The Sentiment Hidden in Italian Texts through the lens of a new dictionary

Giuseppe Bruno, Juri Marcucci, Attilio Mattiocco (**Banca d'Italia**)

Marco Scarnò, Donatella Sforzini (**CINECA**)

Harnessing Big Data & Machine Learning Technologies for Central Banks

Rome, March 26-27, 2018

Sentiment analysis

Discovering emotions, feelings, opinions, etc., about a subject from a written text

General documents

News

Social media:
- Facebook
- Twitter
- Tripadvisor

With the aim to:
- identify the external judgement (point of strengths and weaknesses), of a firm, institution, etc. (to understand *why*, *when*, *what*, etc.)
- find correlations between sentiment and external quantities (also in a predictive framework).
- …

A strategy combined with a new dictionary for Sentiment analysis

Aim:
- Independent from the context
- Usable for the Italian language (and for other languages)

How:
- Selecting and refining one of the most common strategies
  - Lexicon-based;
  - Machine-learning algorithms
- Defining a proper Sentiment index

And then testing the results by considering:
- Internal validation (against other methods of the same *family*)
- External validation
  - Referring to a general problem;
  - By means of a specific economic example

Ingredients of our lexicon-based approach

Dictionary with polarities (+/-) and intensities for a series of terms (in their lemmas)

lemma	polarity	intensity (weigth)
abbacchiamento	Negative	0.7463
abbacchiare	Negative	0.0922
abbacchiarsi	Negative	0.1363
abbacchiato	Negative	0.2408
abbacinante	Positive	0.0924
abbacinare	Negative	0.0456
abbagliante	Positive	0.0958
abbagliare	Negative	0.0156
abbaglio	Negative	0.1499
abbaiare	Negative	0.0069
abbandonare	Negative	0.0399
abbandonarsi	Positive	0.1044
abbandonato	Negative	0.1129
abbandono	Negative	0.1676
abbarbagliare	Negative	0.3731

Formula to evaluate the sentiment index

$$SI = 100 \cdot \frac{\sum W_{PT}}{\sum W_{PT} + \sum W_{NT}}$$

$\sum W_{PT}$: sum of the weights of the positive terms

$\sum W_{NT}$: sum of the weights of the negative ones

A procedure able to analyze the syntactic structure of a text

Considering at least:

- **Negations**
- **Intensifiers** (with weights and able to represent specific syntactic rules)
- **Conditional tenses**

This after having: → Parsing external files → Tokenized terms (sentences) → Part of speech (POS) tagging

An example:

Term	*tuttavia*	*potrebbe*	*non*	*aver*	*eliminato*	*completamente*	*i*	*fattori*	*che*	*rendono*	*problematico*	*usare*	*i*	*VMU*	*quali*	*indicatori*	*dei*	*prezzi*	*del*	*commercio*	*estero*
Lemma	**tuttavia**	**potere**	**non**	**avere**	**eliminare**	**completamente**	**i**	**fattore**	**che**	**rendere**	**problematico**	**usare**	**i**	**VMU**	**quali**	**indicatori**	**del**	**prezzo**	**del**	**commercio**	**estero**
Term (english)	*however*	*(it) could*	*not*	*have*	*deleted*	*completely*	*the*	*factors*	*that*	*make*	*problematico*	*(to) use*	*the*	*VMU*	*as*	*indicators*	*of*	*prices*	*of*	*foreign*	*trade*
Lemma (english)	**however**	**can**	**not**	**have**	**deleted**	**completely**	**the**	**factor**	**that**	**make**	**problematico**	**to use**	**the**	**VMU**	**as**	**indicator**	**of**	**price**	**of**	**foreign**	**trade**
Is negation?	Yes		Yes																		
Is multiplier (or conditional)		Yes (weight 0.8 or 1/0.8)																			
Negative polarity (intensity)					0.102						0.166										
Positive polarity (intensity)		0.192								0.06		0.0189									
Final value	1/0.8*0.192=0.2394		0.102							0.06	0.166	0.0189									

$$\boldsymbol{Sentiment} = \frac{0.102 + 0.06 + 0.0189}{0.2394 + 0.102 + 0.06 + 0.166 + 0.0189} = \mathbf{0.3083}$$

$$\boldsymbol{Sentiment}(\textbf{without rules}) = \frac{0.192 + 0.06 + 0.0189}{0.192 + 0.102 + 0.06 + 0.166 + 0.0189} = \mathbf{0.503}$$

