Physica A 562 (2021) 125344

Contents lists available at ScienceDirect

Physica A

journal homepage: www.elsevier.com/locate/physa

Using virtual edges to improve the discriminability of co-occurrence text networks

Laura V.C. Quispe, Jorge A.V. Tohalino, Diego R. Amancio *

Institute of Mathematics and Computer Science, Department of Computer Science, University of São Paulo, São Carlos, SP, Brazil

ARTICLE INFO

Article history:
Received 18 April 2020
Received in revised form 2 September 2020
Available online 25 September 2020

Keywords:
Network science
Language networks
Text networks
Co-occurrence networks
Semantic networks
Word embeddings

ABSTRACT

Word co-occurrence networks have been employed to analyze texts both in the practical and theoretical scenarios. Despite the relative success in several applications, traditional co-occurrence networks fail in establishing links between similar words whenever they appear distant in the text. Here we investigate whether the use of word embeddings as a tool to create virtual links in co-occurrence networks may improve the quality of classification systems. Our results revealed that the discriminability in the stylometry task is improved when using *Glove*, *Word2Vec* and *FastText*. In addition, we found that optimized results are obtained when *stopwords* are not disregarded and a simple global thresholding strategy is used to establish virtual links. Because the proposed approach is able to improve the representation of texts as complex networks, we believe that it could be extended to study other natural language processing tasks. Likewise, theoretical languages studies could benefit from the adopted enriched representation of word co-occurrence networks.

1. Introduction

The ability to construct complex and diverse linguistic structures is one of the main features that set us apart from all other species. Despite its ubiquity, some language aspects remain unknown. Topics such as language origin and evolution have been studied by researchers from diverse disciplines, including Linguistic, Computer Science, Physics and Mathematics [1–3]. In order to better understand the underlying language mechanisms and universal linguistic properties, several models have been developed [4,5]. A particular language representation regards texts as complex systems [6]. Written texts can be considered as complex networks (or graphs), where nodes could represent syllables, words, sentences, paragraphs or even larger chunks [6]. In such models, network edges represent the proximity between nodes, e.g. the frequency of the co-occurrence of words. Several interesting results have been obtained from networked models, such as the explanation of Zipf's Law as a consequence of the least effort principle and theories on the nature of syntactical relationships [7,8].

In a more practical scenario, text networks have been used in text classification tasks [9,10]. The main advantage of the model is that it does not rely on deep semantical information to obtain competitive results. Another advantage of graph-based approaches is that, when combined with other approaches, it yields competitive results [11]. A simple, yet recurrent text model is the well-known word co-occurrence network. After optional textual pre-processing steps, in a co-occurrence network each different word becomes a node and edges are established via co-occurrence in a desired window. A common strategy connects only adjacent words in the so called word adjacency networks.

* Corresponding author.
E-mail address: diego@icmc.usp.br (D.R. Amancio).

https://doi.org/10.1016/j.physa.2020.125344
0378-4371/

While the co-occurrence representation yields good results in classification scenarios, some important features are not considered in the model. For example, long-range syntactical links, though less frequent than adjacent syntactical relationships, might be disregarded from a simple word adjacency approach [12]. In addition, semantically similar words not sharing the same lemma are mapped into distinct nodes. In order to address these issues, here we introduce a modification of the traditional network representation by establishing additional edges, referred to as "*virtual*" edges. In the proposed model, in addition to the co-occurrence edges, we link two nodes (words) if the corresponding word embedding representation is similar. While this approach still does not merge similar nodes into the same concept, similar nodes are explicitly linked via virtual edges.

Our main objective here is to evaluate whether such an approach is able to improve the discriminability of word co-occurrence networks in a typical text network classification task. We evaluate the methodology for different embedding techniques, including *GloVe*, *Word2Vec* and *FastText*. We also investigated different thresholding strategies to establish virtual links. Our results revealed, as a proof of principle, that the proposed approach is able to improve the discriminability of the classification when compared to the traditional co-occurrence network. While the gain in performance depended upon the text length being considered, we found relevant gains for intermediary text lengths. Additional results also revealed that a simple thresholding strategy combined with the use of stopwords tends to yield the best results.

We believe that the proposed representation could be applied in other text classification tasks, which could lead to potential gains in performance. Because the inclusion of virtual edges is a simple technique to make the network denser, such an approach can benefit networked representations with a limited number of nodes and edges. This representation could also shed light into language mechanisms in theoretical studies relying on the representation of text as complex networks. Potential novel research lines leveraging the adopted approach to improve the characterization of texts in other applications are presented in the conclusion.

2. Related works

Complex networks have been used in a wide range of fields, including in Social Sciences [13], Neuroscience [14], Biology [15], Scientometry [16–18] and Pattern Recognition [19–22]. In text analysis, networks are used to uncover language patterns, including the origins of the ever present Zipf's Law [23] and the analysis of linguistic properties of natural and unknown texts [24,25]. Applications of network science in text mining and text classification encompasses applications in semantic analysis [26–29], authorship attribution [30,31] and stylometry [30,32,33]. Here we focus in the stylometric analysis of texts using complex networks.

In [30], the authors used a co-occurrence network to study a corpus of English and Polish books. They considered a dataset of 48 novels, which were written by 8 different authors. Differently from traditional co-occurrence networks, some punctuation marks were considered as words when mapping texts as networks. The authors also decided to create a methodology to normalize the obtained network metrics, since they considered documents with variations in length. A similar approach was adopted in a similar study [34], with a focus on comparing novel measurements and measuring the effect of considering stopwords in the network structure.

A different approach to analyze co-occurrence networks was devised in [35]. Whilst most approaches only considered traditional network measurements or devised novel topological and dynamical measurements, the authors combined networked and semantic information to improve the performance of network-based classification. Interesting, the combined use of network motifs and node labels (representing the corresponding words) allowed an improvement in performance in the considered task. Networked-based approaches have also been applied to the authorship recognition tasks in other languages, including Persian texts [9].

Co-occurrence networks have not been used only to perform stylometric analysis. The main advantage of this approach is illustrated in the task aimed at diagnosing diseases via text analysis [11]. Because the topological analysis of co-occurrence language networks do not require deep semantic analysis, this model is able to model text created by patients suffering from cognitive impairment [11]. Recently, it has been shown that the combination of network and traditional features could be used to improve the diagnosis of patients with cognitive impairment [11]. Interestingly, this was one of the first approaches suggesting the use of embeddings to address the particular problem of lack of statistics to create a co-occurrence network in short documents [36].

While many of the works dealing with word co-occurrence networks have been proposed in the last few years, no systematic study of the effects of including information from word embeddings in such networks has been analyzed. This work studies how links created via embeddings information modify the underlying structure of networks and, most importantly, how it can improve the model to provide improved classification performance in the stylometry task.

3. Material and methods

To represent texts as networks, we used the so-called word adjacency network representation [30,34]. Typically, before creating the networks, the text is pre-processed. An optional pre-processing step is the removal of *stopwords*. This step is optional because such words include mostly article and prepositions, which may be artlessly represented by network edges. However, in some applications – including the authorship attribution task – stopwords (or *function words*) play an important role in the stylistic characterization of texts [34]. A list of stopwords considered in this study is available in the Supplementary Information.

