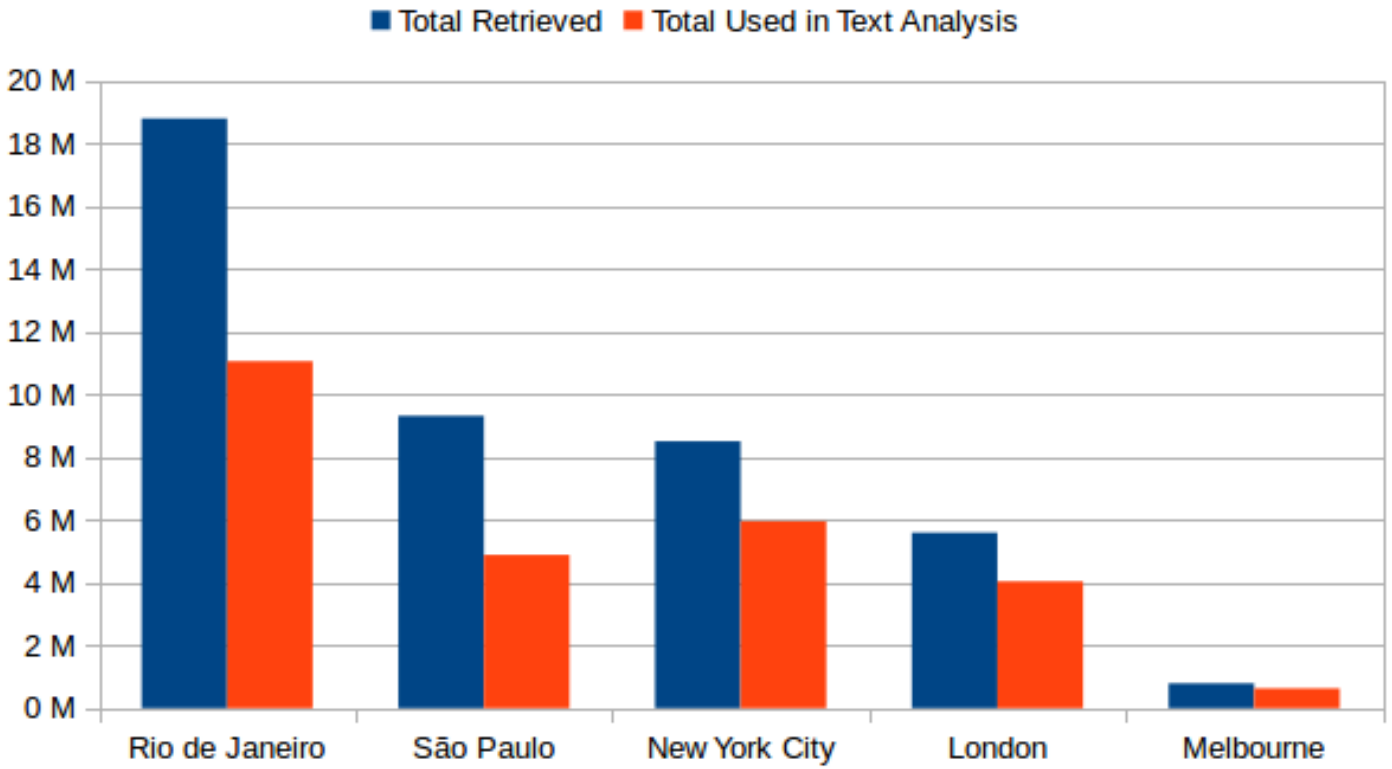
- Plotly is a Python library to produce graphical visualizations of data;
- The library allows local storage of these visualizations in HTML files;
- By exploring the embedding functionality of HTML5 we can demonstrate on-time visualizations.

# Exploratory Data Analysis

We decide to test our framework in 5 cities over the world: **Rio de Janeiro**, **São Paulo**, **New York City**, **London** and **Melbourne**.

We collect a total 43 million tweets over 3 months from March 12 to June 12, 2017.

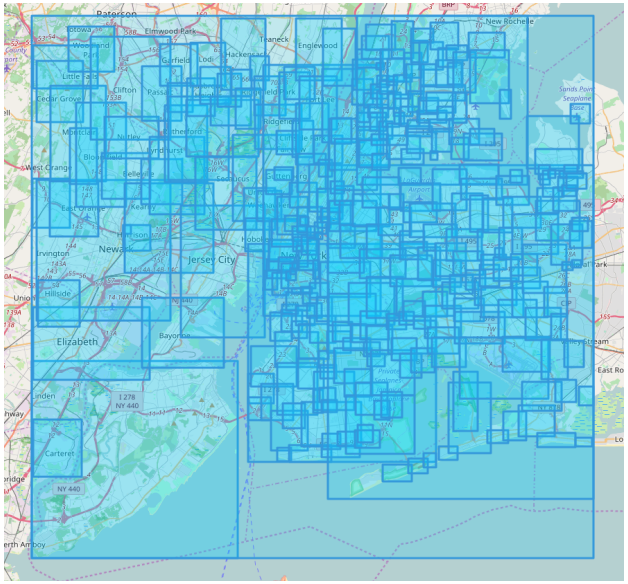
Total volume of tweets collected in each city