Notes:

- *The negations change the polarity (if not even)*
- *The conditional is considered a multiplier.*
- *The weight of the multiplier depends on the presence of a negation*
- *Negations and multipliers are defined in the procedure with a specific syntax*

Negations: «non», «tuttavia», etc.
term_multiplier «quasi»=0.8=2:1

The syntax for the multiplier considers: <term>=<weight>=<terms before the polar one>:<terms after the polar one>

The dictionary

For the Italian language we started from the "**Open Polarity Enhanced Name Entity Recognition**" (OpenER) dictionary, funded by the European Commission under the 7th Framework Program (FP7). But it was not efficient due to:

- Terms with incoherent polarities and/or intensities (*to think*=>negative with weight 0.25)
- Terms for which the polarity should not be present (like jobs, i.e. *tire repairer*)

=> We built **our own dictionary**, using an approach that extends the work of Kim & Hovy (2004)*, who started from the WordNet thesaurus **adding synonyms and antonyms** to a small set of opinion words collected manually.

Our starting point: the dictionary of synonyms and antonyms

(To abandon)

Term	Synonyms	Antonyms
abbandonare	andarsene,emigrare,desistere,tralasciare,trascurare,allentare, abdicare,cedere,distendersi,rilassarsi,affidarsi,arrendersi,astrarre,deporre,dimenticare,disertare,mollare,piantare,sganciare,smettere,sotterrare,evacuare,accantonare,rinunciare	fermarsi,restare,continuare,accudire,curare,mantenere,reggere,incaponirsi,irrigidirsi,tendersi,lottare,resistere,abbracciare,abitare,aiutare,assistere,concludere,detenere,imbarcare,irrigidire,occupare,proseguire,recuperare,ricostruire,riprendere

For each term we derived an evaluation in a four level scale

Lemma	English	High positive (1)	Medium positive (0.5)	Medium negative (-0.5)	High negative (-1)
aberrante	aberrant				X
abietto	abject				X
abile	clever		X		
abilissimo	very able	X			

The evaluation was made manually, supported by the existent dictionary

*Kim S., Hovy E. Determining the sentiment of opinions. In: Proceedings of international conference on Computational Linguistics (COLING'04); 2004

Evaluating the final polarity and intensity for each weight

The idea is that polarity and intensity allow to ***center*** the term with respect to its **synonyms** and **antonyms**.
This implies the solution of a system of equalities given by:

$$\begin{cases} w_1 = \dfrac{\sum_{i=1}^{M} w_i d_i^1}{\sum_{i=1}^{M} \left| d_i^1 \right|}, (i \neq 1) \\ \dots \\ w_m = \dfrac{\sum_{i=1}^{M} w_i d_i^m}{\sum_{i=1}^{M} \left| d_i^m \right|}, (i \neq m) \\ \dots \\ w_M = \dfrac{\sum_{i=1}^{M} w_i d_i^M}{\sum_{i=1}^{M} \left| d_i^M \right|}, (i \neq M) \end{cases}$$

Where:

- $d_i^m = \begin{cases} -1 \text{ if the term } i \text{ is an } \textbf{antonym} \text{ of } m \\ 0 \text{ if the term } i \text{ is neither a synonym nor an antonym of } m \\ 1 \text{ if the term } i \text{ is a } \textbf{synonim} \text{ of } m \end{cases}$
- $|x|$ represents the absolute value of the term *x*.

