Repositório ISCTE-IUL

Deposited in *Repositório ISCTE-IUL*:
2019-11-19

Deposited version:
Post-print

Peer-review status of attached file:
Peer-reviewed

Citation for published item:
Maia, Ferreira, J. & Martins, A. (2019). Rating prediction on yelp academic dataset using paragraph vectors. In Proceedings of 232nd The IIER International Conference. (pp. 56-62).: IIER.

Further information on publisher's website:
http://worldresearchlibrary.org/proceeding.php?pid=2815

RATING PREDICTION ON YELP ACADEMIC DATASET USING PARAGRAPH VECTORS

[1]RUI MAIA, [2]JOAO C. FERREIRA,
[3]ANA LUCIA MARTINS

[1]Inov Inesc Inovação – Instituto de Novas Tecnologias and Instituto Superior Tecnico, Portugal
[1,2,3]Instituto Universitário de Lisboa (ISCTE-IUL), Portugal
[2]Information Sciences, Technologies and Architecture Research Center (ISTAR-IUL), Portugal
[2]Business Research Centre (BRU-IUL)
E-mail: [1]rui.maia@inov.pt, [2]jcafa@iscte-iul.pt, [3]almartins@iscte-iul.pt

Abstract - This work studies the application of Paragraph Vectors to the Yelp Academic Dataset reviews in order to predict user ratings for different categories of businesses like auto repair, restaurants or veterinarians. Paragraph Vectors is a word embeddings techniques were each word or piece of text is converted to a continuous low dimensional space. Then, the opinion mining or senti-ment analysis is observed as a classification task, where each user review is associated with a label - the rating - and a probabilistic model is built with a logistic classifier. Following the intuition that the semantic information pre-sent in textual user reviews is generally more complex and complete than the numeric rating itself, this work applies Paragraph Vectors successfully toYelp dataset and evaluates its results.

Index terms - Prediction, Paragraph Vectors, Learning-to-Rank, dimension reduce.

INTRODUCTION

With the fast growing on-line content, made available by large Companies like Amazon or NetFlix, service providers are interested in recommendation system services that can maximize the probability of a user to consume or buy a product or service. Users are also interested in systems that help to virtually shrink the space of options when, like in big web stores or multimedia websites, there are many thousands or millions of available options. Being developed in the last thirty years, recommendation systems have become an important part of the in-telligence areas of big content providers. Yelp is a platform created in San Fran-cisco in July 2004 that delivers reviews and ratings on local businesses of thirty two different countries. With approximately 95 million reviews on multiple busi-nesses categories such as mechanics, restaurants or dentists, about 86 million unique visitors via mobile devices, approximately 75 millions unique visitors via desktop, Yelp platform uses Artificial Intelligence automated software to recom-mend personalized suggestions based on reliable reviews for each visitor. The recommendation is made using the data set content, including for example imag-es, review texts, business rating, or business location while taking only three-quarters of the available reviews into account for the model training. One of the latest and successful approaches in the recommendation systems area is the Fac-torization Machines hybrid approach, initially proposed by Steffen Rendle [1]. It models the relations between users and items as an aggregation of different fea-tures for the purpose of generating recommendations. This technique is based on matrix factorization approach and feature engineering [2]. Factorization Ma-chines (FMs) can combine the high prediction accuracy of factorization models with the flexibility of feature engineering, nowadays being commonly employed in the development of context-based recommendation systems [3]. A matrix fac-torization based recommendation system usually relies on discrete values that users express about objects they were related to. As such, these systems also rely on complex feature engineering processes. Multiple Natural Language Processing and Machine Learning techniques have also been applied successfully for senti-ment analysis, with different levels of detail as Positive or Negative for coarse-grained evaluation and Very Negative, Negative, Neutral, Positive, a Very Posi-tive, for fine-grained evaluation). Rating prediction can also be seen as a classification problem as far as a discrete number of ratings, for example, 1 to 5 or Like and Dislike, can be observed as different categories. Word embeddings are language modelling technique for words representation as vectors. Some of the latest developments in the word embeddings area were introduced by Mikolov et al. [4] in 2013. The authors presented two new architectures for un-supervised learning algorithms for fixed length vectors that efficiently compute high-quality word vectors even when considering large datasets with billions of words in the dictionary [4]. Later, Quoc Le and Tomas Mikolov [5] extended this work to the representation of variable length sentences, paragraphs or documents which they called Paragraph Vectors. Regarding sentiment analysis as a classification or rating prediction task, this work evaluates the use of the Para-graph Vector algorithm against the Yelp Academic Dataset reviews in order to address one of the essential Recommendation System challenges: the rating pre-diction. Opinion mining and sentiment analysis are interdisciplinary fields of study having contributions from multiple knowledge areas, from linguistics to machine learning or even psychology. The initial syntax-based approaches are being surpassed by more complex ones, as the ones based on machine learning knowledge and sometimes using large context dependent databases, or practical resources, for example. If it is possible to describe ,a human sentiment ex-pressed in natural language using detailed semantic representation it might be possible to enhance the efficiency of rating prediction systems or recommenda-tion systems. The remaining contents of this paper are organized as follows. Sec-tion 2 describes previous fundamental concepts about recommendation and natu-ral language processing, namely on named entities extraction and classification. Section 3 describes the more relevant academic studies that relate recommenda-tion systems and natural language processing, while Section 4 presents the refer-ence implementations and the main results obtained. Section 5 discusses differ-ent modelling choices and approaches that can be valid in the presented context.