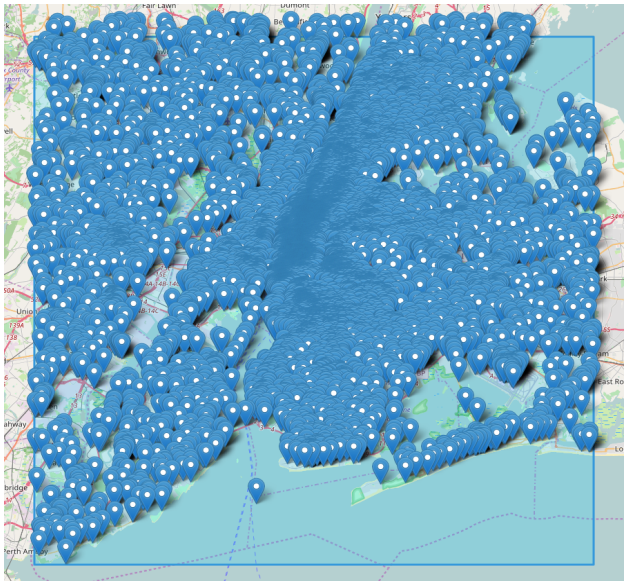
# Geographical Statistics

> 70% of geo-located tweets correspond to areas/regions

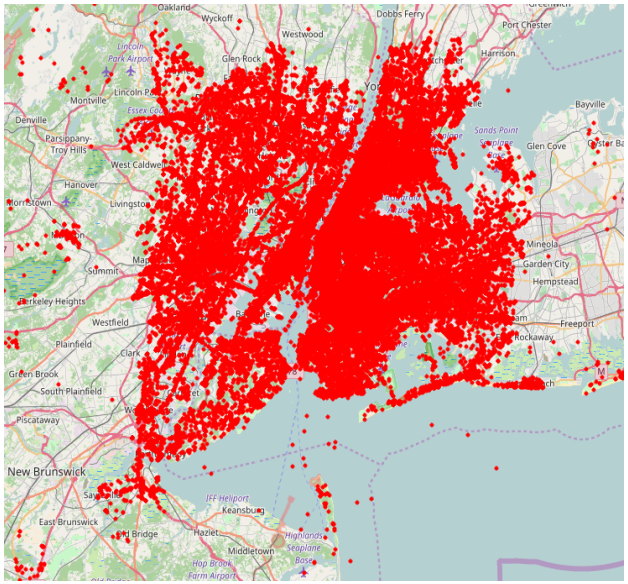
Many studies do not report this fact



Bounding-boxes of variable size



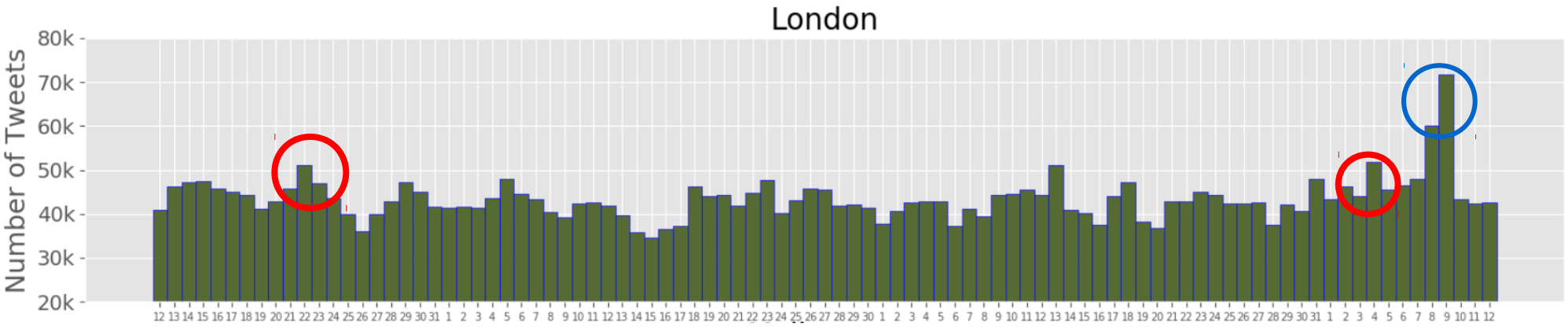
Fixed places



Precise coordinates

# Temporal Frequencies

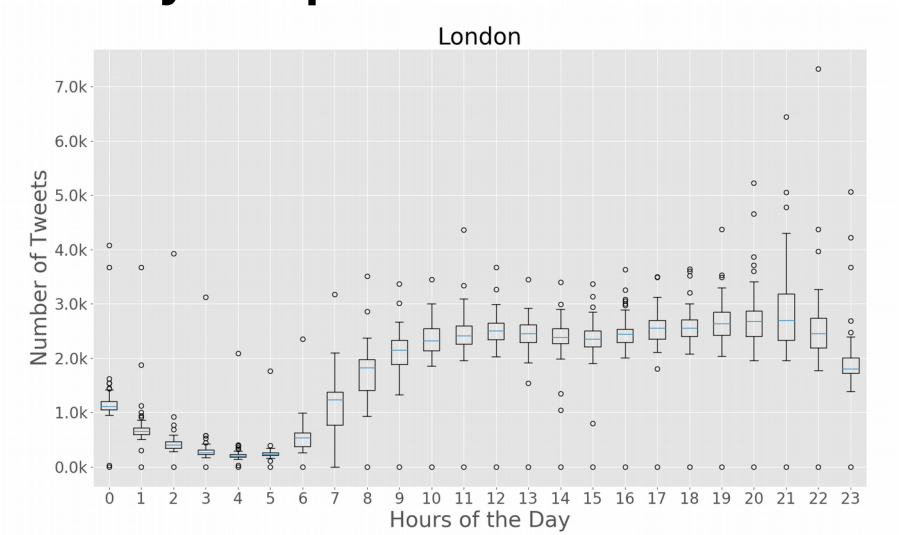
- Daily and hourly distributions of geo-located tweets allow the identification of potential remarkable events;
- The figure below shows a pike of the Twitter activity in London which is associated with United Kingdom General Elections 2017.