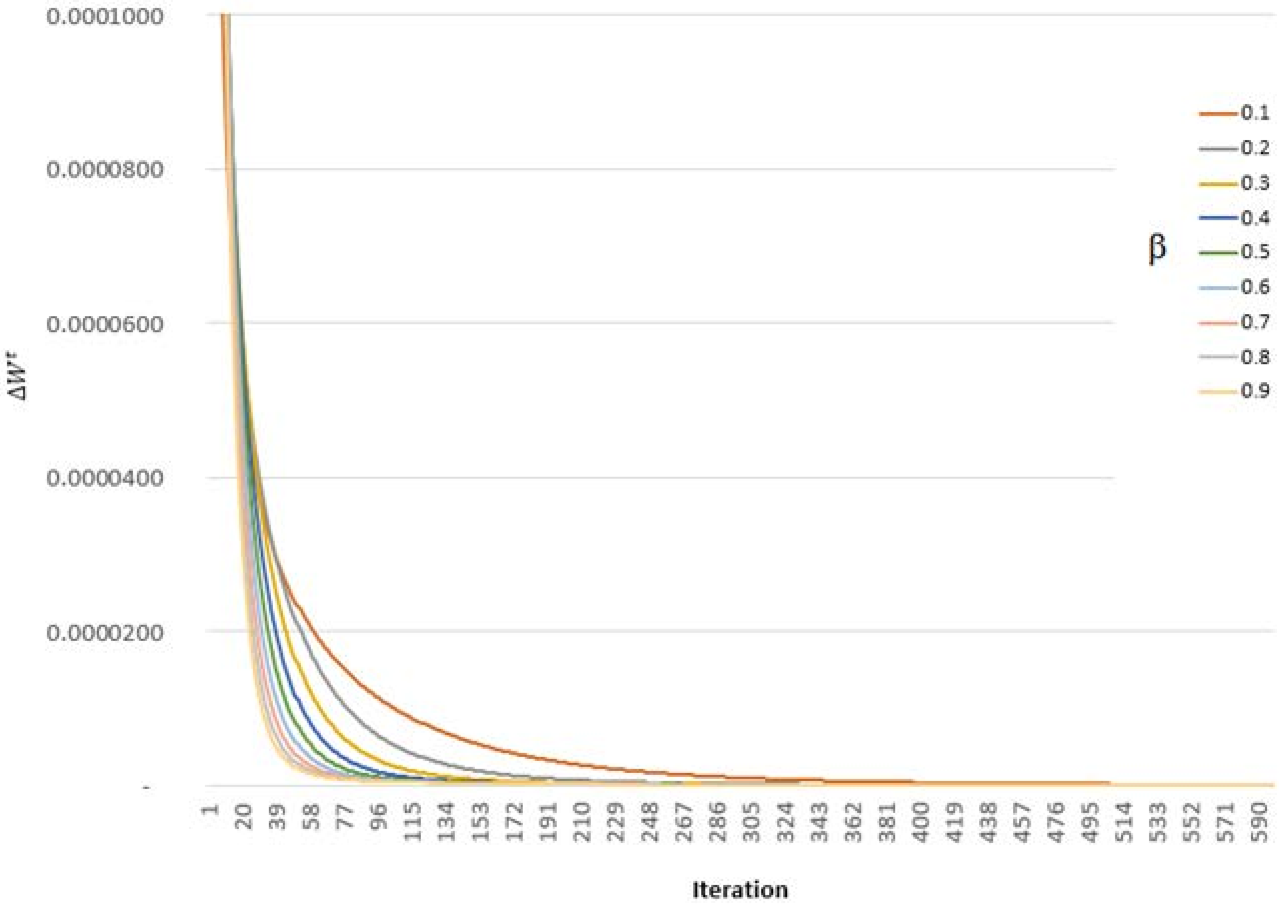
To this purpose we used an **iterative algorithm** that considers:

$$w_m^t == (1-\beta) w_m^{t-1} + \beta \frac{\sum_{i=1}^{M} w_m^{t-1} d_i^m}{\sum_{i=1}^{M} \left| d_i^m \right|}, (i \neq m)$$

Note: the parameter $\beta \in (0,1)$ could be interpreted as a ***learning parameter***, able to smooth the variation of the weights between the iterations

The convergence of the algorithm

With a stopping condition given by: $\Delta W^t = \frac{1}{M}\sqrt{\sum_{m=1}^{M}(w_m^t - w_m^{t-1})^2} < 10^{-9}$, if we consider different β's



We obtained convergence of the algorithm in a few hundreds of iterations.