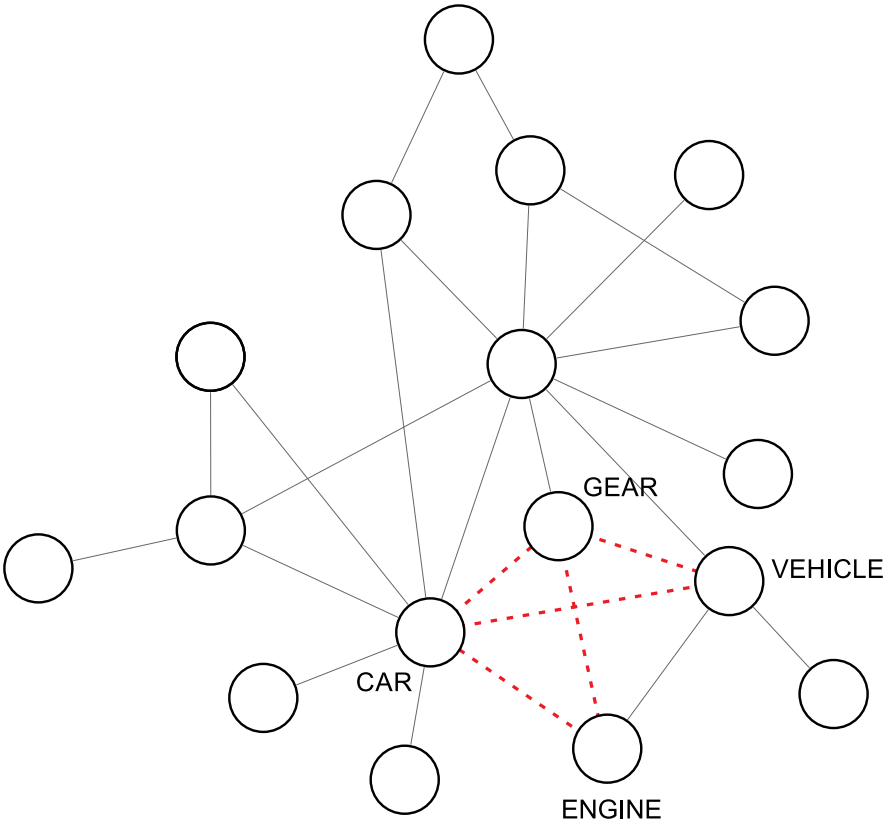

Fig. 1. Example of a enriched word co-occurrence network created for a text. In this model, after the removal of stopwords, the remaining words are linked whenever they appear in the same context. In the proposed network representation, "virtual" edges are included whenever two nodes (words) are semantically related. In this example, virtual edges are those represented by red dashed lines. Edges are included via embeddings similarity. The quantity of included edges is a parameter to be chosen.

The pre-processing step may also include a lemmatization procedure. This step aims at mapping words conveying the same meaning into the same node. In the lemmatization process, nouns and verbs are mapped into their singular and infinite forms. Note that, while this step is useful to merge words sharing a *lemma* into the same node, more complex semantical relationships are overlooked. For example, if "car" and "vehicle" co-occur in the same text, they are considered as distinct nodes, which may result in an inaccurate representation of the text.

Such a drawback is addressed by including "virtual" edges connecting nodes. In other words, even if two words are not adjacent in the text, we include "virtual" edges to indicate that two distant words are semantically related. The inclusion of such virtual edges is illustrated in Figure Fig. 1. In order to measure the semantical similarity between two concepts, we use the concept of *word embeddings* [37,38]. Thus, each word is represented using a vector representation encoding the semantical and contextual characteristics of the word. Several interesting properties have been obtained from distributed representation of words. One particular property encoded in the embeddings representation is the fact the semantical similarity between concepts is proportional to the similarity of vectors representing the words. Similarly to several other works, here we measure the similarity of the vectors via cosine similarity [39].

We used word embeddings to measure the similarity of concepts because this technique has achieved the state-of-the-art in the last few years [40,41]. This model has the advantage of encapsulating both syntactical and semantical properties in a vector representation. Moreover, the computation of words similarity is efficient and simple because it is derived directly from a cosine similarity computation. While wordnets are a different possibility to measure words similarity [42], the use of wordnets is associated with some disadvantages. For example, wordnets are outperformed by word embeddings in similarity tasks. In addition, the computation of a similarity index is not trivial. If the graph structure is used, similarity can be measured according to many different criteria (e.g. shortest path length, structural similarity, neighbors similarity etc.). All these features are encapsulated in the concept of word embeddings [43]. Finally, similarities obtained from wordnets may require the use of a word sense disambiguation method.

The following strategies to create word embedding were considered in this paper:

1. *GloVe*: the Global Vectors (*GloVe*) algorithm is an extension of the *Word2Vec* model [44] for efficient word vector learning [45]. This approach combines global statistics from matrix factorization techniques (such as latent semantic

analysis) with context-based and predictive methods like *Word2Vec*. This method is called as Global Vector method because the global corpus statistics are captured by *GloVe*. Instead of using a window to define the local context, *GloVe* constructs an explicit word-context matrix (or co-occurrence matrix) using statistics across the entire corpus. The final result is a learning model that oftentimes yields better word vector representations [45].

2. *Word2Vec*: this is a predictive model that finds dense vector representations of words using a three-layer neural network with a single hidden layer [44]. It can be defined in a two-fold way: continuous bag-of-words and skip-gram model. In the latter, the model analyzes the words of a set of sentences (or corpus) and attempts to predict the neighbors of such words. For example, taking as reference the word "Robin", the model decides that "Hood" is more likely to follow the reference word than any other word. The vectors are obtained as follows: given the vocabulary (generated from all corpus words), the model trains a neural network with the sentences of the corpus. Then, for a given word, the probabilities that each word follows the reference word are obtained. Once the neural network is trained, the weights of the hidden layer are used as vectors of each corpus word.
3. *FastText*: this method is another extension of the *Word2Vec* model [46]. Unlike *Word2Vec*, *FastText* represents each word as a bag of character n-grams. Therefore, the neural network not only trains individual words, but also several n-grams of such words. The vector for a word is the sum of vectors obtained for the character n-grams composing the word. For example, the embedding obtained for the word "computer" with $n \leq 3$ is the sum of the embeddings obtained for "co", "com", "omp", "mpu", "put", "ute", "ter" and "er". In this way, this method obtains improved representations for rare words, since n-grams composing rare words might be present in other words. The *FastText* representation also allows the model to understand suffixes and prefixes. Another advantage of FastText is its efficiency to be trained in very large corpora.

Note that a thresholding process is required because only a small fraction can be included before syntactical information is lost. If all virtual (weighted) edges were included, all texts with the same vocabulary set would lead to the same network, regardless of words order. This is not desirable in stylometric tasks, because words order is also relevant for the analysis [47].

While multilayer networks could be used here to represent different types of edges, we decided to model text networks with only one layer without distinguishing co-occurrence from virtual and virtual edges (when characterizing the network structure). Our main purpose here is not to analyze the whole semantic network created by virtual (semantical) edges. We actually desire to include only a few additional edges to capture important edges that might be missing from the co-occurrence model. Only a few edges virtual edges are included because the inclusion of many virtual edges can remove the information of word ordering, a feature that is certainly important in many stylometric tasks [12,36,48]. In this sense, if we used a different layer to represent such a small fraction of edges, we would probably obtain only a set of many unconnected components.

Concerning the thresholding process, we considered two main strategies. First, we used a global strategy: in addition to the co-occurrence links (continuous lines in Fig. 1), only "virtual" edges stronger than a given threshold are left in the network. Thus only the most similar concepts are connected via virtual links. This strategy is hereafter referred to as *global* strategy. Unfortunately, this method may introduce an undesired bias towards *hubs* [49].

