A network approach to dimensionality reduction in Text Mining

Un approccio di rete per la riduzione dimensionale nel Text Mining

Michelangelo Misuraca, Germana Scepi and Maria Spano

Abstract There is an ever-increasing interest in developing statistical tools for extracting information from documental repositories. In a Text Mining frame, a knowledge discovery process usually implies a dimensionality reduction of the vocabulary, via a feature selection and/or a feature extraction. Here we propose a strategy designed for reducing dimensionality through a network-based approach. Network tools allow performing the reduction by considering the most important relations among the terms. The effectiveness of the strategy will be shown on a set of tweets about the 2018 Italian General Election.

Abstract *Lo sviluppo di strumenti statistici per estrarre informazioni da archivi documentali è oggi sempre più importante. Nel Text Mining il processo di knowledge discovery solitamente include uno step di riduzione dimensionale, con procedure di selezione o estrazione dei termini. In questo lavoro si propone una strategia per ridurre la dimensionalità attraverso un approccio di rete. Gli strumenti per lanalisi di rete consentono di operare la riduzione considerando le pi importanti relazioni tra i termini. L'efficacia della strategia è mostrata su un insieme di tweet relativi alle Elezioni Italiane del 2018.*

Key words: vector space model, network analysis, community detection.

1 Introduction

The incredible progress of computer technology, as well as the growth of the Internet, has fastened in recent years the transition from analog to digital data communication and storage. This revolution necessarily involved also the statistical rea-

Michelangelo Misuraca
Università della Calabria - Arcavacata di Rende, e-mail: michelangelo.misuraca@unical.it

Germana Scepi, Maria Spano
Università Federico II di Napoli, e-mail: germana.scepi@unina.it, maria.spano@unina.it

soning. In each domain in which Statistics can help researchers in finding trends, revealing patterns or explaining relations, there is nowadays an over-availability of data. This massive amount of data required the development of mining techniques for discovering and extracting knowledge, because only a part of the collected data is relevant and informative with respect to the phenomena of interest.

This instance is particularly important when we consider texts. Texts express a wide, rich range of information, but encode this information in a form difficult to automatically process. Aiming at analysing a collection of documents written in natural language, it is then necessary to transform the original unstructured data into structured data. In this framework, the most common algebraic model for representing documents is the so-called *vector space model* [14]: a document is a vector in the (extremely sparse) space spanned by the terms. Because of vector space model, each document contains a lot of noise. Documents are seen as *bag-of-words*, i.e. as an unordered set of terms, disregarding grammatical and syntactical roles. The focus is on the presence/absence of terms belonging to the collection's vocabulary, their characterisation and their discrimination power. To reduce space dimensionality, it is possible to consider feature selection procedures and/or feature extraction methods. Feature selection allows filtering a subset of the original terms, by excluding the less informative and discriminative ones or by considering only the most relevant ones, with respect to a given criterion. Feature extraction performs a reduction of the original space by combining the terms into new entities. One of the main differences is that selection techniques retain the original meaning of terms, where extraction techniques require an additional effort in interpreting the results.

To reduce the dimensionality and save the readability of the results, here we propose a new feature extraction strategy based on network analysis. Network analyses allow visualising the relations among the terms of a collection by recovering the context of use. This paper is structured as follows. In Section 2 the reference literature is reviewed. Section 3 introduces the problem and describes the proposed strategy. In Section 4 the effectiveness of the proposal is showed on a set of tweets about the 2018 Italian General Election. Final remarks and some possible future development of the research are discussed in Section 5

2 Background and related work

In the statistical analysis of large collections of documents, one of the main problems is the high-dimensionality of data. Once we have pre-processed the documents via the *bag-of-words* coding, every single term belonging to the collection's vocabulary represents a dimension in the vector space. The $documents \times terms$ lexical table − obtained by juxtaposing the different document-vectors − is usually a large and very sparse matrix. Since only a part of the terms is actually relevant for expressing the informative content of the documents, the noise in the data has to be eliminated because it leads to unreliable results and significantly increases the computational

complexity. Two approaches are used to deal with the problem of dimensionality reduction in Text Mining: *feature selection* and *feature extraction*.