Internal validation of the strategy

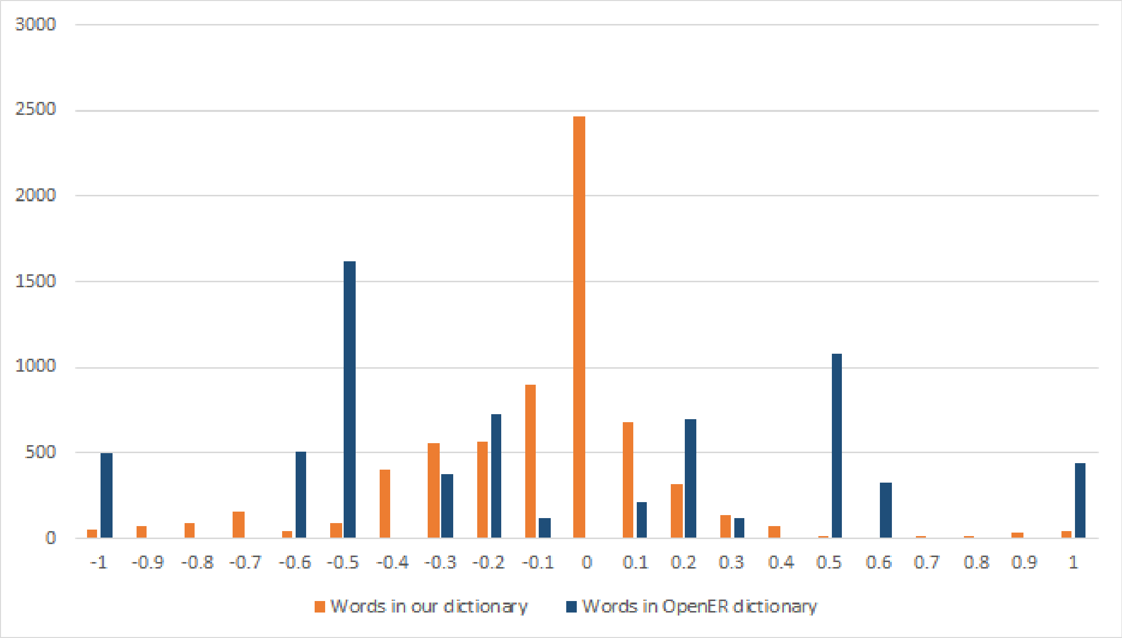
The relation between our dictionary and the OpenER one

		Our dictionary: 18944 terms			
		Negative	Positive	Missing	*Total*
OpenER: 13723 terms	Negative	3608	364	2110	*6082*
	Positive	776	2244	4621	*7641*
	Missing	6677	5275	0	*11952*
	Total	*11061*	*7883*	*6731*	*25675*

$$\textbf{\textit{Concordance Rate}} = 100 \cdot \frac{(3608 + 2244)}{6992} = 83.6$$

Note: we analyzed the discordances and identified inconsistencies in the OpenER dictionary (terms not to be considered or wrong assignment)

Distribution of the intensities (less than 0=negative polarities)



The intensities of the OpenER terms assume few values

External validation of the strategy

*Which dictionary (strategy) can better recognize **pre-labeled sentences**?*

Almost 4,000 Italian sentences (1616 negatives, 2317 positives) from user's comments on products bought from Amazon.

First test: considering positive a sentence when its sentiment is >=50

Without *considering negations and multipliers*

		Our dictionary			OpenER dictionary			
		Negative	Positive	Not assigned	Negative	Positive	Not assigned	Total
Initial classification	Negative	754	852	10	927	676	13	**1616**
	Positive	197	2111	9	426	1865	26	**2317**
	Total	**951**	**2963**	**19**	**1353**	**2541**	**39**	**3933**

