

**TOR technology for  
Counterterrorism content**

# IANCIS\*

## Objective:

Developing a tool able to ***crawl Tor websites*** and ***analyze collected data***

### Consortium

Istituto per le Applicazioni del Calcolo "Mauro Picone"

Arma dei Carabinieri HQs-ICT Office

Expert System

### Project Website

[www.iancis.eu](http://www.iancis.eu)



# IANCIS\*

## Objective:

Developing a tool able to ***crawl Tor websites*** and ***analyze collected data***

### Consortium

Istituto per le Applicazioni del Calcolo "Mauro Picone"

Arma dei Carabinieri HQs-ICT Office

Expert System

### Project Website

[www.iancis.eu](http://www.iancis.eu)



## The tool should:

- Automatically **explore** Tor websites
- **Extract text** from collected resources and **analyze it**
- **Visualize analysis results** and identify evidence of illegal activities

# IANCIS\*

## Objective:

Developing a tool able to ***crawl Tor websites*** and ***analyze collected data***

### Consortium

Istituto per le Applicazioni del Calcolo "Mauro Picone"

Arma dei Carabinieri HQs-ICT Office

Expert System

### Project Website

[www.iancis.eu](http://www.iancis.eu)



## The tool should:

- Automatically **explore** Tor websites
- **Extract text** from collected resources and **analyze it**
- **Visualize analysis results** and identify evidence of illegal activities

*The tool is meant to be the first step towards the development of an investigation instrument for the Tor network*

# Tor websites

Tor websites are accessible only through the Tor network and each website is identified by its **onion address**

An onion address is a **non-mnemonic 16-character string** followed by the domain .onion (e.g <http://duskgytldkxiuqc6.onion>)

Tor websites, often called hidden services, let users publish web content **without revealing the location of the site**

# Tor websites

Tor websites are accessible only through the Tor network and each website is identified by its **onion address**

An onion address is a **non-mnemonic 16-character string** followed by the domain .onion (e.g <http://duskgytldkxiuqc6.onion>)

Tor websites, often called hidden services, let users publish web content **without revealing the location of the site**

Several Tor websites **promote illegal activities** such as:

- Terrorism
- Illicit trafficking in narcotic drugs
- Child pornography
- Trafficking in human beings

# Tor websites

Tor websites are accessible only through the Tor network and each website is identified by its **onion address**

An onion address is a **non-mnemonic 16-character string** followed by the domain .onion (e.g <http://duskgytldkxiuqc6.onion>)

Tor websites, often called hidden services, let users publish web content **without revealing the location of the site**

Several Tor websites **promote illegal activities** such as:

- Terrorism
- Illicit trafficking in narcotic drugs
- Child pornography
- Trafficking in human beings



**Silk Road** was an online marketplace, mostly used for selling illegal goods. In 2013, the FBI shut down the website.

# **Data Analysis Framework**

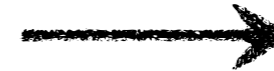


# Data Analysis Framework

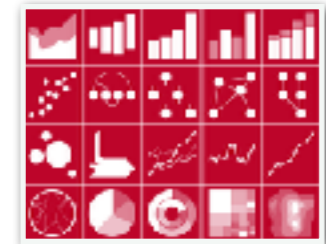
ACQUISITION



ELABORATION

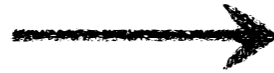


VISUALIZATION

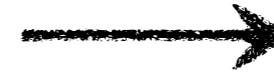


# Data Analysis Framework

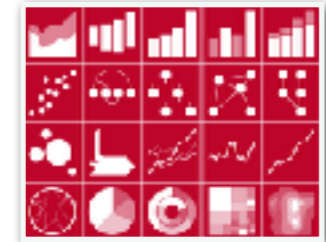
ACQUISITION



ELABORATION



VISUALIZATION



## Acquisition

- Broad crawling
- Focused crawling

# Data Analysis Framework

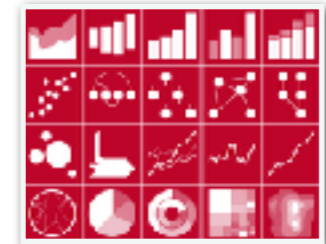
ACQUISITION



ELABORATION



VISUALIZATION



## Acquisition

- Broad crawling
- Focused crawling

## Elaboration

- Data grabbing from web pages
- Text extraction from web resources
- Semantic analysis of extracted texts
- Topic modeling of extracted texts
- Word embedding of extracted texts