To overcome the potential disadvantages of the global thresholding method, we also considered a more refined thresholding approach that takes into account the local structure to decide whether a weighted link is statistically significant [49]. This method relies on the idea that the importance of an edge should be considered in the context in which it appears. In other words, the relevance of an edge should be evaluated by analyzing the nodes connected to its ending points. Using the concept of *disparity filter*, the method devised in [49] defines a null model that quantifies the probability of a node to be connected to an edge with a given weight, based on its other connections. This probability is used to define the significance of the edge. The parameter that is used to measure the significance of an edge e_{ij} is α_{ij}, defined as:

$$\alpha_{ij} = 1 - (k_i - 1) \int_0^{\pi_{ij}} (1 - x)^{k_i - 2} dx, \tag{1}$$

$$\pi_{ij} = w_{ij} \left(\sum_{ik \in E} w_{ik} \right)^{-1}, \tag{2}$$

where w_{ij} is the weight of the edge e_{ij} and k_i is the degree of the ith node. The obtained network corresponds to the set of nodes and edges obtained by removing all edges with α higher than the considered threshold. Note that while the similarity between co-occurrence links might be considered to compute α_{ij}, only "virtual" edges (i.e. the dashed lines in Fig. 1) are eligible to be removed from the network in the filtering step. This strategy is hereafter referred to as *local* strategy.

After co-occurrence networks are created and virtual edges are included, in the next step we used a characterization based on topological analysis. Because a global topological analysis is prone to variations in network size, we focused our analysis on the local characterization of complex networks. In a local topological analysis, we use as features the value of topological/dynamical measurements obtained for a set of words. In this case, we selected as feature the words occurring in all books of the dataset. For each word, we considered the following network measurements: degree, betweenness,

clustering coefficient, average shortest path length, PageRank, concentric symmetry (at the second and third hierarchical level) [34] and accessibility [50] (at the second and third hierarchical level). We chose these measurements because all of them capture some particular linguistic feature of texts [48,51–53]. After network measurements are extracted, they are used in machine learning algorithms. In our experiments, we considered Decision Trees[1] (DT), nearest neighbors (kNN), Naive Bayes (NB) and Support Vector Machines (SVM). We used some heuristics to optimize classifier parameters. Such techniques are described in the literature. The list of optimized parameters are mentioned in [54]. The accuracy of the pattern recognition methods were evaluated using cross-validation [55].

In summary, the methodology used in this paper encompasses the following steps:

1. *Network construction*: here texts are mapped into a co-occurrence networks. Some variations exists in the literature, however here we focused in the most usual variation, i.e. the possibility of considering or disregarding stopwords. A network with co-occurrence links is obtained after this step.
2. *Network enrichment*: in this step, the network is enriched with virtual edges established via similarity of word embeddings. After this step, we are given a complete network with weighted links. Virtually, any embedding technique could be used to gauge the similarity between nodes.
3. *Network filtering*: in order to eliminate spurious links included in the last step, the weakest edges are filtered. Two approaches were considered: a simple approach based on a global threshold and a local thresholding strategy that preserves network community structure. The outcome of this network filtering step is a network with two types of links: co-occurrence and virtual links (as shown in Fig. 1).
4. *Feature extraction*: In this step, topological and dynamical network features are extracted. Here, we do not discriminate co-occurrence from virtual edges to compute the network metrics.
5. *Pattern classification*: once features are extracted from complex networks, they are used in pattern classification methods. This might include supervised, unsupervised and semi-supervised classification. This framework is exemplified in the supervised scenario.

The above framework is exemplified with the most common technique(s). It should be noted that the methods used, however, can be replaced by similar techniques. For example, the network construction could consider stopwords or even punctuation marks [56]. Another possibility is the use of different strategies of thresholding. While a systematic analysis of techniques and parameters is still required to reveal other potential advantages of the framework based on the addition of virtual edges, in this paper we provide a first analysis showing that virtual edges could be useful to improve the discriminability of texts modeled as complex networks.

Here we used a dataset compatible with datasets used recently in the literature (see e.g. [10,30,57]). The objective of the studied stylometric task is to identify the authorship of an unknown document [58]. All data and some statistics of each book are shown in the Supplementary Information.

4. Results and discussion

In Section 4.1, we probe whether the inclusion of virtual edges is able to improve the performance of the traditional co-occurrence network-based classification in a usual stylometry task. While the focus of this paper is not to perform a systematic analysis of different methods comprising the adopted network, we consider two variations in the adopted methodology. In Section 4.2, we consider the use of stopwords and the adoption of a local thresholding process to establish different criteria to create new virtual edges.

4.1. Performance analysis

In this section, we show the performance results obtained when a small fraction of edges are included. Before analyzing performance, we investigated the effects of including virtual edges on network topology. Detailed results are provided in the Figure S1 of the Supplementary Information. For some measurements the behavior is trivial, since it is well known that correlation exists between network metrics and the average network degree [59]. The inclusion of edges tends to increase the following measurements: degree, clustering coefficient, closeness and accessibility. Conversely, a clear decrease in value as virtual edges are included is observed for the shortest path length. A larger variation in the behavior was observed for the symmetry measurement. This was expected because this measurement does not depend on the number of edges, but on how regular neighbors are reached via random walks [34].

Concerning the performance analysis, the following parameters were considered in the analysis. For each network, we considered the following fraction of additional edges $p = \{1\%, 2\%, 3\%, \ldots, 20\%\}$. Results obtained with a higher fraction are not shown because, in preliminary experiments, we found no further significant gain in performance. In addition, we avoided including too many virtual edges because syntactical information might be lost as more virtual edges are included. As a consequence, measurements used in stylometric can become uninformative [60].

In Fig. 2, we show the highest improvements in performance obtained when including a fixed amount of virtual edges using *GloVe* as embedding method. The results are sorted by gain, for different values of p. Each subpanel corresponds

[1] This includes Random Forests.

