Visualization of Big Data Text Analytics in Financial Industry: A Case Study of Topic Extraction for Italian Banks

Živko Krstić
Atomic Intelligence, Data Science Department, Zagreb, Croatia
Sanja Seljan
Faculty of Humanities and Social Sciences, Information and Communication Sciences, Zagreb, Croatia
Jovana Zoroja
Faculty of Economics and Business, Zagreb, Croatia

Abstract

Textual data and analysis can derive new insights and bring valuable business insights. These insights can be further leveraged by making better future business decisions. Sources that are used for text analysis in financial industry vary from internal word documents, email to external sources like social media, websites or open data. The system described in this paper will utilize data from social media (Twitter) and tweets related to Italian banks, in Italian. This system is based on open source tools (R language) and topic extraction model was created to gather valuable information. This paper describes methods used for data ingestion, modelling, visualizations of results and insights.

Keywords: visualization, data science, FinTech, topic modelling, LDA
JEL classification: C55, C8

Introduction

Text analysis and mining is process where analyst interacts with collection of documents using set of analytical tools with goal of extracting useful information that can be used for making better business decisions (Feldman et al., 2007). Foundation for any text analysis is corpus or document collection.

Various data mining approaches have been used in different areas. Pejic Bach and her colleagues investigate usage of text mining in finance where they focus on text mining for big data analysis in financial sector (Pejić Bach et al., 2019a, 2019b), from pharmaceutical documentation (Seljan, et al., 2017), in medical domain to better understand users' needs (Seljan et al., 2014), in legal domain (Gašpar et al., 2016), etc. In order to forecast macroeconomic indicators, Elshendy (2017) investigate big data analysis of economic news. Furner et al. (2017) examine the Role of Mobile Self-Efficacy regarding mobile products reviews. Text mining can be used in order to analyse and extract textual data aiming to get better bibliometric review for Twitter usage in Tourism (Ćurlin et al., 2019) and for ten-year publishing of INDECS journal (Stepanić et al., 2017).

In this paper topic extraction technique will be presented based on documents collected from Twitter throughout standard API. Topic modelling is a process of identifying different concepts embedded in corpus of documents (Uys et al., 2008). Topics generated by modelling process are made up by significant words and they

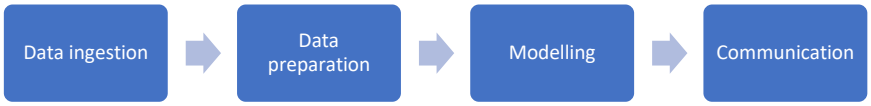
provide summary of documents (tweets) in a specific time frame. With these techniques we can see similarities and differences between different banks and trends in topics generated over time.

Results from topic modelling will be presented with simple data application built for the purpose of this paper. This data application will contain visual data created as a results of text analysis performed on big data. All analysis and data application, including visualizations, are built using programming language R. Data science packages built in R will be also presented for each section in this paper.

Methodology

Methodology used for work in this paper follows standard data science workflow (Wickham et al., 2016) and it is comprised from these steps:

Figure 1
Data science methodology



Source: Wickham et al., 2016

Data ingestion is a first step and it is the process of data collection and import into the system, either for storage or database building, and for further data analysis. Source used for work and data ingestion is Twitter. Twitter data are available from public API and open source package in programming language R. Data collected for this paper will be stored locally for further analysis in optimized format. The next step is data preparation which involves steps of data transformation and cleaning, with the end goal to create „tidy" dataset (Wickham, 2014). The concept of „tidy" data is widely used in R community and it involves these principles:

- Column represents one variable
- Row represents one observation
- Table represents each type of observational unit

These principles determine whether one dataset is considered as „tidy". Opposite of this would be „messy" data. Techniques to handle textual data in „tidy" format are also available for R community (Silge et al., 2016). „Tidy" text format contains one token per document and per row. Modelling is a process where we take „tidy" dataset as input and provide simple low-dimensional representation of data as output. Statistical analysis or machine learning can be used to gain new insights in modelling section. This process helps us to better understand our datasets and provide us new insights that can be used for making better business decisions. The final part of data science methodology is communication. Better preparation for these final steps is important for success of every data science project. This is the part where all results and prior work need to be presented in simple and insightful way to the consumer.

Communication involves the process of presenting results and insights from analysis to all interested parties. Key part is the representation in simple and informative way, using proper visualization types and using various communication formats to accommodate different types of demands for insights. Common communication formats in data science with R are markdown or bookdown formats and data applications. Other formats are graphical format and workflow format called notebooks. Markdown or bookdown are tools for integration of code, textual

explanations and results. Data applications are the most complex type and they can be built in R using only R syntax, while R generates either html, CSS or JavaScript needed to create data application (Beeley, 2013). In this paper, data application will be presented for result communication.