Feature selection methods aim at finding a subset of the original terms, by focusing on their importance in determining the content itself. One of the main advantages of these methods is that the selected features retain the original meaning and provides a better understanding of the results. The most common approach is using a *stop-list*. Typically in a stop-list we can find articles, prepositions, conjunctions and interjection. These terms can be considered unuseful, because they have a very general and weak lexical meaning. In addition, stop-lists may include terms related to the main topics listed in the document collection. On the other hand, many methods have been proposed aiming at selecting the most important terms in a collection [1]. The simplest criterion for term selection is the *document frequency*, i.e. the number of documents in which a term occurs. An alternative is the well-known *tf-idf* [13], which jointly considers the frequency of a term in a document and the document frequency of the term in the collection. Other methods for feature selection such as *term strength* [18], *term contribution* [10] and *entropy-based ranking* [3] are based on the concept of documents' similarity. More recently, other two similar methods for feature selection was proposed: *term variance* [9] and *term variance quality* [5]. Term variance evaluates the quality of terms by computing the variance of each term in the collection. It follows the same idea of document frequency, where terms with low frequency are not important. When class labels are available for a document collection, there are also some supervised methods for term selection. Methods such as *information gain*, *mutual information* and *Chi-square* statistics, have been successfully used in text classification [19].

Feature extraction methods aim at minimising the amount of resources required to describe a large collection of documents. The main task of this approach is to obtain the most relevant information from the original features − through some functional mapping − and represent the information in a lower dimensionality space. The dimensionality reduction is reached by constructing linear combinations of the original features that still describe the document collection with sufficient accuracy. The obvious drawback of feature extraction is that the new features may not have a clear physical meaning, therefore the results are difficult to interpret. In textual data analysis, methods like *principal component analysis* (PCA) [7], *lexical correspondences analysis* (LCA) [8], and *latent semantic analysis* (LSA) [4] are widely used. Although they have been developed in different contexts and with different aims, they are based on a common algebraic frame. A low-rank approximation of the lexical table is obtained via a generalised singular value decomposition. The differences among these techniques are related to the weights assigned to the elements, introducing different orthonormalising constraints. LSA is popular in an information retrieval framework for representing the semantic structures in a collection of documents. LCA is generally used to identify the association structure in the lexical table on factorial maps. More interesting is a non-linear approach to the problem, also known as *manifold learning*, that encompass for example the *isometric feature mapping* (ISOMAP) [16] and the *local linear embedding* (LLE) [12].

3 Problem definition and proposed strategy

Basically, both feature selection and feature extraction are carried out on the $documents \times terms$ lexical table. The vector space model ignores the context in which each term is used. It is possible to get back part of the structural and semantic information by constructing a $terms \times terms$ co-occurrence matrix. In general, each element of this latter matrix is the number of times two terms co-occur in the collection. This data structure can be represented as a network, where each vertex is a term and each edge is an element of the matrix different from 0. In this way, we can visualise both single terms and subsets of terms frequently co-occurring together.

Aiming at reducing the original dimensionality by a feature extraction process, in this paper we propose a community detection based strategy. Differently from the methods described in Sec. 2, our strategy preserves the original meaning of the terms and allows better readability. In a network, a community is a set of similar nodes that can be easily grouped. There is a lack of a universally accepted definition of community, but it is well known that most real-world networks display this kind of structures [6]. It is usually thought as a group where vertices are densely inter-connected and sparsely connected to other parts of the network [17]. From a theoretical viewpoint, community detection is not very different from clustering. Several algorithms have been proposed. Traditional approaches are based on the well-known hierarchical or partitional clustering [15]. Divisive approaches do not introduce substantial conceptual advances with respect to traditional ones. The main difference is that instead of removing edges between pairs of vertices with low similarity, inter-cluster edges are removed. The most popular algorithm for community detection was proposed by Newman and Girvan [11]. This work introduced the concept of *modularity* as a stopping criterion for the algorithm. Here in the follow, we consider the *fast-greedy* algorithm [2]. The advantage of using this algorithm is that the problem of choosing a grouping criterion is overcome by the direct use of modularity as optimisation function.