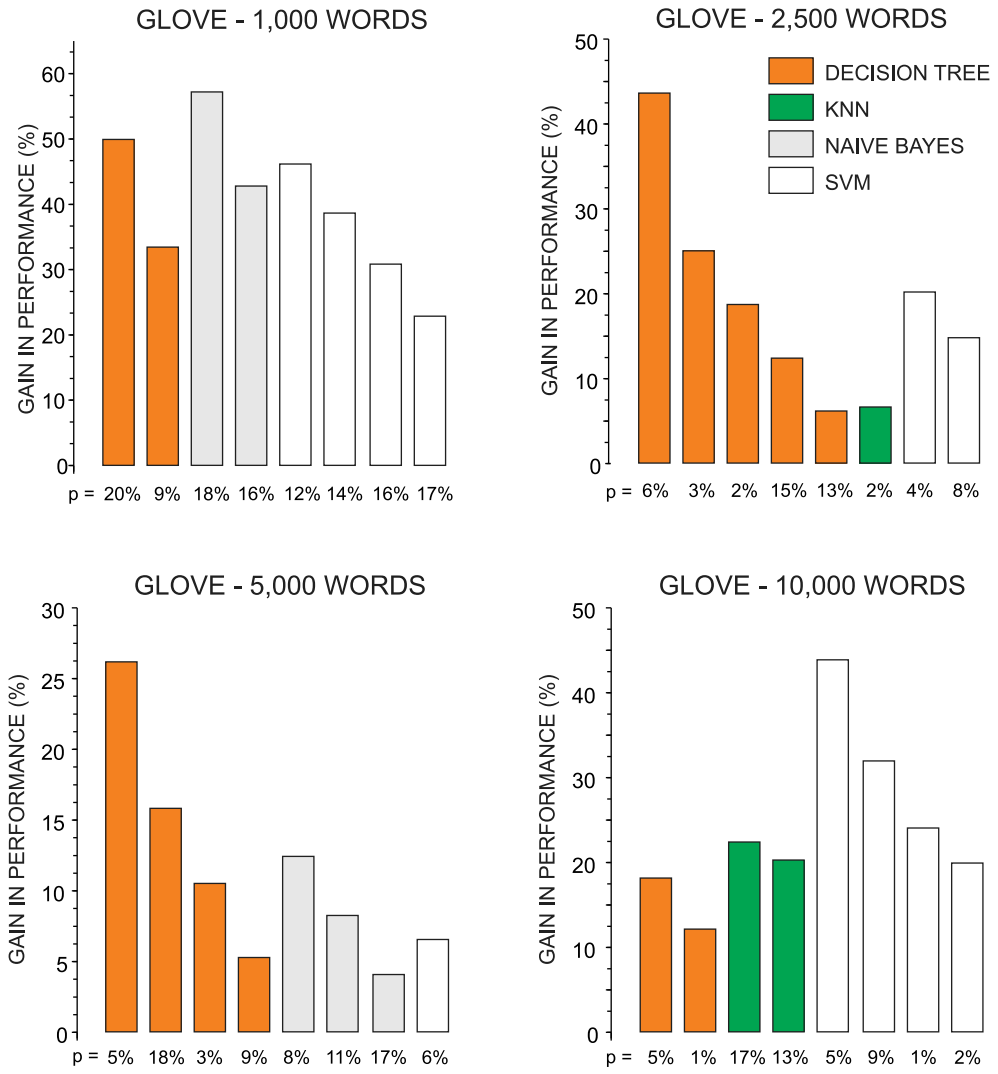

Fig. 2. Gain in performance when considering additional virtual edges created using *GloVe* as embedding method. Each sub-panel shows the results obtained for distinct values of text length. For each document length, we show the highest improvements in performance. Each color corresponds to the improvement obtained in different classifiers.

to a different text length. In this section, we considered the traditional co-occurrence network as starting point. In other words, the network construction disregarded stopwords. The list of stopwords considered in this paper is available in the Supplementary Information. We also considered the global approach to filter edges.

The relative improvement in performance is given by $\Gamma_+(p)/\Gamma_0$, where $\Gamma_+(p)$ is the accuracy rate obtained when p% additional edges are included and $\Gamma_0 = \Gamma_+(p = 0)$, i.e. Γ_0 is the accuracy rate measured from the traditional co-occurrence model. In our analysis, we considered also samples of text with distinct length, since the performance of network-based methods is sensitive to text length [36]. In this figure, we considered samples comprising $w = \{1.0, 2.5, 5.0, 10.0\}$ thousand words.

The results obtained for *GloVe* show that the highest relative improvements in performance occur for decision trees. This is apparent specially for the shortest samples. For $w = 1000$ words, the decision tree accuracy is enhanced by a factor of almost 50% when $p = 20$%. An excellent gain in performance is also observed for both Naive Bayes and SVM classifiers, when $p = 18$% and $p = 12$%, respectively. When $w = 2500$ words, the highest improvements was observed for the decision tree algorithm. A minor improvement was observed for the kNN method. A similar behavior occurred for $w = 5000$ words. Interestingly, SVM seems to benefit from the use of additional edges when larger documents are considered. When only 5% virtual edges are included, the relative gain in performance is about 45%. A summary of results obtained with all of the considered parameters is available in the Supplementary Information.

The relative gain in performance obtained for *Word2Vec* is shown in Fig. 3. Overall, once again decision trees obtained the highest gain in performance when short texts are considered. Similar to the analysis based on the *GloVe* method, the

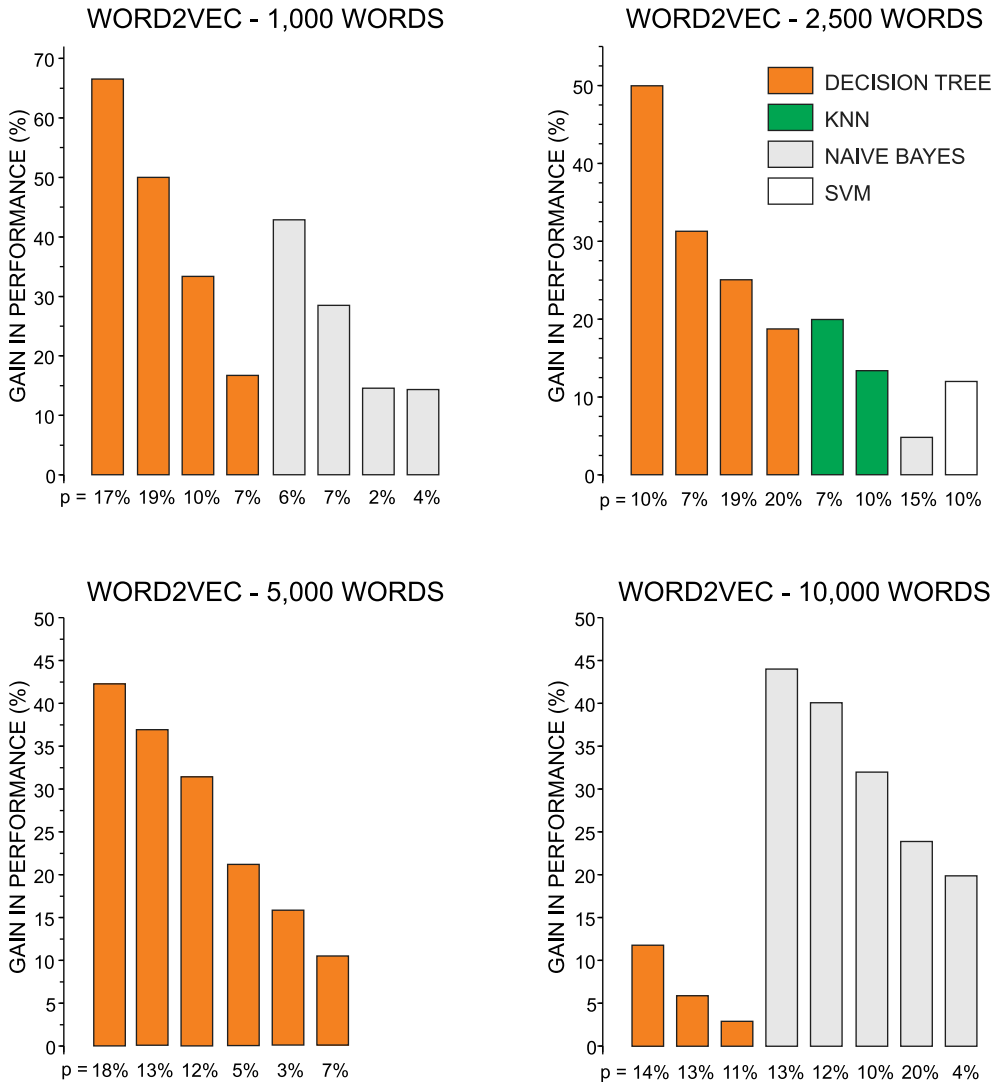

Fig. 3. Gain in performance when considering additional virtual edges created using *Word2Vec* as embedding method. Each sub-panel shows the results obtained for distinct values of text length. For each document length, we show the highest improvements in performance. The results are sorted by gain. Each color corresponds to the improvement obtained in different classifiers.

gain for kNN is low when compared to the benefit received by other methods. Here, a considerable gain for SVM in only clear for $w = 2500$ and $p = 10\%$. When large texts are considered, Naive Bayes obtained the largest gain in performance.

Finally, the relative gain in performance obtained for *FastText* is shown in Fig. 4. The prominent role of virtual edges in decision tree algorithm in the classification of short texts once again is evident. Conversely, the classification of large documents using virtual edges mostly benefit the classification based on the Naive Bayes classifier. Similarly to the results observed for *Glove* and *Word2Vec*, the gain in performance obtained for kNN is low compared when compared to other methods. Another interesting finding is that the behavior of the performance as p increases is not a monotonic function (see Table S9 of the Supplementary Information).