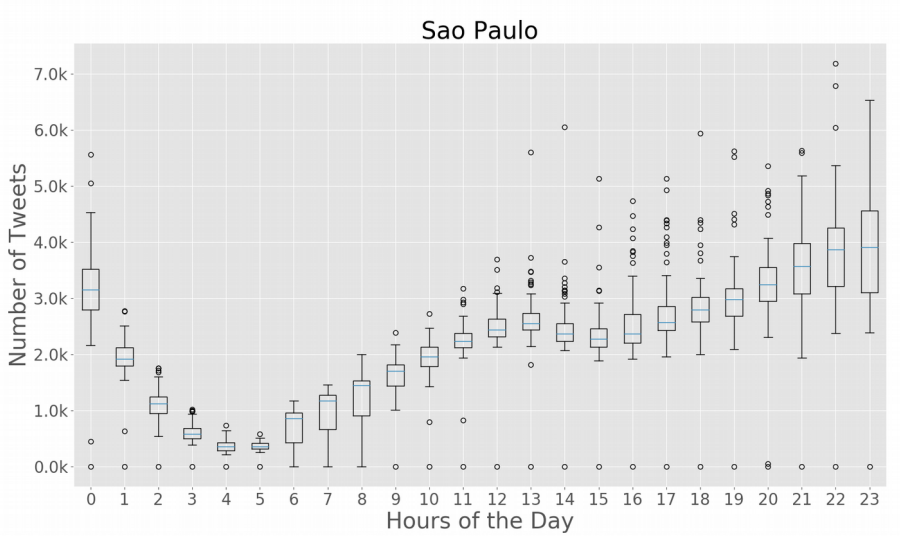


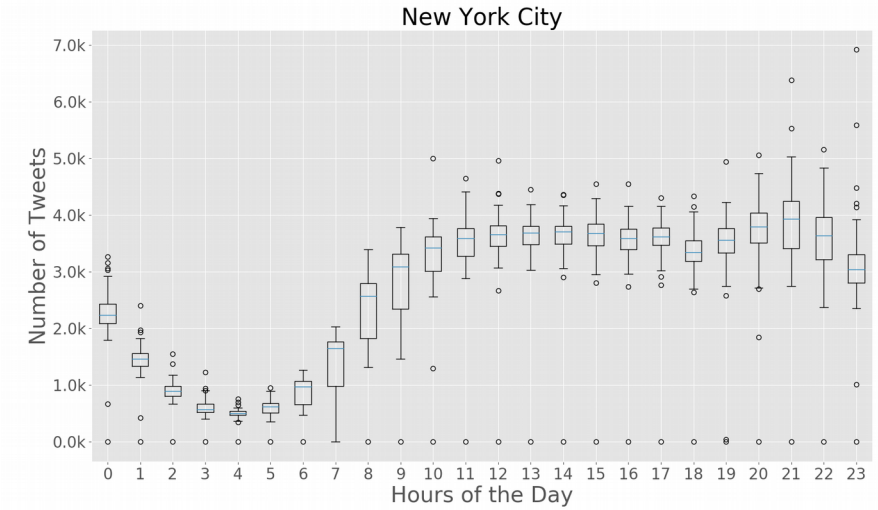

**red** – Westminster attack and London Bridge attack
**blue** – UK General Elections 2017

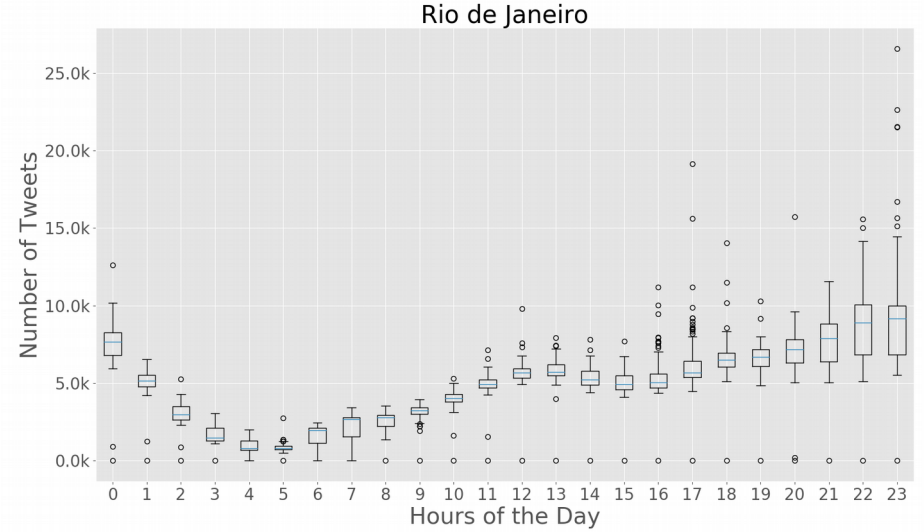
# Temporal Frequencies

## Hourly Frequencies



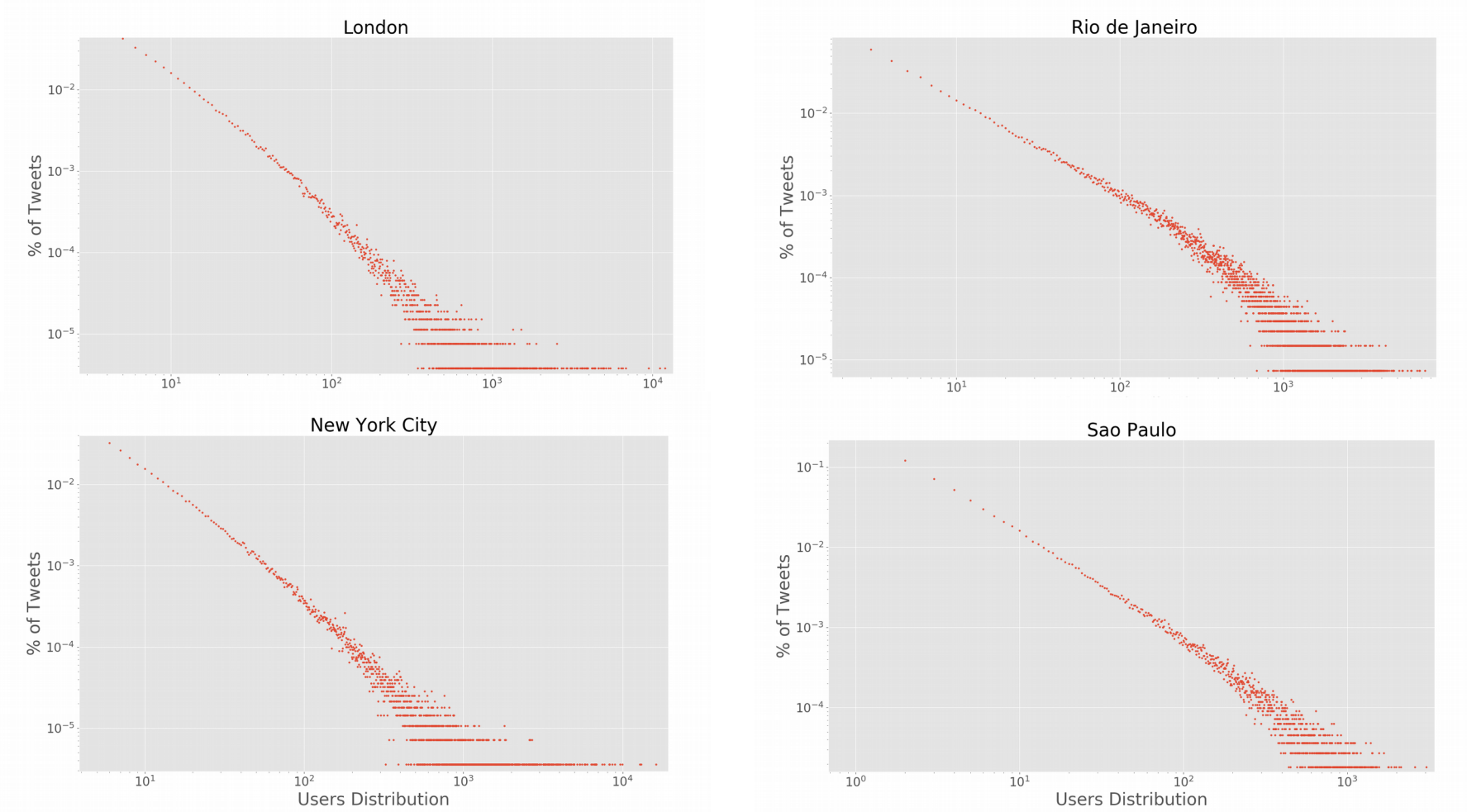



# Users Distribution

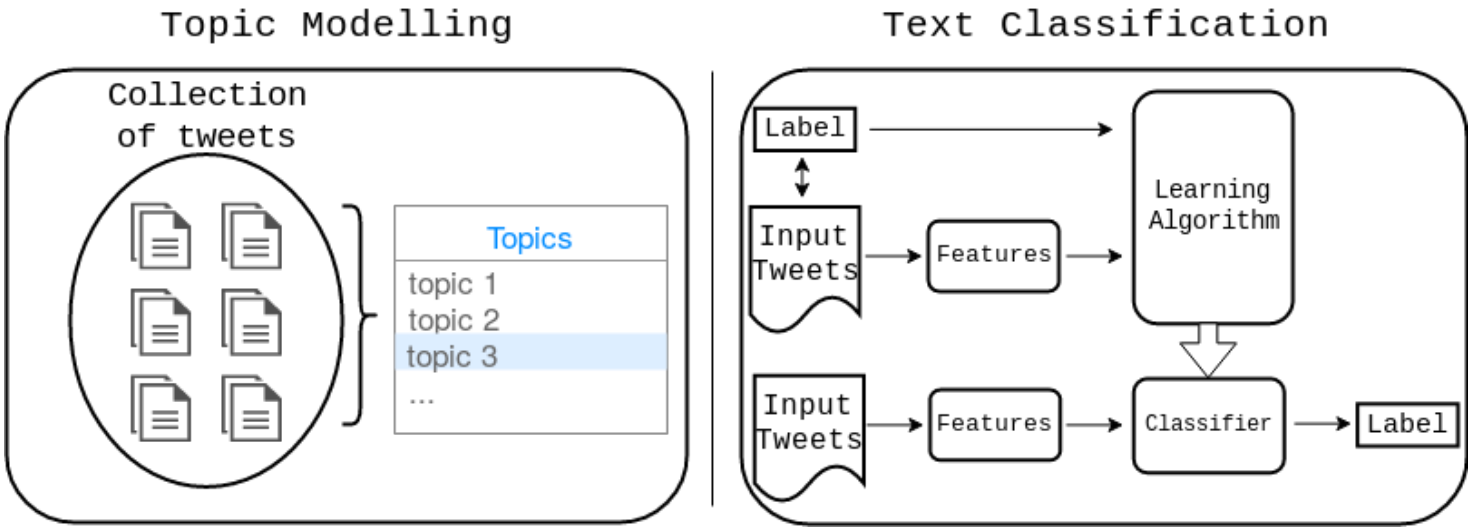
- All cities presented similar user-tweets distributions
- Similar to a power-law



# Text Analytics Experiments

Three experiments to text analysis

- Topic Modelling over Rio de Janeiro and São Paulo;
- Travel-related classification over Rio de Janeiro and São Paulo;
- Travel-related classification over New York City.