3.1 Basic notation and data structure

Let $\mathbf{T} = \{\mathbf{d_1}, \ldots, \mathbf{d_n}\} \subset \Re^p$ be a set of n document vectors in a term space of dimension p. This set can be represented as a $terms \times documents$ lexical table, where each element t_{ij} represents the number of occurrences of a term i into a document j $(i = 1, \ldots, p; j = 1, \ldots, n)$. Let transform $\mathbf{T}$ into a binary matrix $\mathbf{B}$, where the generic element b_{ij} is equal to 1 if the term i occurred at least once in document j, 0 otherwise. From the matrix $\mathbf{B}$, we derive the $terms \times terms$ co-occurrence matrix $\mathbf{A}$ by the product $\mathbf{A} \equiv \mathbf{B}\mathbf{B}^\top$. The generic element $a_{ii'}$ is the number of documents in which the term i and the term i' co-occur $(i \neq i')$. According to network theory, $\mathbf{A}$ is a $p \times p$ undirected weighted adjacency matrix that can be used to visualise the relations existing among the different terms.

3.2 Network-based feature extraction

On the matrix **A**, we perform a community detection through a fast-greedy algorithm. This algorithm falls in the general family of agglomerative hierarchical clustering methods. As we said above, it is based on the optimisation of a quality function known as modularity. Modularity is the difference between the observed fraction of edges that fall within the given communities and the expected fraction in the hypothesis of random distribution. Suppose that the vertices of matrix **A** can be divided into two communities. The membership to one community or the other one is detected by a variable s, assuming values 1 or -1 respectively. The modularity Q is defined as:

$$Q = \frac{1}{2h} \sum_{ii'} \left[a_{ii'} - \frac{\delta_i \delta_{i'}}{2h} \right] s_i s_{i'} \tag{1}$$

where δ_i is the degree of the i-th term, h is the total number of edges in the network, and s_i represents the membership value of the term i to a community. When we consider G communities, eq. 1 can be expressed in terms of additional contribution ΔQ to the modularity. In matrix form we have:

$$\Delta Q = \frac{1}{4h} \mathbf{s}^{\mathsf{T}} \mathbf{M}^{(g)} \mathbf{s} \tag{2}$$

where $\mathbf{M}^{(g)}$ is the modularity matrix referred to the g-th community ($g = 1, \ldots, G$), and $\mathbf{s}$ is the binary column vector indicating the membership of each term to the community. The fast-greedy algorithm, starting with a state in which each term is the sole member of one of G communities, repeatedly joins communities together in pairs choosing in each step the join that results in the greatest increase in modularity. At the end of the detection process, we obtain a $terms \times communities$ matrix **C**, a complete disjunctive table where the c_{ig} element is 0 or 1 when a term i belongs or not belongs to a community. The result of the dimensionality reduction is a $documents \times communities$ matrix $\mathbf{T}^{\star} \equiv (\mathbf{T}^{\mathsf{T}}\mathbf{C})\mathbf{D}_G{}^{-1}$, where $\mathbf{D}_G{}^{-1}$ is the diagonal matrix obtained from the column marginal distribution of **C**. Each cell of $\mathbf{T}^{\star}$ contains the proportion of terms belonging to a community detected in the documents.

4 A study on the 2018 Italian Election campaign

The campaign for the 2018 General Election in Italy was different from the previous ones. In the past, streets were lined with political posters and candidates rallied potential voters around the country. But for the first time, there was no public refund to Italian parties for their campaign spending. Social media offered a less expensive but affordable way to reach voters in a largely unregulated forum.

Twitter is one of the most popular − and worldwide leading − social networking service. It can be seen as a blend of instant messaging, microblogging and texting, with brief content and a very broad audience. The embryonic idea was developed considering the exchange of texts like Short Message Service in a small group of users. We decided to focus our study on the official Twitter accounts of the first ten Italian political parties, considering the Election results. By using the *Twitter Archiver add-on* for Google Sheet, we collected 6094 tweets. We referred only to the last three months of the 2018 Election campaign, from January 1st to March 4th, 2018. We decided not filtering the so-called retweets, so that in the collection some texts were replicated more times. In Table 1, it is possible to see the main characteristics of the dataset. About 50% of the tweets were posted by the official account of the *Lega* party. Some of the parties, such as *Movimento 5 Stelle* and *Partito Democratico*, seemed to be less active on Twitter, but they showed a greater consensus both regarding the average number of *retweets* and *likes*.