# Technologies

## ACQUISITION



**BUBING**: high performance crawler, originally developed by the Laboratory for Web Algorithmics (LAW) of the University of Milan

**SCRAPY**: an open source and collaborative application framework for crawling web sites, which can be used for a wide range of useful applications

# Technologies

## ACQUISITION



**BUBING:** high performance crawler, originally developed by the Laboratory for Web Algorithmics (LAW) of the University of Milan

**SCRAPY:** an open source and collaborative application framework for crawling web sites, which can be used for a wide range of useful applications

## ELABORATION **Text Extraction**



**TIKA:** open-source software suite for the identification and extraction of text from more than 1000 different file types

## ELABORATION **Text Analysis**



**COGITO:** semantical analysis engine by Expert System, which classifies a text according to a suitable taxonomy, providing both quantitative and qualitative information (*what topic* is the text about and *to what extent* the text discusses such topic)

# Technologies

## ACQUISITION



**BUBING**: high performance crawler, originally developed by the Laboratory for Web Algorithmics (LAW) of the University of Milan

**SCRAPY**: an open source and collaborative application framework for crawling web sites, which can be used for a wide range of useful applications

## ELABORATION Text Extraction



**TIKA**: open-source software suite for the identification and extraction of text from more than 1000 different file types

## ELABORATION Text Analysis



**COGITO**: semantical analysis engine by Expert System, which classifies a text according to a suitable taxonomy, providing both quantitative and qualitative information (*what topic* is the text about and *to what extent* the text discusses such topic)

## VISUALIZATION



**R**: a free software environment for statistical computing and graphics

# **Analysis of Unstructured Textual Data**

# I. Semantic Analysis



# Semantic Analysis: Categorization

**COGITO assigns to each document zero or more categories**, representing the **topics** contained in the document (*the number of categories assigned to a document may vary on the basis of the size of the document*)

# Semantic Analysis: Categorization

**COGITO assigns to each document zero or more categories**, representing the **topics** contained in the document (*the number of categories assigned to a document may vary on the basis of the size of the document*)

CATEGORIES ARE **GROUPED BY TAXONOMY**

EACH **TAXONOMY** CONTAINS SEVERAL CATEGORIES

EACH CATEGORY HAS A **NAME**

## Crime Taxonomy

- |Counterfeiting of currency
- |Corruption
- |Fraud
- |Theft
- |Gambling
- |Trafficking in stolen vehicles
- |Rape
- |Racism and xenophobia

## Intelligence Taxonomy

- |Terrorism
- |Logistics
- |Intelligence Agencies
- |Explosive
- |Recruitment, Radicalisation, and Ideology
- |Terrorist Financing
- |Terrorist Propaganda
- |Biological and Chemical Weapons

# Semantic Analysis: Categorization

**COGITO assigns to each document zero or more categories**, representing the **topics** contained in the document (*the number of categories assigned to a document may vary on the basis of the size of the document*)

CATEGORIES ARE **GROUPED BY TAXONOMY**

EACH **TAXONOMY** CONTAINS SEVERAL CATEGORIES

EACH CATEGORY HAS A **NAME**

The **list of categories** has been defined on the basis of IANCIS project's objectives

## Crime Taxonomy

- |Counterfeiting of currency
- |Corruption
- |Fraud
- |Theft
- |Gambling
- |Trafficking in stolen vehicles
- |Rape
- |Racism and xenophobia

## Intelligence Taxonomy

- |Terrorism
- |Logistics
- |Intelligence Agencies
- |Explosive
- |Recruitment, Radicalisation, and Ideology
- |Terrorist Financing
- |Terrorist Propaganda
- |Biological and Chemical Weapons

# Semantic Analysis: Categorization

**A score is associated to each category,** representing the **relevance** of the category in the document

```
<category name="Fraud" score="2430.0" taxonomy="Crime">
```

```
<category name="Cyber Security" score="20410.0" taxonomy="Cyber illegal">
```

# Semantic Analysis: Categorization

**A score is associated to each category,** representing the **relevance** of the category in the document

```
<category name="Fraud" score="2430.0" taxonomy="Crime">
```

```
<category name="Cyber Security" score="20410.0" taxonomy="Cyber illegal">
```