While Figs. 2–4 show the relative behavior in the accuracy, it still interesting to observe the absolute accuracy rate obtained with the classifiers. In Table 1, we show the best accuracy rate (i.e. $\max \Gamma_+ = \max_p \Gamma_+(p)$) for *GloVe*. We also show the average difference in performance ($\langle \Gamma_+ - \Gamma_0 \rangle$) and the total number of cases in which an improvement in performance was observed (N_+). N_+ ranges in the interval $0 \leq N_+ \leq 20$. Table 1 summarizes the results obtained for $w = \{1.0, 5.0, 10.0\}$ thousand words. Additional results for other text length are available in Tables S3–S5 of the Supplementary Information.

In very short texts, despite the low accuracy rates, an improvement can be observed in all classifiers. The best results were obtained with SVM when virtual edges were included. For $w = 5000$ words, the inclusion of new edges has no

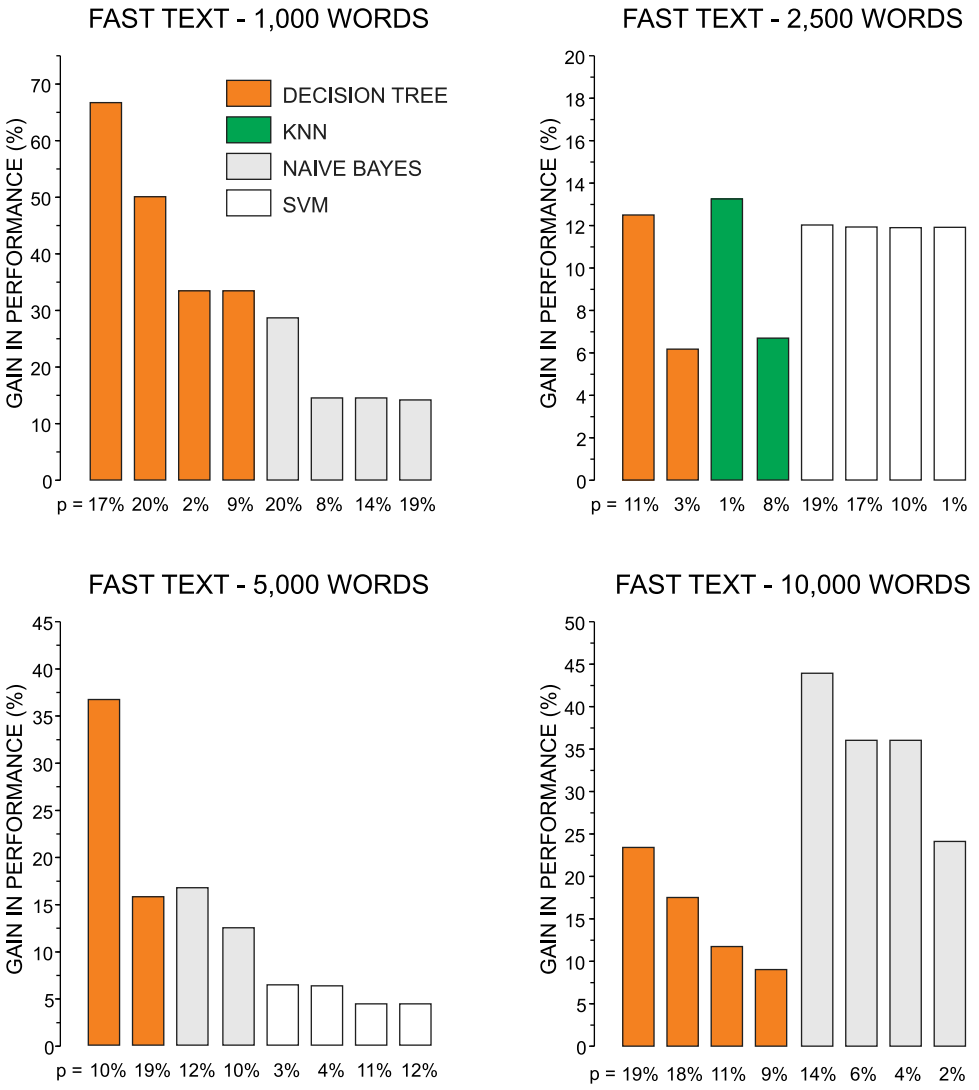

Fig. 4. Gain in performance when considering additional virtual edges created using *FastText* as embedding method. Each sub-panel shows the results obtained for distinct value of text length. The results are sorted by gain. Each color corresponds to the improvement obtained in different classifiers.

positive effect on both kNN and Naive Bayes algorithms. On the other hand, once again SVM could be improved, yielding an optimized performance. For $w = 10{,}000$ words, SVM could not be improved. However, even without improvement it yielded the maximum accuracy rate. The Naive Bayes algorithm, in average, could be improved by a margin of about 10%.

The results obtained for *Word2Vec* are summarized in Table S4 of the Supplementary Information. Considering short documents ($w = 1000$ words), here the best results occur only with the decision tree method combined with enriched networks. Differently from the *GloVe* approach, SVM does not yield the best results. Nonetheless, the highest accuracy across all classifiers and values of p is the same. For larger documents ($w = 5000$ and $w = 10{,}000$ words), no significant difference in performance between *Word2Vec* and *GloVe* is apparent.

The results obtained for *FastText* are shown in Table 2. In short texts, only kNN and Naive Bayes have their performance improved with virtual edges. However, none of the optimized results for these classifiers outperformed SVM applied to the traditional co-occurrence model. Conversely, when $w = 5000$ words, the optimized results are obtained with virtual edges in the SVM classifier. Apart from kNN, the enriched networks improved the traditional approach in all classifiers. For large chunks of texts ($w = 10{,}000$), once again the approach based on SVM and virtual edges yielded optimized results. All classifiers benefited from the inclusion of additional edges. Remarkably, Naive Bayes improved by a margin of about 13%.

Table 1
Statistics of performance obtained with *GloVe* for different text lengths. Additional results considering other text lengths are shown in the Supplementary Information. Γ_0 is the accuracy rate obtained with the traditional co-occurrence model and *max* Γ_+ is the highest accuracy rate considering different number of additional virtual edges. $\langle \Gamma_+ - \Gamma_0 \rangle$ is the average absolute improvement in performance, $\langle \Gamma_+ / \Gamma_0 \rangle$ is the average relative improvement in performance and N_+ is the total number of cases in which an improvement in performance was observed. In total we considered 20 different cases, which corresponds to the addition of $p = 1\%, 2\% \ldots 20\%$ additional virtual edges. The best result for each document length is highlighted.

	1000 words			
	DT	KNN	NB	SVM
Γ_0	15.38%	8.97%	8.97%	14.10%
max Γ_+	**16.67**%	10.26%	11.54%	**16.67**%
$\langle \Gamma_+ - \Gamma_0 \rangle$	1.29	1.29	1.61	2.57
$\langle \Gamma_+ / \Gamma_0 \rangle$	1.08	1.14	1.18	1.18
N_+	3	3	8	2
	5000 words			
Γ_0	24.36%	43.59%	30.77%	58.97%
max Γ_+	34.62%	–	–	**61.54**%
$\langle \Gamma_+ - \Gamma_0 \rangle$	6.70	–	–	2.57
$\langle \Gamma_+ / \Gamma_0 \rangle$	1.27	–	–	1.04
N_+	18	0	0	1
	10,000 words			
Γ_0	42.31%	62.82%	32.05%	**85.90**%
max Γ_+	48.72%	74.36%	46.15%	–
$\langle \Gamma_+ - \Gamma_0 \rangle$	2.84	5.06	9.68	–
$\langle \Gamma_+ / \Gamma_0 \rangle$	1.07	1.08	1.30	–
N_+	14	18	20	0

Table 2
Statistics of performance obtained with *FastText* for different text lengths. Additional results considering other text lengths are shown in the Supplementary Information. Γ_0 is the accuracy rate obtained with the traditional co-occurrence model and $\max \Gamma_+$ is the highest accuracy rate considering different number of additional virtual edges. $\langle \Gamma_+ - \Gamma_0 \rangle$ is the average absolute improvement in performance, $\langle \Gamma_+ / \Gamma_0 \rangle$ is the average relative improvement in performance and N_+ is the total number of cases in which an improvement in performance was observed. In total we considered 20 different cases, which corresponds to the addition of $p = 1\%, 2\% \ldots 20\%$ additional virtual edges. The best result for each document length is highlighted.