Data Ingestion

Data are foundation for any data science work. Textual data can be collected from many data sources and be utilized later with text analytics. We will concentrate only on one source, specifically from social media, Twitter. Reason for this is abundance of textual data publicly and easily available through Twitter API. Twitter API had several changes over time, mostly regarding restricting access to some data to keep privacy of users. Now, all developers need to complete developer account application in order to create and maintain apps.

Regarding all features that Twitter API is providing, we will use one in particular, Search Tweets feature. In general, Twitter has several versions of API: Standard, Premium and Enterprise (web resource – Twitter, N/A). Most important difference is data availability and query capability. With standard API (free version) we can access last 7 days of tweets and we can use standard operators for data collection, while in other versions we can extract full archive of tweets from year 2006 and premium operators are available. It is also important to state that in standard search API we don't have full access to all data since it is focused on relevance and not on completeness (Bruns et al., 2012). To over surpass this obstacle, premium and enterprise APIs can be used. To properly collect textual data from Twitter we need to specify correct parameters. These parameters need to filter only tweets of interest. Parameters in standard API that can be used are:

- Search query (important and required) – We are extracting only tweets that contain certain string and it can be maximum 500 characters long. Support of operators is also available.
- Geocode – With this parameter we can filter only users from specific location radius.
- Language – Filtering of specific language (Italian in our case)
- Result type – Has 3 states: mixed (returns popular and real time results), recent (only real time results) and popular (only popular results). Default is mixed.
- Count – How many tweets to retrieve per page. Maximum number is 100.

These are the most important parameters that can be used, but others are also available. For the purpose of this paper, search query parameter is used (which is also required) language parameter and count. Search query in our case study is the name of the bank. For example: Bank name1, Bank name 2, Bank name 3. For purpose of this paper and its simplicity we will concentrate on 3 banks. Language is filtered to only Italian and Count is set to maximum. Other settings are set to default.

Methodology used was to retrieve 2000 unique tweets for each of the three banks, which makes 6000 tweets in total, in one specific month. Since we can't collect 6000 tweets with just one API request, this was iterated until final number of tweets was collected. Limits in standard search API are 180 requests in 15 minutes window for user authentication and 450 requests in 15 minutes window for app authentication. These limits can be overpassed with other API types like premium. Another way to overcome this is to use Streaming API which is more suitable for live systems that will track real-time activity on Twitter for specific keywords/users. Limits for standard API are: 400 keywords, 5,000 user IDs and 25 location boxes.

Features available after data collection are: text, favourite count, retweet count, screen name, language, profile image URL, location, longitude, latitude, created

time, user data group (id, statuses count, time zone and other), entity group (hashtags, user mentions, symbols, URLs) and other. Some data availability can vary depending on user privacy settings. Since we are interested in textual data, several most important features are used. These features are presented below in Table 1.

Table 1
Financial Features List

Column name	Original column	Column type	Example
Tweet	Text	character	Questa banca è la migliore!!!
Account	Name	character	User1950
Date	Created at	timestamp	2019-02-24 18:55:06
Number of favourites	Favourite count	numeric (integer)	5
Number of retweets	Retweet count	numeric (integer)	1
Search keyword		character	Bank name

Source: Authors' work

Data Preparation

Data preparation step is the process of cleaning and preparing dataset for the purpose of analysis and machine learning. In our case. we need to prepare textual data for topic modelling or extraction. The first step in this preparation was done initially, upon data ingestion when initial source dataset was filtered to only 5 features and JSON results from API search query where converted to „tidy" dataset also called in R „tibble". After obtaining „tidy" dataset as input, several preparation steps needed for modelling follow:

- Keyword extraction
- Lowercase conversion
- URL filtering
- Punctuation and number filtering
- Stop words filtering
- Whitespace filtering
- Nchar filtering
- Document-term matrix creation
- Frequency calculation

Simple keyword extraction is the first step in data preparation. Since one tweet can have several banks mentioned, new column is created where all keywords mentioned in specific tweet will be placed. This is created with simple string search by pattern available in „stringr" package in R. These results are compared with column „Search keyword" for testing purposes. „Search keyword" column was created in process of data ingestion to state which search query parameter was used to extract that specific tweet. This step is crucial for final part of communication and visualization since we can use this information to properly present difference between banks.