**A list of sentences is associated to each category,** on the basis of their relevance for each given category

```
<sentence start="8946" end="8976">  
  <text>private keys for your bitcoins.</text>
```

```
<sentence start="10763" end="10792">  
  <text>export due to attempted fraud.</text>
```

# Semantic Analysis: Categorization

## OVERVIEW OF CATEGORIES DISTRIBUTION IN THE DATASET

**15.2**

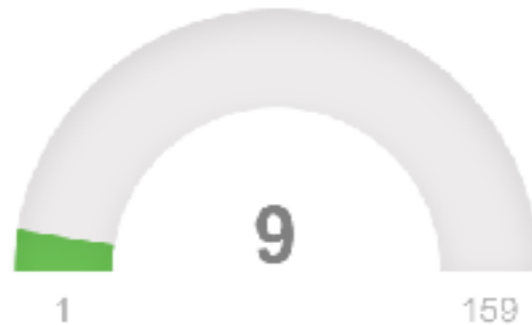
Mean - number of categories per document



Minimum - number of categories per document



Median - number of categories per document



Maximum - number of categories per document



**HTML REPORT REALIZED WITH R**



# Semantical Analysis: Entities Extraction

**COGITO** extracts three different types of entities from documents:

- **PEOPLE**
- **PLACES**
- **ORGANIZATIONS**



# Semantical Analysis: Entities Extraction

**COGITO** extracts three different types of entities from documents:

- **PEOPLE**
- **PLACES**
- **ORGANIZATIONS**

**EACH ENTITY** is specified with the **type of the entity** (PEOPLE, PLACES, ORGANIZATIONS) and a **list of properties** that add information about the entity

```
<entity type="PEOPLE">  
  <text>Tony Abbott</text>  
  <properties>  
    <property source="base" name="sex" value="M" />  
    <property source="base" name="name" value="Tony" />  
    <property source="base" name="surname" value="Abbott" />  
  </properties>  
</entity>
```

```
<entity type="PLACES">  
  <text>Australia</text>  
  <properties>  
    <property source="base" name="georef" value="Oceania" />  
    <property source="base" name="lat" value="S25.0.0" />  
    <property source="base" name="long" value="E135.0.0" />  
  </properties>  
</entity>
```

# Semantical Analysis: Entities Extraction



Each **WORD** represents an entity

**WORD'S COLOR** represents entity's type

**WORD'S SIZE** represents entity's occurrences

# **2. Topic Modeling**

# Topic Modeling

WHAT IS THE DOCUMENT ABOUT?

## Topics

gene 0.04  
dna 0.02  
genetic 0.01  
...

life 0.02  
evolve 0.01  
organism 0.01  
...

brain 0.04  
neuron 0.02  
nerve 0.01  
...

data 0.02  
number 0.02  
computer 0.01  
...

## Documents

### Seeking Life's Bare (Genetic) Necessities

COLD SPRING HARBOR, NEW YORK—  
"How many **genes** does an **organism** need to **survive**?" Last week at the genome meeting here, two genome researchers with radically different approaches presented complementary views of the basic genes needed for **life**. One research team, using **computer analysis** to compare known **genomes**, concluded that today's **organisms** can be sustained with just 250 genes, and that the earliest life forms required a mere 125 **genes**. The other researcher, using **data** from a simple parasite and estimated that for this organism, 600 genes are plenty to do the job—but that anything shorter than 100 wouldn't be enough.

Although the numbers don't match precisely, these **predictions**

are all cut far apart, especially in comparison to the 25,000 **genes** in the human genome, notes V. Andersson, a professor at the University of Sweden, who arrived at the 125-gene figure by coming up with a **computer program** that may be more than just a **simple** **numbers** game, particularly as more and more **genomes** are sequenced and analyzed. "It may be a way of arguing that any newly **sequenced genome** explains why **genes** are important," explains Aravind Muthuraman, an **computational molecular biologist** at the National Center for Biotechnology Information (NCBI) in Bethesda, Maryland. Comparing the



\* Genome Mapping and Sequencing, Cold Spring Harbor, New York, May 9 to 12.

Stripping down. Computer analysis yields an estimate of the minimum modern and ancient genomes.