# Topic Modelling in Brazil

6.6M geo-located tweets for Rio de Janeiro and 2.7M geo-located tweets for São Paulo

Text pre-processing operations – all previous mentioned

**Features and Model parametrization:**

- Train over 20 iterations (Lansley et al. [4])
- 5, 10, 20, 25, 50 latent topics
- Bag-of-words:
  - Dictionary limited to 10,000 words
  - Words beloging to a maximum of 40% of documents
  - Word minimal frequency - 10

# Topic Modelling in Brazil

High number of overlap terms between topics – 5, 10, 20 and 25.

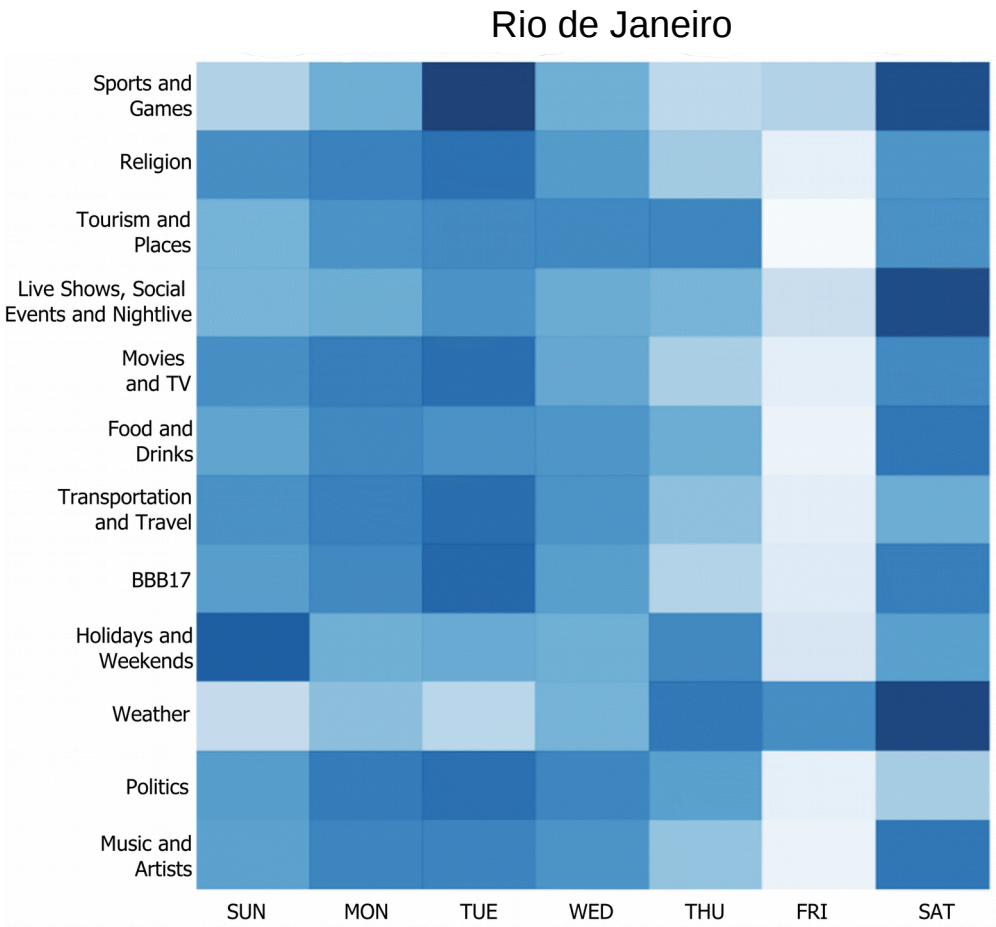
Final experiment model - 50 latent topics