Next are standard text analysis preparation steps which include converting textual data to all lower cases to avoid making distinction between same words but written with first upper letter. URLs are filtered with regular expressions since this information is not needed for our case study. Punctuation filtering involves removing dots, commas and other characters that are not needed for further analysis. Number filtering and punctuation filtering are performed with standard functions in R with specific package for text analysis and mining (tm).

Stop words filtering is the process where common words are removed from text since we can save space and modelling time (Vijayarani, 2015). Stop words list is

available in many packages in R but the standard „tm“ package was used to filter Italian stop words. Supported languages are Danish, Dutch, English, Finnish, French, German, Hungarian, Italian and other. To remove leading and trailing whitespace from text, „tm“ function is used to perform this task. With Nchar filter, we are removing all tokens or words, which in our case contain less than 3 characters in order to save some space and lower computation speed for modelling. All these steps are performed on original tweet column. The new column was created, called „Clean tweet“, so that we can easily see difference between the original tweet and the new cleaned tweet.

Document-term matrix needs to be created in order to perform topic modelling since this is input for it and not raw text. To create document-term matrix we can use the same „tm“ package which has function that transforms raw text to document-term matrix. This matrix describes the frequency of each term occurring in collection of documents/texts. For frequency calculation TF-IDF calculation was used (Ramos, 2003). Each value in document-term matrix will be TF-IDF frequency for that token and document. Term frequency measures how frequently specific term occurs in a document. To calculate this, we need to divide number of time specific term appearing in a document with total number of terms in the document. IDF or inverse document frequency calculates term importance. This is calculated by dividing total number of documents with number of documents with specific term in it, and then using a log value of this result. To get final TF-IDF calculation we need to multiply TF with IDF.

Modelling

Topic modelling is a part of unsupervised classification, similar to clustering. Topic modelling or extraction has goal of extracting groups or topics from textual data that explain why some tweets are similar. Input for topic modelling is our „tidy“ dataset prepared in previous steps. The column „Clean tweet“ is used as an input for algorithm. Then this column was converted to document-term matrix which is format that topic extraction algorithm can use for computation. Algorithm used in this paper is LDA or Latent Dirichlet allocation (Hong et al., 2010). LDA algorithm is used to group similar textual data or documents and get patterns from these documents.

LDA treats every text/document as mixture of topics and each of these topics is mixture of different tokens or words. This overlapping is something that is common in natural language. Topics generated with LDA algorithm are used as key point in communication step, since they can help us present main topics by date. Pre-processing steps are important for better quality analysis of these models as in any text analysis task.

LDA algorithm has document-term matrix as input, created in data processing step. This algorithm has two inputs, where first one is document-term matrix and second is number of topics that we want to generate. There are certain methods that can be used to find optimal number of topics but for the purpose of this paper fixed number is used, which is equal to 4. After LDA topic generation steps we can extract results in „tidy“ way using „tidytext“ package where final output is table presented below in Table 2.

Table 2
Top Terms for Each Topic Based on Beta

Topic	Term	Beta
1	grazie	1.00e-111
2	servizio	3.28e- 6
3	banca	4.85e-14
...	...	...

Source: Authors' work

The last column Beta gives probability of term for that topic. We can also extract information about probability and "winning" topic for each document which we can see below in table 3.

Table 3
"Winning" Topic for Each Document (Tweet)

Document (Tweet)	Topic	Gamma
Docid1950	1	0.00005
Docid1911	2	0.08745
Docid2019	1	0.000009
...	...	...

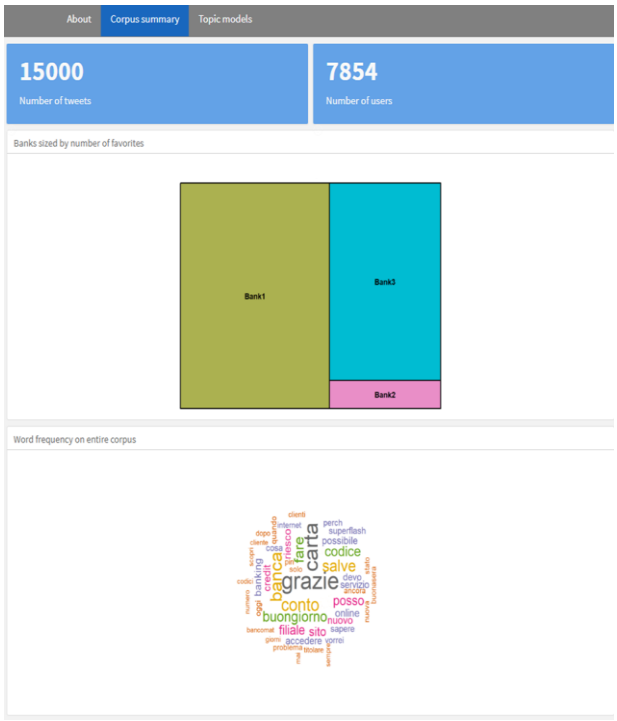
Source: Authors' work

Communication

The final part is to properly communicate results gathered from LDA algorithm. In this chapter data application as a way of communication will be presented. Graphical representations will describe information and new insights gathered in this text analysis. This application is created with package called „shiny" and this package is used for data application development in R (RStudio, N/A).