	1000 words			
	DT	KNN	NB	SVM
Γ_0	**15.38** %	8.97%	8.97%	14.10%
$\max \Gamma_+$	–	10.26%	11.54%	–
$\langle \Gamma_+ - \Gamma_0 \rangle$	–	1.29	1.57	–
$\langle \Gamma_+ / \Gamma_0 \rangle$	–	1.14	1.18	–
N_+	0	2	9	0
	5000 words			
Γ_0	24.36%	43.59%	30.77%	58.97%
$\max \Gamma_+$	33.33%	–	35.90%	**62.82**%
$\langle \Gamma_+ - \Gamma_0 \rangle$	3.33	–	2.96	2.34
$\langle \Gamma_+ / \Gamma_0 \rangle$	1.14	–	1.10	1.04
N_+	10	0	13	11
	10,000 words			
Γ_0	42.31%	62.82%	32.05%	85.90%
$\max \Gamma_+$	53.85%	76.92%	48.72%	**87.18**%
$\langle \Gamma_+ - \Gamma_0 \rangle$	6.49	9.17	12.96	1.28
$\langle \Gamma_+ / \Gamma_0 \rangle$	1.15	1.15	1.40	1.01
N_+	54	20	20	4

4.2. Effects of considering stopwords and local thresholding

While in the previous section we focused our analysis on the traditional word co-occurrence model (i.e. networks without *stopwords* and created via global thresholding strategy), here we probe if the idea of considering virtual edges can also yield optimized results in particular modifications of the framework described in the methodology. We considered two independent modifications in the original co-occurrence model: first we considered the use of stopwords. Then we

Table 3
Performance analysis of the adopted framework when considering *stopwords* in the construction of the networks. Only the best results obtained across all considered classifiers are shown. In this case, all optimized results were obtained with SVM. Γ_0 corresponds to the accuracy obtained with no virtual edges and max Γ_+ is the best accuracy rate obtained when including virtual edges. For each text length, the highest accuracy rate is highlighted. A full list of results for each classifier is available in the Supplementary Information.

Length (words)	Γ_0	max Γ_+ (*GloVe*)	max Γ_+ (*Word2Vec*)	max Γ_+ (*FastText*)
1,000	**29.49%**	29.49%	29.49%	29.49%
1,500	37.18%	37.18%	37.18%	**38.46%**
2,000	30.77%	34.62%	**35.90%**	**35.90%**
2,500	41.03%	48.72%	**51.28%**	48.72%
5,000	62.82%	**65.38%**	64.10%	**65.38%**
10,000	**88.46%**	88.46%	88.46%	88.46%

Table 4
Comparison between the best results obtained via global and local thresholding. For each text length and embedding method, we show $\max \Gamma_+^{(G)}/\max \Gamma_+^{(L)}$, where $\Gamma_+^{(G)}$ and $\Gamma_+^{(L)}$ are the accuracy obtained with global and local thresholding strategy, respectively. We only show the results obtained with the SVM, since it turned out to be the classifier yielding the highest accuracy rates. These results point that the use of this local strategy in the filtering process does not improve the performance of the classification.

Length	GloVe	Word2Vec	FastText
1,000	1.026	1.026	1.079
1,500	1.122	1.093	1.019
2,000	1.068	1.091	1.091
2,500	1.020	1.061	1.082
5,000	1.036	1.054	1.071
10,000	1.045	1.030	1.015

modified the original model by creating networks with a local thresholding strategy. The first modification in the co-occurrence model is the use of *stopwords*. While in semantical application of network language modeling *stopwords* are disregarded, in other application *stopwords* can unravel interesting linguistic patterns [10]. Here we analyzed the effect of using *stopwords* in enriched networks. We summarize the obtained results in Table 3. We only show the results obtained with SVM, as it yielded the best results in comparison to other classifiers. The accuracy rate for other classifiers is shown in the Supplementary Information.

The results in Table 3 reveal that even when *stopwords* are considered in the original model, an improvement can be observed with the addition of virtual edges. However, the results show that the degree of improvement depends upon the text length. In very short texts ($w = 1000$), none of the embeddings strategy was able to improve the performance of the classification. For $w = 1,500$, a minor improvement was observed with *FastText*: the accuracy increased from $\Gamma_0 = 37.18\%$ to 38.46%. A larger improvement could be observed for $w = 2000$. Both *Word2Vec* and *FastText* approaches allowed an increase of more than 5% in performance. A gain higher than 10% was observed for $w = 2500$ with *Word2Vec*. For larger pieces of texts, the gain is less expressive or absent. All in all, the results show that the use of virtual edges can also benefit the network approach based on *stopwords*. However, no significant improvement could be observed with very short and very large documents. The comparison of all three embedding methods showed that no method performed better than the others in all cases. When comparing the best values of max Γ_+ obtained with and without *stopwords* (see Tables S3–S8 of the Supplementary Information), we found that the best performance is always achieved with *stopwords*. In this case, the highest improvements in accuracy were found for the shortest texts.

We also investigated if more informed thresholding strategies could provide better results. While the simple global thresholding approach might not be able to represent more complex structures, we also tested a more robust method based on the local approach proposed by Serrano et al. [49]. Both local and global thresholding strategies were analyzed in the traditional model where *stopwords* are disregarded. In Table 4, we summarize the results obtained with this thresholding strategies. The table shows $\max \Gamma_+^{(G)}/\max \Gamma_+^{(L)}$, where $\Gamma_+^{(G)}$ and $\Gamma_+^{(L)}$ are the accuracy obtained with the global and local thresholding strategy, respectively. The results were obtained with the SVM classifier, as it turned to be the most efficient classification method. We found that there is no gain in performance when the local strategy is used. In particular cases, the global strategy is considerably more efficient. This is the case e.g. when *GloVe* is employed in texts with $w = 1,500$ words. The performance of the global strategy is 12.2% higher than the one obtained with the global method. A minor difference in performance was found in texts comprising $w = 1000$ words, yet the global strategy is still more efficient than the local one.

Table 5
Summary of best results obtained in this paper. For each document length we show the highest accuracy rate obtained, the relative gain obtained with the proposed approach and the embedding method yielding the highest accuracy rate: *GloVe* (GL), *Word2Vec* (W2V) or *FastText* (FT). All the results below were obtained when stopwords were used and the SVM was used as classification method.

Length	Accuracy	Gain	Embedding
1,000	29.49%	–	–
1,500	38.46%	3.44%	FT
2,000	35.90%	16.67%	W2V, FT
2,500	51.28%	24.98%	W2V
5,000	65.38%	4.07%	GL, FT
10,000	88.46%	–	–

To summarize all results obtained in this study we show in Table 5 the best results obtained for each text length. We also show the relative gain in performance with the proposed approach and the embedding technique yielding the best result. All optimized results were obtained with the use of stopwords, global thresholding strategy and SVM as classification algorithm. A significant gain is more evident for intermediary text lengths.