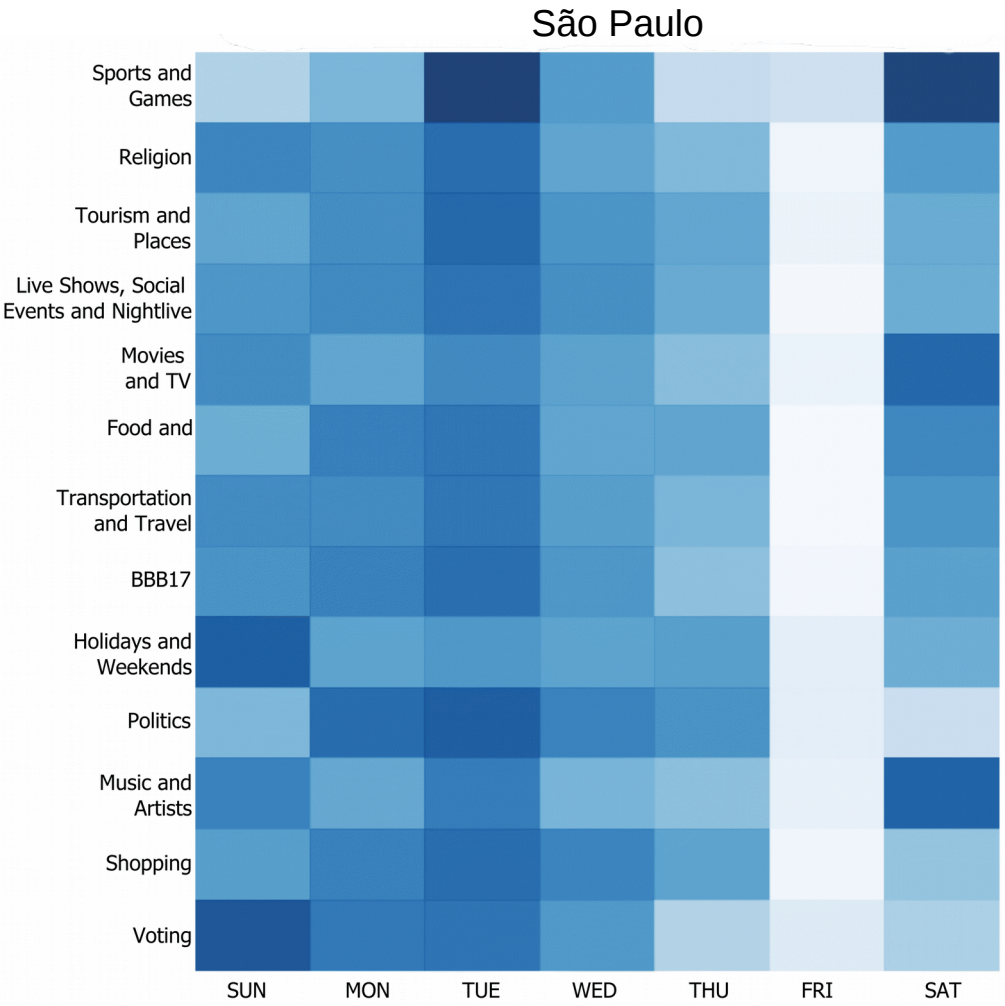
**Results:**

- Topics labelling using a pre-defined taxonomy (Lansley et al. [4])
- Such topics were impossible to label according to the taxonomy
  - For instance, “Academic Activities”, “BBB17”, “Relationships and Friendship”
- Overlapping of words between topics -> Topics Aggregation
  - For instance, “European Football *vs* Brazilian Football”
- 29 different topics, in which 2 are unique in each city

# Topic Modelling in Brazil

## Temporal Distribution of the Latent Topics



# Topic Modelling in Brazil

## Latent Topics Word Cloud

# Travel-related Classification

**Linear SVM**, **Logistic Regression** and **Random Forest**

**Text pre-processing operations:**

- Lower casing
- Transformation of repeated characters
- Cleaning of *metadata* (user mentions and URLs)
- Removal of NLTK stop words

**Evaluation Metrics:**

- Precision, recall and F1-score
- ROC and AUC in Travel-related classification in Brazil

# Travel-related Classification

**Features and Models parametrization:**

- Bag-of-embeddings:
  - Context window of 2;
  - Embeddings of 50, 100 and 200 dimensions
- Bag-of-words:
  - Dictionary limited to 3,000 words
  - Words beloging to a maximum of 60% of documents
  - Word minimal frequency – 10
- Combination of both group of features.

# Travel-related Classification in Brazil

7.7M geo-located tweets, during a period of 1 month, from March 12 to April 12, 2017

**Training and Test datasets:**

- Semi-automatic labeling approach
- Queries using terms of Maghrebi et al. [5]
  - For instance, “carro”, “trem”, “ônibus”
- Balanced training set composed by 2,000 positive and 2,000 negative examples
- Existence of unknown terms in the test dataset
  - For instance, “Busão” and “Uber”
- Test dataset - 71 travel-related tweets and 929 non-related.