The first tab is about summary of our collected dataset. This tab contains information about number of collected and number of users who tweeted. We can also see interesting comparison between 3 banks based on measure „Number of favourites". Also document-term matrix is used to visualize word cloud of most frequent words for entire corpus (Figure 2.).

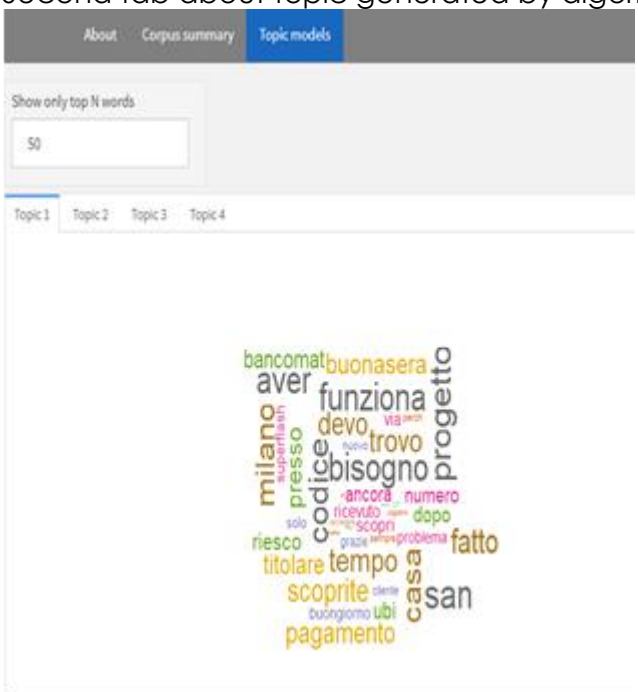
Figure 2
First Tab in Application Presenting Summary of Dataset



Source: Authors' illustration

The second tab is about topic comparison (Figure 3.). Here we can see top 50 words/terms for each topic which are visualized by word cloud. We have also input filter which is interactive and users can change preferred number of words in word cloud.

Figure 3
Second tab about topic generated by algorithm



Source: Authors' illustration

Conclusion

Methodology and techniques used in this paper can be utilized in similar case studies, not just with Twitter data, but also with any textual data. Methods presented in the paper are not specific only for the financial industry, but can be used for any other domain to extract information. Insights created with topic modelling can be used to improve business decisions and to better understand customers. These analyses can be further expanded with other models like sentiment analysis models, named entity extraction models, churn prediction models and others. Textual data and these models can be used to improve already available models in financial organizations, like credit risk models. In the paper open source technologies are presented, capable of solving various data science tasks which could be used to generate new insights.