All in all, our results confirms our hypothesis that virtual edges can improve the performance of the classification in text network tasks. To test the robustness of results, we also verified in additional datasets if the strategy can also improve the discriminability of the traditional co-occurrence model. The results in Tables S10–S12 of the Supplementary Information confirm that improved results can also observed in other datasets. The highest improvements were observed when discriminating larger number of classes. While we focused in texts analysis, it remains to be shown in future studies that virtual edges could be used to enrich the structure of other types of networks and thus improve the classification of other complex systems.

5. Conclusion

Textual classification remains one of the most important facets of the Natural Language Processing area. Here we studied a family of classification methods, the word co-occurrence networks. Despite this apparent simplicity, this model has been useful in several practical and theoretical scenarios. We proposed a modification of the traditional model by establishing virtual edges to connect nodes that are semantically similar via word embeddings. The reasoning behind this strategy is the fact the similar words are not properly linked in the traditional model and, thus, important links might be overlooked if only adjacent words are linked.

Taking as reference task a stylometric problem, we showed – as a proof of principle – that the use of virtual edges might improve the discriminability of networks. When analyzing the best results for each text length, apart from very short and long texts, the proposed strategy yielded optimized results in all cases. The best classification performance was always obtained with the SVM classifier. In addition, we found an improved performance when stopwords are used in the construction of the enriched co-occurrence networks. Finally, a simple global thresholding strategy was found to be more efficient than a local approach that preserves the community structure of the networks. Because complex networks are usually combined with other strategies [11], we believe that the proposed could be used in combination with other methods to improve the classification performance of other text classification tasks.

The main contribution of our manuscript was to analyze whether the inclusion of virtual edges is able of improving the quality of the classification. While we did not provide a strategy to select the most effective of p (i.e. the fraction of additional edges), this parameter can be identified via optimization strategy [61], as it is the case of similar networked-based classification systems [47]. In future works, it would be interesting to develop a strategy to identify, beforehand, the optimal value of p before any measurement is extracted from the network. In our study, we also found that it is not trivial to know why different fraction of virtual edges can lead to different improvements in performance, since virtual edges are distributed over the network in unpredictable ways. While virtual edges certainly provide a semantical improvement (similar words are linked), at the same time syntactical information can be lost (word ordering is lost when a virtual edge is included) [60]. Because it is not trivial to identify a priori which of these two characteristics is more important for the classification process for each edge, the quality of the classification may vary for different fractions of additional edges. Future studies should further investigate whether there is some pattern that is able to identify a priori if syntax or semantic plays the most important role in a particular edge.

Our findings paves the way for research in several new directions. While we probed the effectiveness of virtual edges in a specific text classification task, we could extend this approach for general classification tasks. A systematic comparison of embeddings techniques could also be performed to include other recent techniques [62,63]. We could also identify other relevant techniques to create virtual edges, allowing thus the use of the methodology in other networked systems. For example, a network could be enriched with embeddings obtained from graph embeddings techniques. A simpler approach could also consider link prediction [64] to create virtual edges. Finally, other interesting family of studies concerns the discrimination between co-occurrence and virtual edges, possibly by creating novel network measurements considering heterogeneous links.

CRediT authorship contribution statement

Laura V.C. Quispe: Software, Validation, Data curation, Formal analysis. **Jorge A.V. Tohalino:** Software, Validation, Data curation, Formal analysis. **Diego R. Amancio:** Conceptualization, Methodology, Formal analysis, Investigation, Resources, Visualization, Supervision, Project administration, Writing - original draft, Writing - review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The authors acknowledge financial support from CNPq-Brazil (Grant no. 304026/2018-2). This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001.

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.physa.2020.125344.

References

[1] M. Faggian, F. Ginelli, F. Rosas, Z. Levnajić, Synchronization in time-varying random networks with vanishing connectivity, Sci. Rep. 9 (1) (2019) 1–11.
[2] X. Kong, L. Liu, S. Yu, A. Yang, X. Bai, B. Xu, Skill ranking of researchers via hypergraph, PeerJ Comput. Sci. 5 (2019) e182.
[3] Y. Shimada, M. Tatara, K. Fujiwara, T. Ikeguchi, Formation mechanisms of local structures in language networks, Europhys. Lett. 127 (5) (2019) 56003.
[4] S. Miller, R. Bobrow, R. Ingria, R. Schwartz, Hidden understanding models of natural language, in: Proceedings of the 32nd Annual Meeting on Association for Computational Linguistics, Association for Computational Linguistics, 1994, pp. 25–32.
[5] A. Baronchelli, R. Ferrer-i Cancho, R. Pastor-Satorras, N. Chater, M.H. Christiansen, Networks in cognitive science, Trends cogn. sci. 17 (7) (2013) 348–360.
[6] J. Cong, H. Liu, Approaching human language with complex networks, Phys. Life Rev. 11 (4) (2014) 598–618.
[7] R.F. Cancho, R.V. Solé, Least effort and the origins of scaling in human language, Proc. Natl. Acad. Sci. 100 (3) (2003) 788–791.
[8] R. Cancho, Why do syntactic links not cross?, Europhys. Lett. 76 (6) (2006) 1228.
[9] A. Mehri, A.H. Darooneh, A. Shariati, The complex networks approach for authorship attribution of books, Physica A 391 (7) (2012) 2429–2437.
[10] S. Segarra, M. Eisen, A. Ribeiro, Authorship attribution through function word adjacency networks, IEEE Trans. Signal Process. 63 (20) (2015) 5464–5478.
[11] L.B. Santos, E.A. Corrêa Jr, O.N. Oliveira Jr, D.R. Amancio, L.L. Mansur, S.M. Aluísio, Enriching complex networks with word embeddings for detecting mild cognitive impairment from speech transcripts, in: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, 2017, pp. 1284.
[12] R.F. Cancho, R.V. Solé, R. Köhler, Patterns in syntactic dependency networks, Phys. Rev. E 69 (5) (2004) 051915.
[13] S.P. Borgatti, A. Mehra, D.J. Brass, G. Labianca, Network analysis in the social sciences, Science 323 (5916) (2009) 892–895.
[14] B.C. Van Wijk, C.J. Stam, A. Daffertshofer, Comparing brain networks of different size and connectivity density using graph theory, PLoS ONE 5 (10) (2010) e13701.
[15] F.A. Rodrigues, L.F. Costa, A.L. Barbieri, Resilience of protein–protein interaction networks as determined by their large-scale topological features, Mol. Biosyst. 7 (4) (2011) 1263–1269.
[16] A. Zeng, Z. Shen, J. Zhou, J. Wu, Y. Fan, Y. Wang, H.E. Stanley, The science of science: From the perspective of complex systems, Phys. Rep. 714 (2017) 1–73.
[17] D.R. Amancio, O.N. Oliveira Jr, L.d.F. Costa, On the use of topological features and hierarchical characterization for disambiguating names in collaborative networks, Europhys. Lett. 99 (4) (2012) 48002.
[18] F.-X. Ren, H.-W. Shen, X.-Q. Cheng, Modeling the clustering in citation networks, Physica A 391 (12) (2012) 3533–3539.
[19] F. Breve, L. Zhao, Fuzzy community structure detection by particle competition and cooperation, Soft Comput. 17 (4) (2013) 659–673.
[20] F. Breve, Interactive image segmentation using label propagation through complex networks, Expert Syst. Appl. 123 (2019) 18–33.
[21] F. Breve, Building networks for image segmentation using particle competition and cooperation, in: International Conference on Computational Science and Its Applications, Springer, 2017, pp. 217–231.
[22] A.L. Barbieri, G. De Arruda, F.A. Rodrigues, O.M. Bruno, L.F. Costa, An entropy-based approach to automatic image segmentation of satellite images, Physica A 390 (3) (2011) 512–518.
[23] R.F. Cancho, R.V. Sole, Least effort and the origins of scaling in human language, Proc. Natl. Acad. Sci. USA 100 (3) (2003) 788–791.
[24] E. Estevez-Rams, A. Mesa-Rodriguez, D. Estevez-Moya, Complexity-entropy analysis at different levels of organisation in written language, PLoS One 14 (5) (2019) e0214863.
[25] M.A. Montemurro, D.H. Zanette, Keywords and co-occurrence patterns in the voynich manuscript: An information-theoretic analysis, PloS ONE 8 (6) (2013) e66344.
[26] S. Hassan, R. Mihalcea, C. Banea, Random walk term weighting for improved text classification, Int. J. Semant. Comput. 1 (04) (2007) 421–439.
[27] E.A. Correa Jr, A.A. Lopes, D.R. Amancio, Word sense disambiguation: A complex network approach, Inform. Sci. 442 (2018) 103–113.
[28] M. Stella, S. de Nigris, A. Aloric, C.S.Q. Siew, Forma mentis networks quantify crucial differences in STEM perception between students and experts, PLoS ONE 14 (10) (2019) e0222870.
[29] M. Stella, M. Brede, Patterns in the english language: Phonological networks, percolation and assembly models, J. Stat. Mech. Theory Exp. 2015 (5) (2015) P05006.
[30] T. Stanisz, J. Kwapie, S. Drod, Linguistic data mining with complex networks: A stylometric-oriented approach, Inform. Sci. 482 (2019) 301–320.