SCIENCE • VOL. 272 • 14 MAY 1997

## Topic proportions and assignments

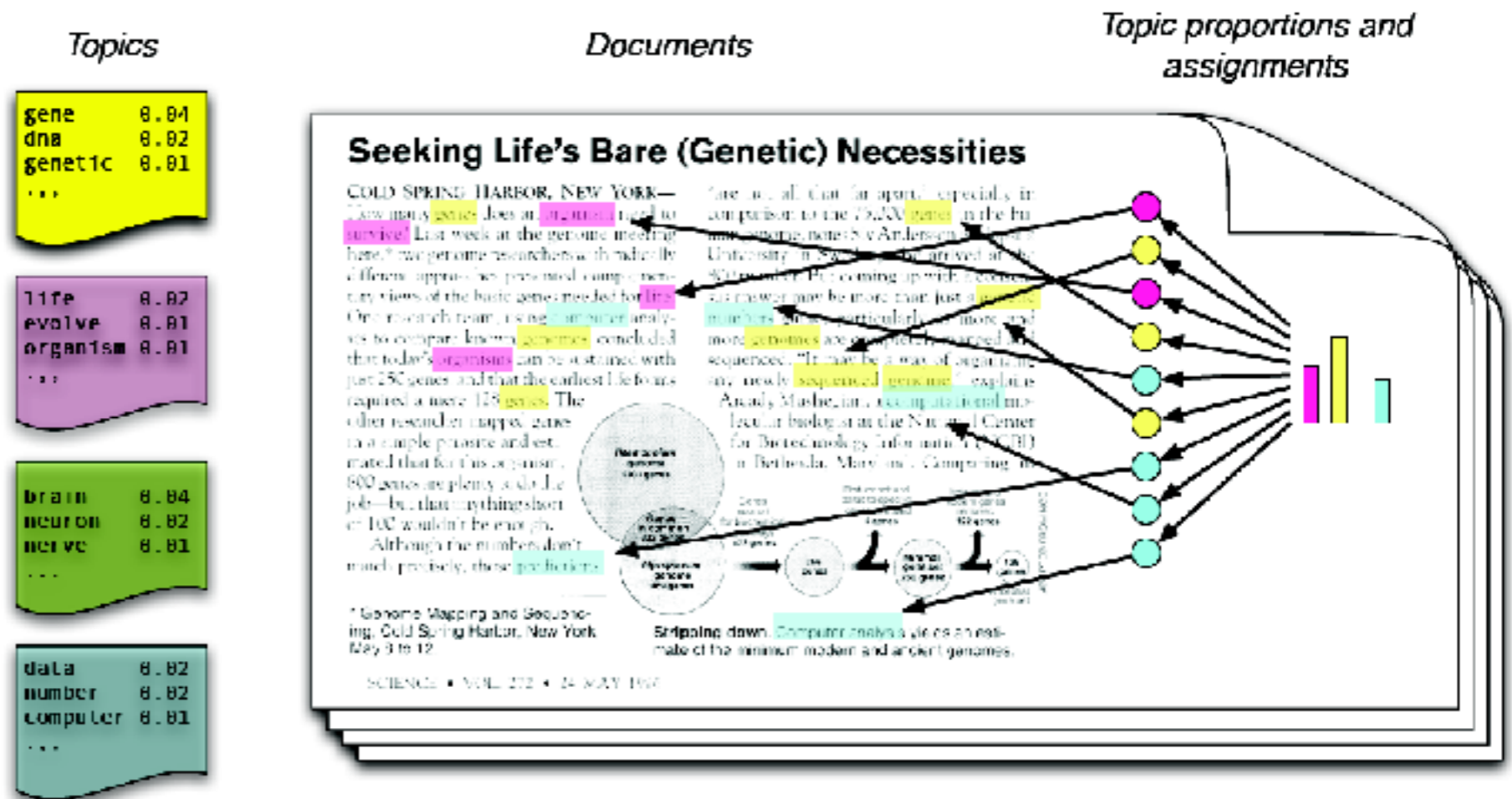


# Topic Modeling

## Topic Modeling:

- A **Topic** consists of a **set of words** that frequently occur together
- **Documents** are generated combining content from **one or more topics**