# Travel-related Classification in Brazil

**Results:**

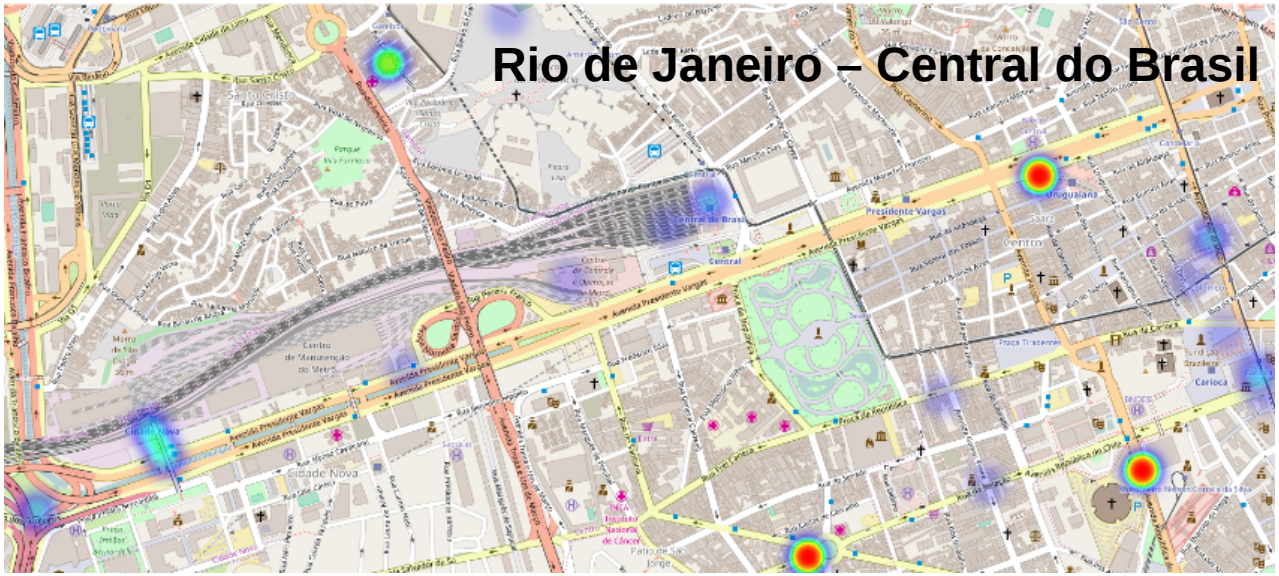
Best model – **Linear SVM**

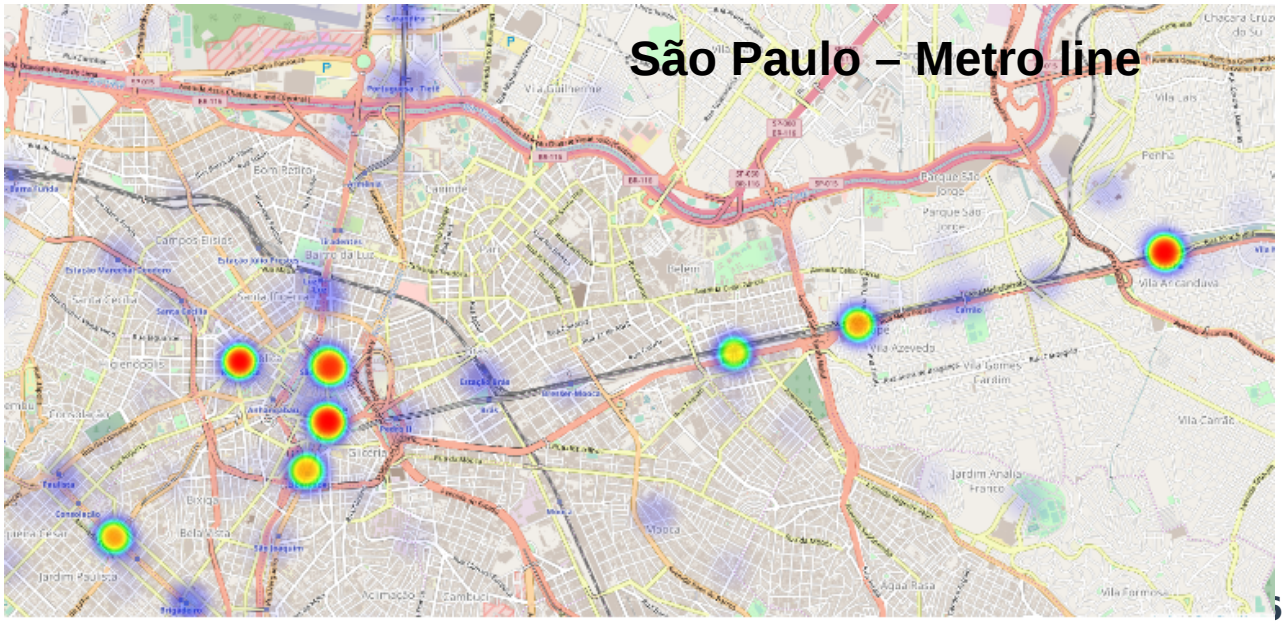
F1-score - **0,8548**

Group of features - **BoW + BoE**

AUC - **0,97**

**37,300 travel-related tweets**



# Travel-related Classification in NYC

4M geo-located tweets in New York City, during a period of 2 months, from March 12 to May 12, 2017

**Training and Test datasets:**

- Two-phase approach - polysemy of English terms
  - For instance, “walk” and “train”
- Balanced training set composed by 1,686 positive and 1,686 negative examples
- Balanced travel-mode classes in the positive examples
- Inclusion of ambiguous geo-located tweets in the negative examples set

# Travel-related Classification in NYC

**Training Strategy:**

k-fold cross-validation - 10 iterations

**Preliminary Results:**

Best Classifier - **Logistic Regression** with **BoE (200) + BoW**

F1-score - **0,98324**

| Classifier | Features | F1-score |
|---|---|---|
| Linear SVM | BoE (200) | 0,87089 |
| | BoW | 0,96962 |
| | **BoE (200) + BoW** | **0,98170** |
| Logistic Regression | BoE (100) | 0,87447 |
| | BoW | 0,97222 |
| | **BoE (200) + BoW** | **0,98324** |
| Random Forests | BoE (100) | 0,82394 |
| | **BoW** | **0,97764** |
| | BoE (50) + BoW | 0,96701 |

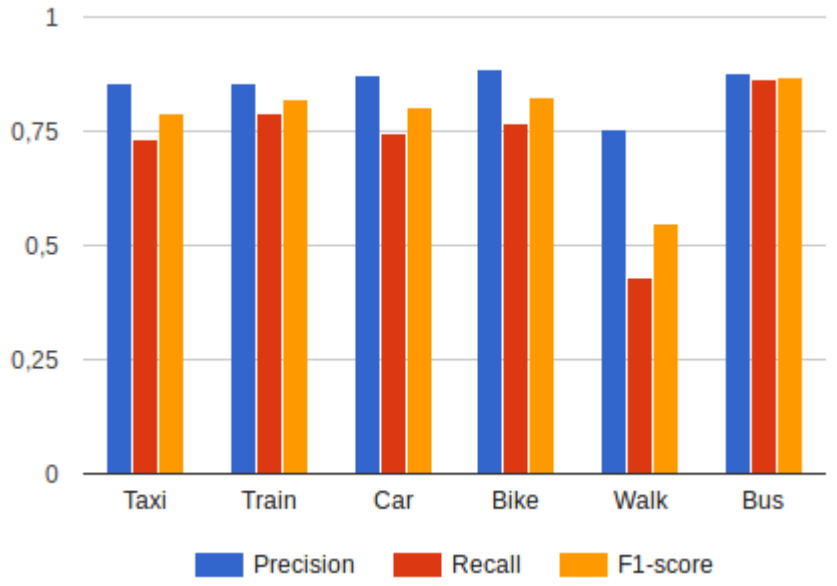
Bag-of-words dependency???