[31] H. Chen, X. Chen, H. Liu, How does language change as a lexical network? An investigation based on written Chinese word co-occurrence networks, PLoS ONE 13 (2) (2018) e0192545.
[32] Y. Gao, W. Liang, Y. Shi, Q. Huang, Comparison of directed and weighted co-occurrence networks of six languages, Physica A 393 (2014) 579–589.
[33] M. Garg, M. Kumar, The structure of word co-occurrence network for microblogs, Physica A 512 (2018) 698–720.
[34] D.R. Amancio, F.N. Silva, L.F. Costa, Concentric network symmetry grasps authors' styles in word adjacency networks, Europhys. Lett. 110 (6) (2015) 68001.
[35] V.Q. Marinho, G. Hirst, D.R. Amancio, Labelled network subgraphs reveal stylistic subtleties in written texts, J. Complex Netw. 6 (4) (2018) 620–638.
[36] D.R. Amancio, Probing the topological properties of complex networks modeling short written texts, PLoS ONE 10 (2) (2015) e0118394.
[37] O. Levy, Y. Goldberg, I. Dagan, Improving distributional similarity with lessons learned from word embeddings, Trans. Assoc. Comput. Linguist. 3 (2015) 211–225.
[38] S. Rothe, H. Schütze, Autoextend: Extending word embeddings to embeddings for synsets and lexemes, 2015, arXiv preprint arXiv:1507.01127.
[39] E. Nalisnick, B. Mitra, N. Craswell, R. Caruana, Improving document ranking with dual word embeddings, in: Proceedings of the 25th International Conference Companion on World Wide Web, International World Wide Web Conferences Steering Committee, 2016, pp. 83–84.
[40] I. Iacobacci, M.T. Pilehvar, R. Navigli, Sensembed: Learning sense embeddings for word and relational similarity, in: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), 2015, pp. 95–105.
[41] T. Kenter, M. De Rijke, Short text similarity with word embeddings, in: Proceedings of the 24th ACM International on Conference on Information and Knowledge Management, 2015, pp. 1411–1420.
[42] G.A. Miller, WordNet: An electronic lexical database, MIT press, 1998.
[43] Z. Luo, J. He, J. Qian, Y. Wang, J. Chen, W. Lu, Can Scientific Publication's Network Structural Features Predict its Citation?, in: Proceedings of the ACM/IEEE Joint Conference on Digital Libraries in 2020, 2020, pp. 485–486.
[44] T. Mikolov, G. Corrado, K. Chen, J. Dean, Efficient estimation of word representations in vector space, arXiv abs/1301.3781 (2013).
[45] J. Pennington, R. Socher, C.D. Manning, Glove: Global vectors for word representation, in: Empirical Methods in Natural Language Processing (EMNLP), 2014, pp. 1532–1543.
[46] P. Bojanowski, E. Grave, A. Joulin, T. Mikolov, Enriching word vectors with subword information, Trans. Assoc. Comput. Linguist. (ISSN: 2307-387X) 5 (2017) 135–146.
[47] H.F. Arruda, L.F. Costa, D.R. Amancio, Using complex networks for text classification: Discriminating informative and imaginative documents, EPL (Europhys. Lett.) 113 (2) (2016) 28007.
[48] D.R. Amancio, S.M. Aluisio, O.N. Oliveira Jr, L.F. Costa, Complex networks analysis of language complexity, Europhys. Lett. 100 (5) (2012) 58002.
[49] M.Á. Serrano, M. Boguná, A. Vespignani, Extracting the multiscale backbone of complex weighted networks, Proc. Natl. Acad. Sci. 106 (16) (2009) 6483–6488.
[50] B.A.N. Travençolo, L. Costa, Accessibility in complex networks, Phys. Lett. A 373 (1) (2008) 89–95.
[51] H. Liu, The complexity of chinese syntactic dependency networks, Physica A 387 (12) (2008) 3048–3058.
[52] H. Liu, W. Li, Language clusters based on linguistic complex networks, Chin. Sci. Bull. 55 (30) (2010) 3458–3465.
[53] S. Yu, H. Liu, C. Xu, Statistical properties of chinese phonemic networks, Physica A 390 (7) (2011) 1370–1380.
[54] M.Z. Rodriguez, C.H. Comin, D. Casanova, O.M. Bruno, D.R. Amancio, L.F. Costa, F.A. Rodrigues, Clustering algorithms: A comparative approach, PLoS ONE 14 (1) (2019) e0210236.
[55] E. Frank, M. Hall, L. Trigg, G. Holmes, I.H. Witten, Data mining in bioinformatics using weka, Bioinformatics 20 (15) (2004) 2479–2481.
[56] A. Kulig, J. Kwapień, T. Stanisz, S. Drożdż, In narrative texts punctuation marks obey the same statistics as words, Inform. Sci. 375 (2017) 98–113.
[57] V.Q. Marinho, G. Hirst, D.R. Amancio, Authorship attribution via network motifs identification, in: 2016 5th Brazilian Conference on Intelligent Systems (BRACIS), IEEE, 2016, pp. 355–360.
[58] C. Basile, D. Benedetto, E. Caglioti, M.D. Esposti, An example of mathematical authorship attribution, J. Math. Phys. 49 (12) (2008) 125211.
[59] J.R.F. Ronqui, G. Travieso, Analyzing complex networks through correlations in centrality measurements, J. Stat. Mech. Theory Exp. 2015 (5) (2015) P05030.
[60] D.R. Amancio, E.G. Altmann, D. Rybski, O.N. Oliveira Jr, L.F. Costa, Probing the statistical properties of unknown texts: Application to the voynich manuscript, PLoS One 8 (7) (2013) e67310.
[61] J.S. Bergstra, R. Bardenet, Y. Bengio, B. Kégl, Algorithms for hyper-parameter optimization, in: Advances in Neural Information Processing Systems, 2011, pp. 2546–2554.
[62] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, 2018, arXiv preprint arXiv:1810.04805.
[63] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. Salakhutdinov, Q.V. Le, Xlnet: Generalized autoregressive pretraining for language understanding, 2019, arXiv preprint arXiv:1906.08237.
[64] D. Liben-Nowell, J. Kleinberg, The link-prediction problem for social networks, J. Am. Soc. Inf. Sci. Technol. 58 (7) (2007) 1019–1031.