**WHAT IS THE DOCUMENT ABOUT?**

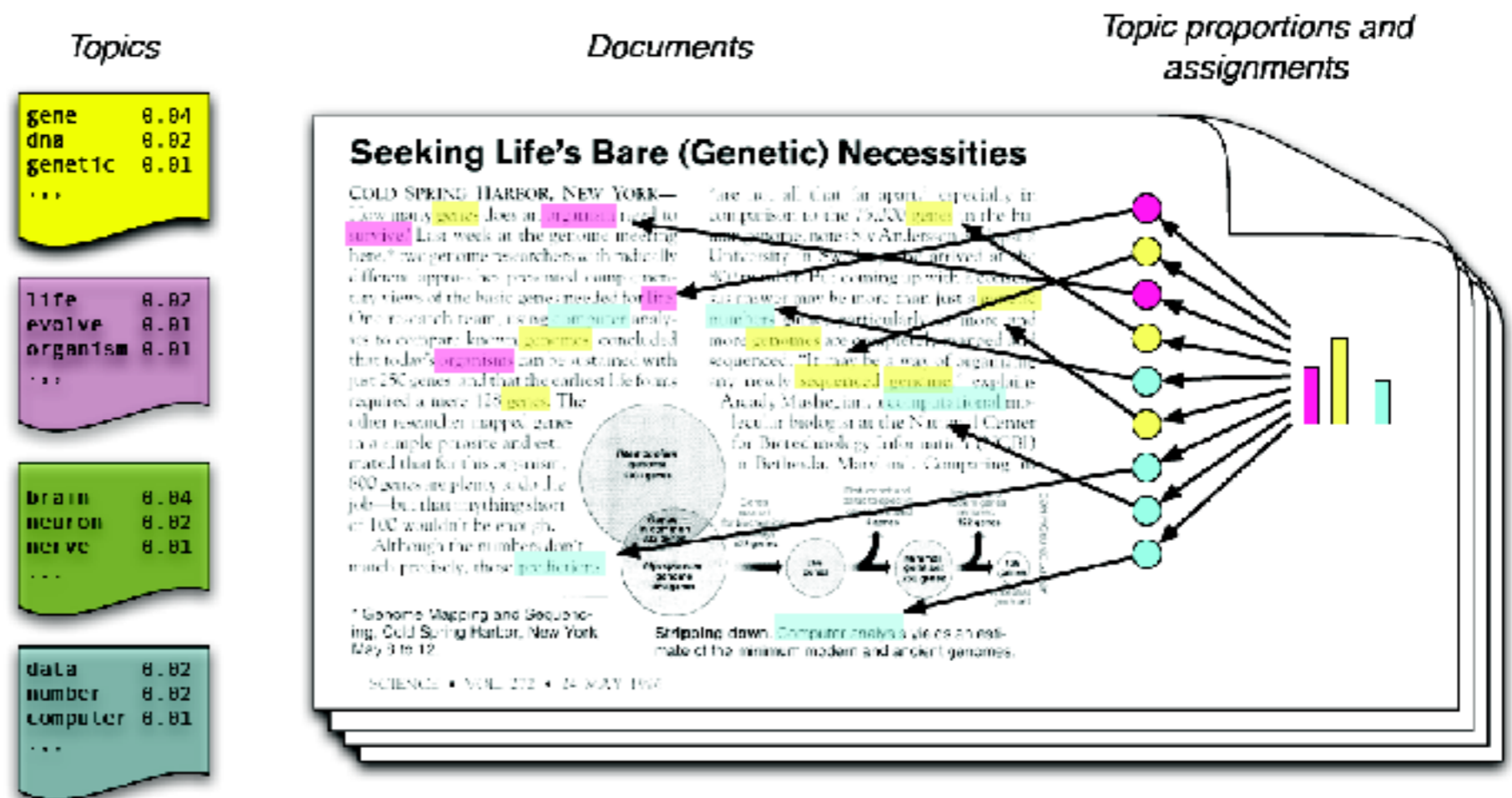


# Topic Modeling

## Topic Modeling:

- A **Topic** consists of a **set of words** that frequently occur together
- **Documents** are generated combining content from **one or more topics**

**WHAT IS THE DOCUMENT ABOUT?**



Given a set of documents, **we can identify** both their **topics** and the **extent to which** each topic is addressed **in each document**

# Topic Modeling

## Topic Modeling VS Categorization

### **CATEGORIZATION:**

*The user defines the categories* to identify in the documents

### **TOPIC MODELING:**

*The system automatically identifies the topics* in the documents

The only required input (besides the set of documents) is  
**the number** of expected topics

# Topic Modeling



Our **dataset** is composed by the **pages of items for sale** in Tor marketplaces