References

1. Beeley, C. (2013), Web application development with R using Shiny, Packt Publishing Ltd.
2. Bruns, A., Liang, Y. E. (2012), "Tools and methods for capturing Twitter data during natural disasters", First Monday, Vol. 17, No. 4, pp. 1-8.
3. Ćurlin, T., Jaković, B., Miloloža, I. (2019), "Twitter usage in Tourism: Literature Review", Business Systems Research, Vol. 10, No. 1, pp. 102-119.
4. Elshendy, M., Fronzetti Colladon, A. (2017), "Big data analysis of economic news: Hints to forecast macroeconomic indicators", International Journal of Engineering Business Management, Vol. 9.
5. Feldman, R., Sanger, J. (2007), The text mining handbook: advanced approaches in analyzing unstructured data, Cambridge University Press.
6. Furner, C. P., Zinko, R., Zhu, Z. (2017), "Examining the Role of Mobile Self-Efficacy in the Word-of-Mouth/Mobile Product Reviews Relationship", International Journal of E-Services and Mobile Applications, Vol.10, No. 4, pp. 40-60.
7. Gašpar, A., Seljan, S. (2016), "Consistency of Translated Terminology Measured by the Herfindahl-Hirshman Index (HHI)", Lecture Notes in Computer Science (LNCS), Springer.
8. Hong, L., Davison, B. D. (2010), "Empirical study of topic modeling in twitter", in the Proceedings of the 1st Workshop on Social Media Analytics, Washington D.C., District of Columbia, USA, ACM, pp. 80-88.
9. Pejić Bach, M., Krstić, Ž., Seljan, S., Turulja, L. (2019a), "Text Mining for Big Data Analysis in Financial Sector: A Literature Review", Sustainability, Vol. 11, No. 5.
10. Pejić Bach, M., Krstić, Ž., Seljan, S. (2019b), "Big data text mining in the financial sector", in Metawa, N., Elhoseny, M., Hassanien, A. E., Hassan, M. K. (Eds.), Expert Systems in Finance: Smart Financial Applications in Big Data Environments, Routledge, London, pp. 80-96.
11. Ramos, J. (2003), "Using TF-IDF to determine word relevance in document queries", in the Proceedings of the first instructional conference on machine learning, Piscataway, NJ, USA, Vol. 242, pp. 133-142.
12. Seljan, S., Baretić, M., Kučiš, V. (2014), "Information Retrieval and Terminology Extraction In Online Resources for Patients with Diabetes", Collegium antropologicum, Vol. 38, No. 2, pp. 705-710.
13. Seljan, S., Dunđer, I., Stančić, H. (2017), "Extracting Terminology by Language Independent Methods", in the Proceedings of the 2nd International Conference on Translation and Interpreting Studies, Innsbruck, Austria, Peter Lang, pp. 141-147.
14. RStudio (N/A), "Shiny", available at: https://shiny.rstudio.com/ (16 March 2019).
15. Silge, J., Robinson, D. (2016), "tidytext: Text Mining and Analysis Using Tidy Data Principles in R", The Journal of Open Source Software, Vol. 1, No. 3.
16. Stepanić, J., Zoroja, J., Šimičević, V. (2017), "Case Study in Interdisciplinary Scientific Communication: A Decade of the INDECS Journal", Business Systems Research, Vol. 8, No. 2, pp. 101-114.
17. Twitter (N/A), "Twitter Developer", available at: https://developer.twitter.com

(5 April 2019).
18. Uys, J. W., Du Preez, N. D., Uys, E. W. (2008), "Leveraging unstructured information using topic modelling", in the Proceedings of the Portland International Conference on Management of Engineering & Technology, Cape Town, South Africa, IEEE, pp. 955-961.
19. Vijayarani, S., Ilamathi, M. J., Nithya, M. (2015), "Preprocessing techniques for text mining-an overview", International Journal of Computer Science & Communication Networks, Vol. 5, No. 1, pp. 7-16.
20. Wickham, H., Grolemund, G. (2016), R for data science: import, tidy, transform, visualize, and model data, O'Reilly Media, Inc.
21. Wickham, H. (2014), "Tidy data", Journal of Statistical Software, Vol. 59, No. 10, pp. 1-23.

About the authors

Zivko Krstic is a Data Scientist at Atomic Intelligence. He graduated at the Faculty of Economics & Business, University of Split, Department of Information management. Zivko participated in EU FP7 project "FERARI project". His area of expertise are text analytics, big data analytics and data visualization. He participated in development of several big data products such as JupiterOne, Pandora Insight and other. Zivko is author of several paper in field of data science and he participated in many scientific international conferences. The author can be contacted **atzkrstic@atmc.ai**.

Sanja Seljan, Ph. D. is a full professor at the Faculty of Humanities and Social Sciences, University of Zagreb, Department of Information and Communication sciences. Her research interests include language technologies, text mining, data analysis and visualization, machine translation (MT), localization and natural language processing (NLP). She has published more than 70 professional and scientific papers. She was the project manager of the national scientific and research project, three grants, and participated in eight projects (FP7, Tempus, national). She was invited lecturer at 10 European universities, in the European Commission and in the European Parliament. She is a member of different professional associations and international conference and journal review boards. The author can be contacted at **sanja.seljan@ffzg.hr**.

Jovana Zoroja, Ph.D. is an Assistant Professor at the Faculty of Economics and Business, University of Zagreb, Department of Informatics. She received PhD in Information Systems at the Faculty of Economics and Business Zagreb with the dissertation thesis "Influence of the Information and Communication Technologies on the Competitiveness of the European Union Countries". She was also educated at the LSE – Summer School in London in the field of Business Development and ICT Innovation. She participated in Erasmus-Preparatory-Visit-Program in Rimini, Italy. Her main research interests are information and communication technology, e-learning, simulation games and simulation modeling. She is actively engaged in number of science projects (FP7-ICT, bilateral cooperation, national projects). Jovana Zoroja published several scientific papers in international and national journals and participated in many scientific international conferences. The author can be contacted at **jzoroja@efzg.hr.**