Concordance rate=73.2
$\varphi^2 = 0.19$

Concordance rate=71.7
$\varphi^2 = 0.16$

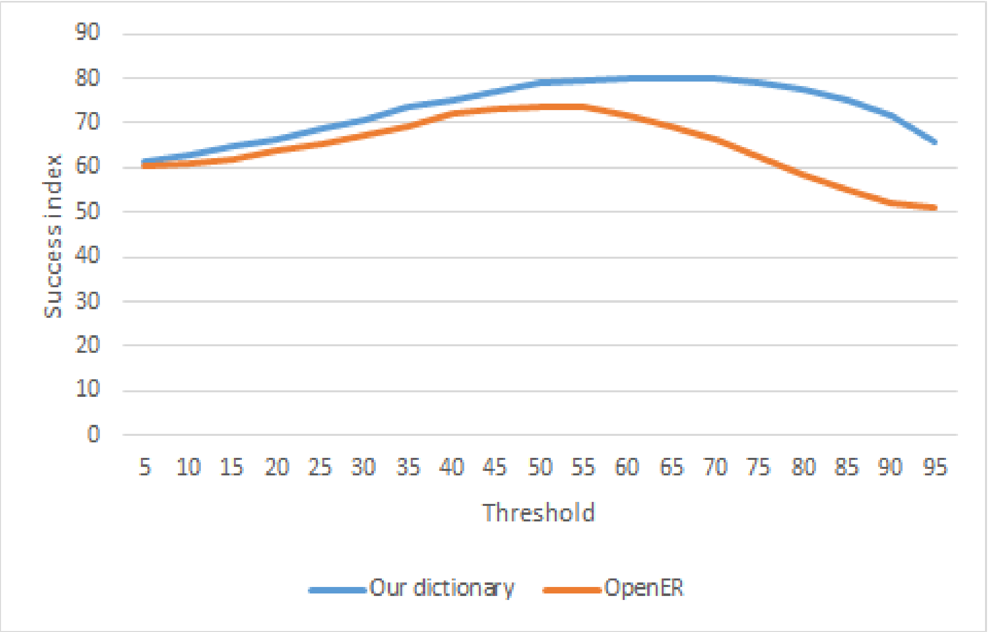
Considering negations and multipliers

		Our dictionary			OpenER dictionary			
		Negative	Positive	Not assigned	Negative	Positive	Not assigned	Total
Initial classification	Negative	966	640	10	1010	593	13	**1616**
	Positive	185	2123	9	427	1864	26	**2317**
	Total	**1151**	**2763**	**19**	**1437**	**2457**	**39**	**3933**

Concordance rate=78.9
$\varphi^2 = 0.32$

Concordance rate=73.8
$\varphi^2 = 0.20$

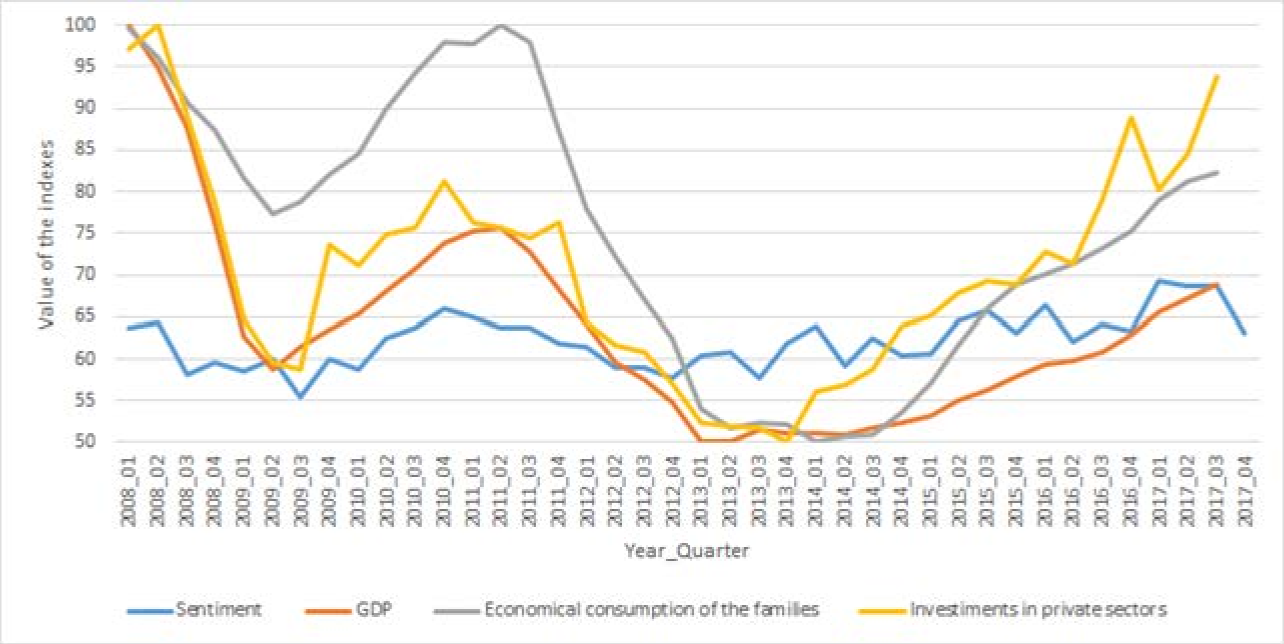
Second test: varying threshold (from sentiment)



External validation of the strategy

Can our dictionary be adequate to evaluate a sentiment that has a sense in respect to a given phenomenon?