Specifically, we focus our analysis on **illicit drugs trade**



# Topic Modeling



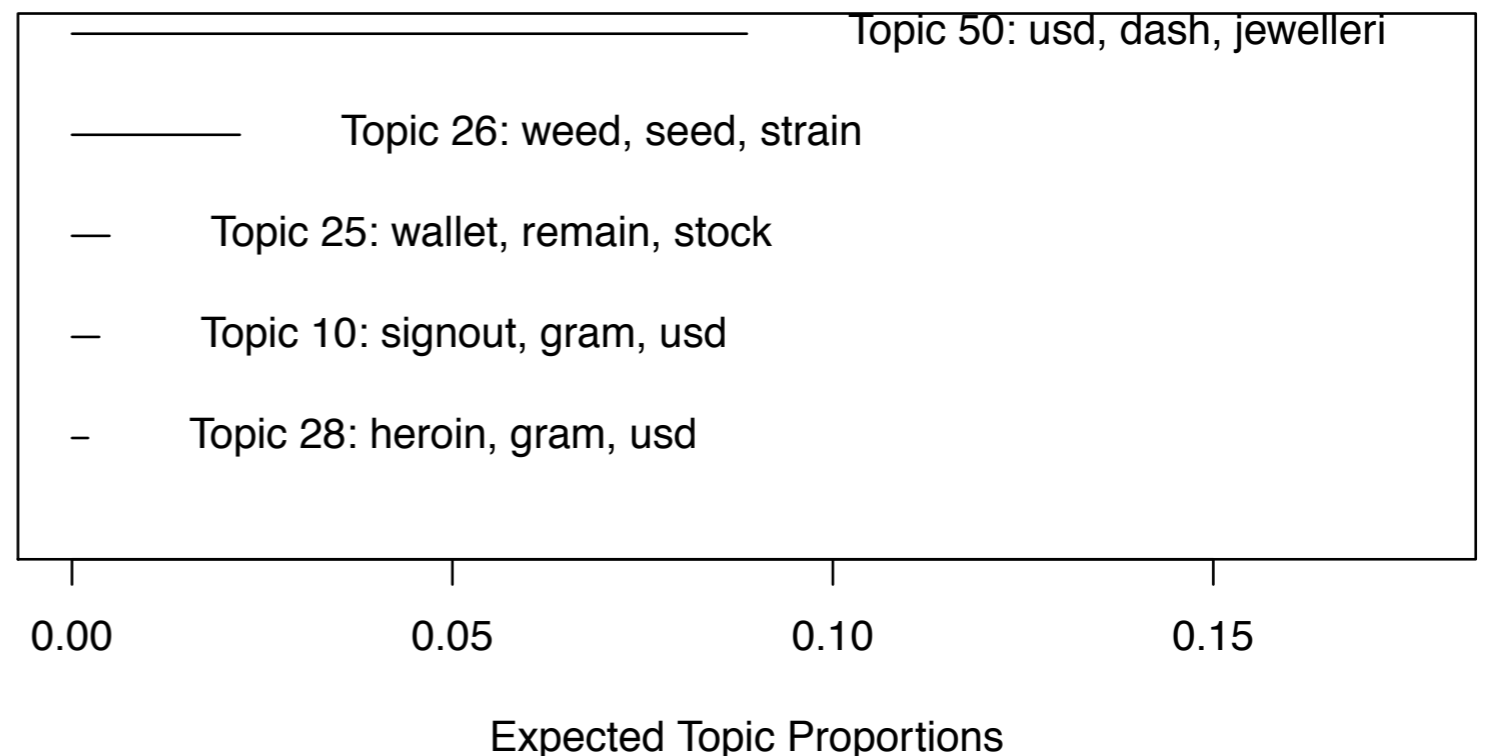
Our **dataset** is composed by the **pages of items for sale** in Tor marketplaces