Table 1 Statistics on the parties' accounts from January 1st to March 4th 2018

Party	**# tweets**	**% tweets**	**avg. # retweets**	**avg. # like**
+ *Europa*	417	6.84	37.8	103.2
Civica Popolare	86	1.41	7.5	13.5
Forza Italia	122	2.00	25.0	38.6
Fratelli d'Italia	789	12.95	7.8	18.3
Insieme	201	3.30	10.7	17.0
Lega	3025	49.64	7.3	14.5
Liberi e Uguali	560	9.19	36.3	75.9
Movimento 5 Stelle	134	2.20	232.7	398.3
Partito Democratico	288	4.73	114.5	227.7
Potere al Popolo	472	7.75	44.2	84.7

The pre-processing was performed in two phases, because of the peculiarity of tweets. Firstly, we stripped URLs, usernames, hashtags, emoticons and RT prefixes, and we normalised the tweets by removing other special characters and any separators than the blank spaces. Secondly, we performed a lemmatisation and tagged each term with the corresponding grammatical category. We decided to consider only the substantives and the adjectives because of their significant content-bearing role. Moreover, we deleted the terms appearing in the collection just one time from the vocabulary. The result of this step was a $documents \times terms$ matrix $\mathbf{T}$ with 6094 rows and 3610 columns, and a $terms \times terms$ co-occurrence matrix $\mathbf{A}$.

We performed the community detection procedure on $\mathbf{A}$ to extract the features. For better highlighting the relationships, we fixed a threshold of 15 co-occurrences and deleted the isolated terms. The greedy algorithm detected 20 communities. The high value of modularity ($Q = 0.83$) reveals the effectiveness of the procedure. In Table 2, the size of each community and the characterising terms are showed.

It is interesting to note that the algorithm identifies the communities related to the different parties' electoral manifestos. For instance, *C1* contains terms emphasising

Table 2 Communities in collection of tweets

ID	Size	Terms
C01	16	*fratello, diritto, futuro, italia, grande, unico, paese, ospite, tutto, europa, libertá, civile, proprio, unione, europeo, piúeuropa*
C02	9	*lavoratore, scuola, grasso, aggiunto, account, libero, uguale, cgil, buono*
C03	13	*italiano, marzo, estero, politico, istruzione, simbolo, elezione, melone, giorgia, semplice, palermo, croce, inciucio*
C04	9	*nuovo, comunitá, diretto, forza, momento, pagina, ordine, ufficiale, facebook*
C05	4	*potere, popolo, pubblico, spesa*
C06	9	*programma, partito, voto, utile, governo, pd, sinistro, ambiente, democratico*
C07	7	*anno, euro, popolare, ultimo, prossimo, casa, mila*
C08	2	*piano, natalitá*
C09	6	*centrodestra, berlusconi, lega, salvini, premier, matteo*
C10	2	*milano, duomo*
C11	4	*intervista, live, capitano, replay*
C12	3	*donna, violenza, uomo*
C13	2	*museo, egizio*
C14	2	*fake, news*
C15	3	*campagna, elettorale, legge*
C16	2	*made, italy*
C17	2	*giulia, bongiorno*
C18	2	*flat, tax*
C19	2	*movimento, stella*
C20	2	*sociale, centro*

the role of Italy in the European Union (e.g. *diritto, futuro, grande, unico, paese, libertá,...*) and the name of the party that promotes these aspects (+ *Europa*). Community *C6*, as well as *C9*, contains more trivial terms but specifically related to main opposite coalitions (*Partito Democratico* vs *Lega* and *Forza Italia*). Some of the communities are smaller, containing only a couple of terms, but still very important because they identify key themes of the campaign, such as *flat tax* and *piano natalitá*. The community detection procedure helps in reducing the dimensionality also by automatically identifying collocations and multiword expressions such as *fake news*, *centro sociale*, *made italy*.

By selecting only the terms belonging to the different communities, we obtain a 6094×20 matrix $\mathbf{T}^{\star}$, which can be used for further statistical analyses.