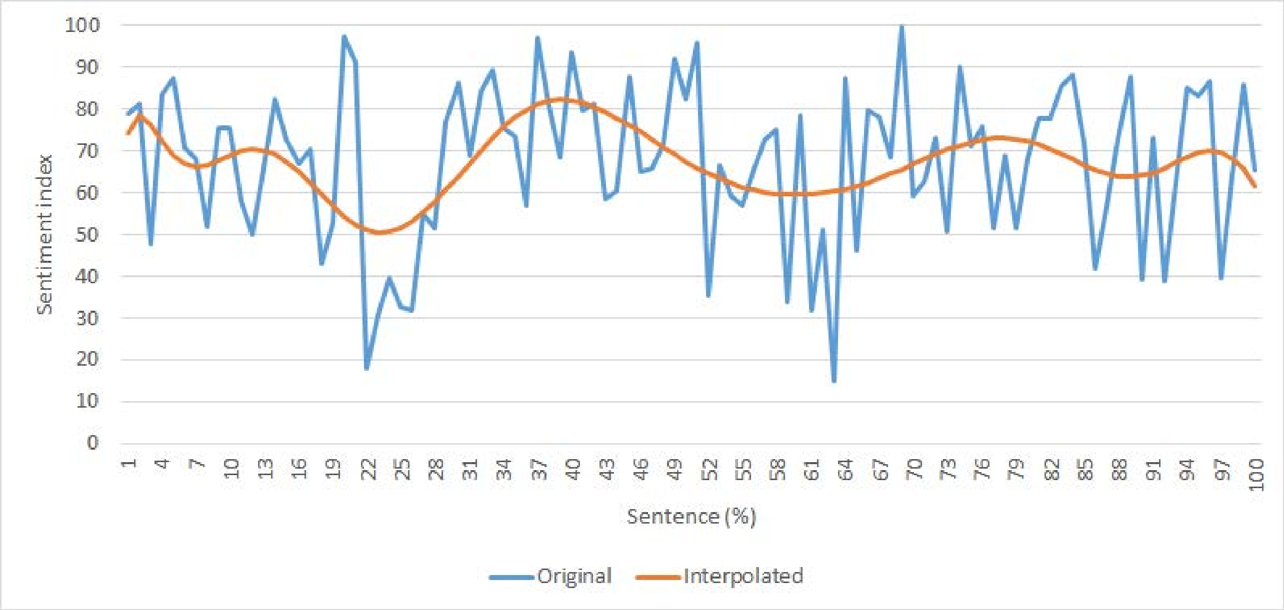
We considered the quarterly **Economic Bulletin** published by the Bank of Italy from 2008 to 2017 (**40 documents**)



The **correlations** between the sentiment index and the quantities considered (that are discussed and interpreted in the bulletins) are evidence of a reasonable capacity of our strategy to catch the “affective states” that are latent in the text.

Sentiment as a *story-telling* indicator

We considered the last quarterly Economic Bulletin published by the Bank of Italy in October 2017



Note: the interpolation considers a polynomial of 14th degree (that corresponds also to the number of sections in each document)

In this case the **Sentiment index** is evaluated **for each sentence**.

The chart shows how in a long text the index could represent the evolution of the ***communication attitudes***

Concluding remarks

We adopted a **lexicon-based approach** to evaluate the **Sentiment index** using:

1. a **procedure** that analyzes the **syntactic structure of a text**
2. a **dictionary** that we built

We verified that:

- **Negations**, **multipliers**, specific types of lemmas **improve** the determination of the **Sentiment**;
- Our **Italian dictionary** gives **coherent results**.

Moreover:

- We evaluated the Sentiment index for the **Economic bulletins** of the Bank of Italy (2008Q1 – 2017Q4), identifying trends and useful relations with other external quantities;
- Specifically, we observed that the **sentiment** associated to each Bulletin is **always positive**, and the lowest values correspond to the Great Financial Crisis and the peak of the sovereign debt crisis;

At this point we have a **full automatic strategy** that, starting from a series of texts (files), can be used to derive the sentiment index associated to each of these (or to every sentence).

Next steps:

- Publication of an **extended article**, in which it will be shown also a method to characterize topics;
- **Distribute** the procedure and the Italian dictionary;
- **Extension** of the strategy to Twitter feeds and to news articles from newspapers

Thank you very much!

Questions?