Specifically, we focus our analysis on **illicit drugs trade**

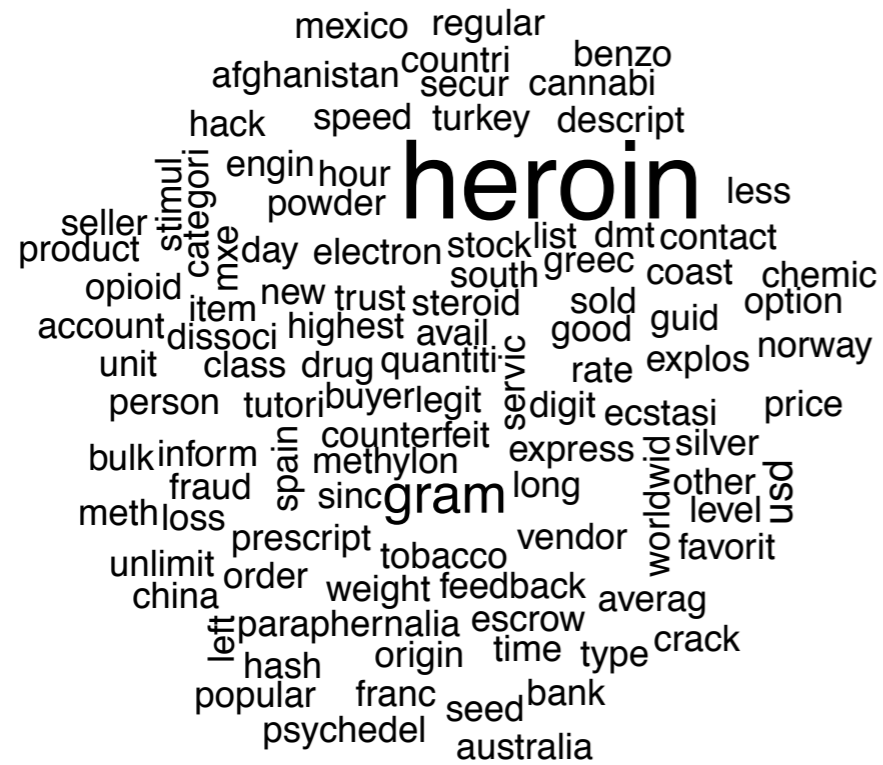
**Which are the topics** in our dataset?

Which are the **most recurrent**?

## Top Topics



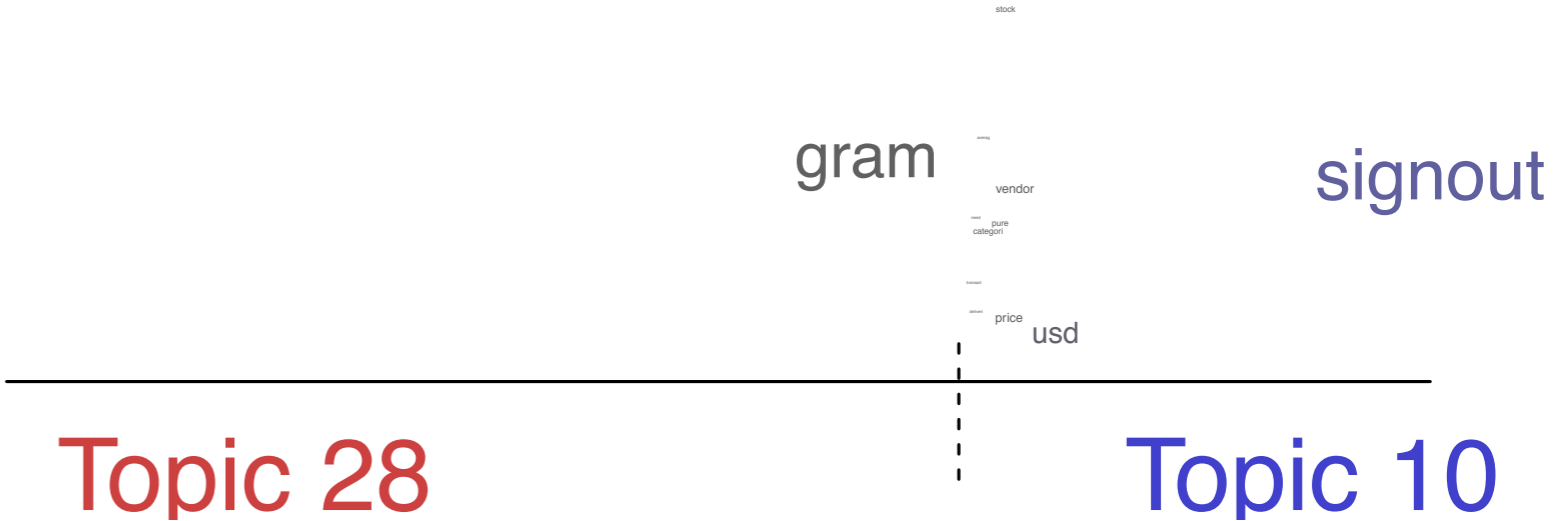
# Topic Modeling



**What is a topic about?**  
Cloud word for *Heroin* topic

heroin

Which words are **specific** for a topic?  
*Heroin* is a **specific word** for topic 28 (not *gram*!)