5 Some remarks and future developments

The proposed strategy aims at extracting features from a collection of documents by detecting high-level structures, i.e. communities. Each community is a new feature that retains the meaning of the single terms, and it can be seen as a concept/topic relevant for the domains to which the collection is referred. The strategy is suitable when we deal with short texts, as in the case study, but can also be applied to other kind of documents. One of the advantages of this approach, compared with the other

proposal in the literature, is that the dimensionality is reduced by detecting collocations, multiwords and other structure. This reduction can also be seen as the first step for other analyses. Future developments of this work are devoted to automatically set a co-occurrence threshold in the community detection step, and to evaluate alternative similarity indices for measuring the relation strength among terms.

Acknowledgements The research work was supported by the *Start-(h)open* project, funded in the frame of the OP ERDF ESF 14/20 by the grant J28C17000380006 of the Regional Government of Calabria (Italy). This paper is dedicated to the memory of our beloved colleague, Simona Balbi.

References

1. Bharti, K. K., Singh, P. K.: A Survey on Filter Techniques for Feature Selection in Text Mining. In: Babu, B. V. *et al.* (eds.) Proc. of the 2^{nd} Int. Conference on Soft Computing for Problem Solving (SocProS12), pp. 1545-1559. Springer (2014)
2. Clauset A., Newman, M. E., Moore, C.: Finding community structure in very large networks. Phys. Rev. E. **70**, 066111 (2004)
3. Dash, M., Liu, H.: Feature Selection for Clustering. In: Proc. of the 4^{th} Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD00), pp.110-121. Springer (2000)
4. Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., Harshman, R.: Indexing by Latent Semantic Analysis. J. Am. Soc. Inform. Sci. **41**, 391–407 (1990)
5. Dhillon, I., Kogan, J., Nicholas, C.: Feature selection and document clustering. In: Berry, M.W. (ed.) Survey of Text Mining. Clustering, Classification, and Retrieval, pp. 73-100. Springer (2004)
6. Fortunato, S.: Community detection in graphs. Phys. Rep. **486**, 75–174 (2010)
7. Jolliffe, I. T.: Principal Component Analysis. Springer-Verlag, New York (1986)
8. Lebart, L., Salem, A., Berry, L.: Exploring textual data. Kluwer, Dordrecht (1988)
9. Liu, L., Kang, J., Yu, J., Wang, Z.: A comparative study on unsupervised feature selection methods for text clustering. In: Proc. of the IEEE Int. Conference on Natural Language Processing and Knowledge Engineering (IEEE NLP-KE05), pp. 597-601. IEEE (2005)
10. Liu, T., Liu, S., Chen, Z., Ma, W.: An Evaluation on Feature Selection for Text Clustering. In: Proc. of the 20^{th} Int. Conference on Machine Learning (ICML03), pp. 488-495. ACM (2003)
11. Newman, M. E. J., Girvan, M.: Finding and evaluating community structure in networks. Phys. Rev. E. **69**, 026113 (2004)
12. Roweis, S. T., Saul, L. K.: Nonlinear dimensionality reduction by locally linear embedding. Science. **290**, 2323–2326 (2000)
13. Salton, G., Buckely, C.: Term-weighting approaches in automatic text retrieval. Inform. Process. Manag. **24**, 513–523 (1988)
14. Salton, G., Wong, A., Yang, C. S.: A vector space model for automatic indexing. Commun. ACM. **18**, 613–620 (1975)
15. Scott, J.: Social Network Analysis: a handbook. Sage, London (2000)
16. Tenenbaum, J., de Silva, V., Langford, J.: A global geometric framework for nonlinear dimensionality reduction. Science. **290**, 2319–2323 (2000)
17. Wasserman, S., Faust, K.: Social network analysis. Cambridge University Press (1994)
18. Wilbur, W. J., Sirotkin, K.: The automatic identification of stop words. J. Inform. Sci. **18**, 45–55 (1992)
19. Yang, Y., Pedersen, J. O.: A comparative study on feature selection in text categorization. In: Proc. of the 14^{th} Int. Conference on Machine Learning (ICML97), pp. 412-420. ACM (1997)