# Travel-related Classification in NYC

***Leave-one-group-out* strategy:**

- Hiding one travel-mode class at a time from the training
- Compare models using BoW and BoE features separately
- 10-fold cross-validation

| Classifier | Features | Mean F1-score |
|---|---|---|
| Random Forests | BoW | 0,12629 |
| | **BoE (50)** | **0,78447** |
| Logistic Regression | BoW | 0,12629 |
| | **BoE (50)** | **0,80219** |
| Linear SVM | BoW | 0,12203 |
| | **BoE (200)** | **0,81289** |

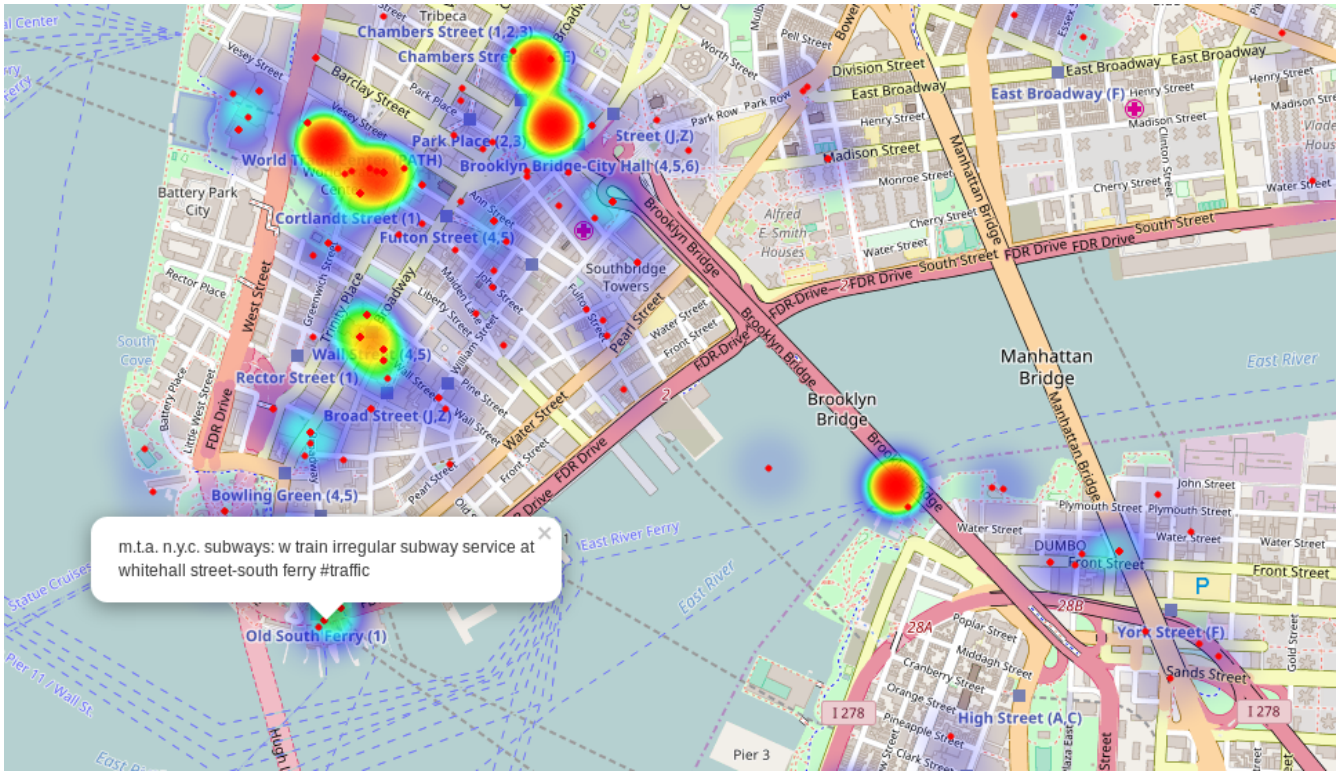



The x-axis are the hidden classes.

# Travel-related Classification in NYC

**Final model**

- BoE with 200 dimensions
- Trained with all travel-mode classes together
- Dataset prediction, 300,000 travel-related tweets




South of Manhattan and over look at the Brooklyn Bridge.

# Conclusions

# Main Results

1. Continuous collection of geo-located tweets from multiple bounding-boxes in parallel and in compliance with Twitter API usage limits
2. Tackling Twitter Streaming API inconsistencies and filtering noisy tweets
3. Implement standard text pre-processing methods for social media texts
4. Content analysis using topic modeling and comparative characterization among different bounding boxes (e.g. cities)
5. Travel-related classification of tweets using supervised learning
6. Train word embeddings from geo-located tweets
7. Study the impact of word embeddings in travel-related classification
8. Creation of gold-standard data for travel-related supervised learning
9. Aggregation and visualization of results

# Contributions

**A) Technical Contributions**

i) Design and implementation of a framework in Python to collect geo-located tweets

ii) The framework allows the monitoring of multiple bounding-boxes (cities and regions)

**B) Applicational Contributions**

i) First large scale study with respect to topic modelling and geo-located tweets in Brazil

ii) Application of word embeddings to text classification tasks in the context of smart cities and ITS

**C) Scientific Contributions**

i) Document the results of experiments and approaches in scientific papers

ii) Sharing of the gold-standard datasets created in this dissertation

# Publications

i. Transportation in Social Media: an automatic classifier for travel-related tweets. In *Portuguese Conference on Artificial Intelligence (EPIA)*, 2017. Published.

ii. Classifying Travel-related Tweets Using Word Embeddings. In *IEEE 20th International Conference on Intelligent Transportation Systems (IEEE ITSC)*, 2017. Under review.

iii. Characterizing Geo-located Tweets in Brazilian Megacities. In *The Third International Smart Cities Conference (ISC2)*, 2017. Under review.

# Future Work

1. Evaluate intrinsically the embeddings produced and share the final benchmarks using the taxonomy of Kuflik et al. [5]

2. Application of a sentiment analysis module to enrich the extracted information

3. Training of embeddings with deep learning models

4. Produce statistics to the unexplored scenarios, London and Melbourne

5. Creation of more complex documents to retrain the model responsible for identification of latent topics

6. Cross the results from travel-related classification with official data of the transportation services

# References

# Social Media Text Processing and Semantic Analysis for Smart Cities

Thank you



Special thanks to *Pedro Saleiro, Rosaldo Rossetti* and *Arian Pasquali.*