# Topic Modeling

## Topic 28 – Heroin

W649kFkVrEzJPXn45oHpv4ujqFScsjAopExHjQDK.html

sZG2aJqGREdKowrMH2zGmPMV46C3aOQCOwDYd3F4.html

aTthLjbRF6uBByDooyC6kctaDq8Bpu5ynnyQ7Q2Xw.html

**Most representative  
DOCUMENTS** for each topic

**Most representative  
WORDS** for each topic

Topic 28:  
heroin, gram, usd, price, china, drug

Topic 10:  
signout, gram, usd, price, vendor, stock

# **3. Word Embedding**

# Word Embedding

In **word embedding**, words are *embedded* into a vector space, i.e.  
*each word is represented by a vector*

**WHAT IS THE  
MEANING OF A  
GIVEN WORD?**

**Word embedding** is a state-of-the-art approach to numerically/geometrically represent the meaning of a word

# Word Embedding

In **word embedding**, words are *embedded* into a vector space, i.e. *each word is represented by a vector*

**WHAT IS THE  
MEANING OF A  
GIVEN WORD?**

**Word embedding** is a state-of-the-art approach to numerically/geometrically represent the meaning of a word

**How do we associate a vector to a word?**

There are several options, but **learning algorithms** based on **neural networks** seems to be the most promising one

# Word Embedding

We can use **words vectors** to evaluate **similarity among words**  
*(there might be several different types of similarities between words)*


“Which are the words most similar to **France**?”

France → Italy - Japan - Germany

# Word Embedding

We can use **words vectors** to evaluate **similarity among words**  
(*there might be several different types of similarities between words*)

“Which are the words most similar to **France**?”

France  Italy - Japan - Germany

**Paris - France + Italy = Rome**

Paris - France + Japan = Tokyo

King - Man + Woman = Queen

“What is the word that is similar to **Italy** in the same sense as **Paris** is similar to **France**?”

We can reason on words meaning and answer questions by performing **simple algebraic operations** with the vector representation of words



# Word Embedding

We can use **words vectors** to evaluate **similarity among words**  
(*there might be several different types of similarities between words*)

“Which are the words most similar to **France**?”

France → Italy - Japan - Germany

**Paris - France + Italy = Rome**

Paris - France + Japan = Tokyo

King - Man + Woman = Queen

“What is the word that is similar to **Italy** in the same sense as **Paris** is similar to **France**?”

We can reason on words meaning and answer questions by performing **simple algebraic operations** with the vector representation of words

We can combine words

Czech + currency = koruna

German + capital = Berlin

# Word Embedding

Words meaning is **learned from the context** (i.e. the dataset)

The same word, used in a **different context**, might have a **different meaning**

**Close** the door behind you when you leave.  
Keep your phone **close**, in case he calls!

America is home to many species of **bear**.  
The bridge must **bear** the weight of the cars and trucks.

A hunter's **bow** is often made of flexible wood.  
The violinist takes good care of her **bow**.

# Word Embedding

Words meaning is **learned from the context** (i.e. the dataset)

The same word, used in a **different context**, might have a **different meaning**

**Close** the door behind you when you leave.  
Keep your phone **close**, in case he calls!

America is home to many species of **bear**.  
The bridge must **bear** the weight of the cars and trucks.

A hunter's **bow** is often made of flexible wood.  
The violinist takes good care of her **bow**.

**HOW CAN WE  
USE WORD  
EMBEDDING?**

**Find “codewords”**, e.g. in a marketplace cheese means cocaine

**Disambiguate**, e.g. detect pages in which green means marijuana

**Entity recognition**, e.g. find codename of a terrorist

# Conclusions

- We presented a **general framework for data analysis** composed of three main unities: acquisition, elaboration and visualization.
- We showed that **each unit can be customized** on the basis of objectives
- We discussed different possible approaches for the **elaboration of unstructured textual data** (*semantic analysis, topic modeling and word embedding*)

**Thank you for your time**