II. FUNDAMENTAL CONCEPTS

Mining people’s opinion presents scientific challenges due to the multiple fields of study involved. It requires syntactical and semantical knowledge on the processed language but also on machine learning models, for example. Regarding Natural Language Processing perspective, opinion mining can be represented as a restricted problem of identifying positive and negative sentiments about entities or situations [6]. In order to formally understand what the object of someone’s review is and how it is being classified, it is relevant to understand Part-of-Speech tagging and Polarity Classification concepts. Part-of-Speech (POS) tagging is usually an essential part of the text processing, for classifying and disambiguating names, verbs, adjectives or other types of language structure parts. The detection of pre-specified POS patterns, not only adjectives, can be a relevant indicator of the sentiment or opinion being expressed by someone [6]. Not less frequently, individual punctuation marks and even symbols, emphasize a specific idea, sometimes resembling spoken dialogues (ex: "Cool movie!!!!") and claiming a reasonable interpretation and additional formalization and processing phases. Polarity classification or sentiment polarity classification is a language processing task that classifies a piece of text as being positive or negative. Some approaches calculate the polarity indicator as a value inside a degree of positivity, not binary, but somewhere between a positive and a negative limit, as Awful, Negative, Positive and Awesome. When user reviews contain opinions on more than one item or when an opinion is not clearly positive or negative the processing approach has to deal with subjectivity detection. Correction identifies the multiple tuples of opinion and its topic. Opinion mining can be seen as extraction of a formal representation of the most relevant features presented in a text. Some of the most typical addressed textual features are the Term Presence, Position or Frequency, n-Grams and Skip-Grams. Presence is represented as a vector of binary values - 1 or 0 – indicating the presence or absence of relevant domain terms in the analyzed text. Nonetheless, a higher frequency of a term in a text does not mean necessarily that it is being evaluated as positive or negative, but it might indicate that the term is a relevant topic. Position feature refers to the location of the term in the analyzed text, which might impact the sentiment exposed by a user review. Bi-grams, Tri-grams Presence or Skip-Grams are useful, considered features in an opinion mining process as they describe how the terms relate with each other. Rating inference or Ordinal regression is the process of predicting a rating value given by a reviewer on an item. This work focuses mainly on the application of a specific word embeddings technique - Paragraph Vectors - although it has to deal with simple textual preprocessing tasks. Most of the research on opinion mining was done for the English written language which helped the development of English sentiment lexicon and corpora resources. This implies that specific processing steps must be taken into account when doing opinion mining over different languages. The main approaches to the opinion mining problem can be listed as follows:

Keyword Spotting This approach classifies texts or opinions based on the detection of unambiguous sentiment expressions as "happy" or "bored". It is a simple and widely used technique. This approach can be refined with the detection of auxiliary refinement terms, as intensity modifiers like "extremely" or "somehow", and cue phrases like "wanted to" or "pretend that". However, keyword spotting technique appears to be insufficient at accurately identify the inversion of a sentiment expression like "today wasn’t a happy day at all" or at unravelling underlying sentiments not exposed by adjectives like "I have no words to describe what I felt at the wedding ceremony.". Some more relevant resources on the English language useful annotation are Clark Elliott’s Affective Reasoner [7], Andrew Ortony et al. Affective Lexicon [8] and Janyce Wiebe et al. Linguistic Annotation Scheme [9].

Lexical Affinity It represents an advance when compared to Keyword Spotting, by modelling the relation between common words and sentiment. As an example, it can be seen that "collision" has a high level of probability to be related with negative sentiment about an event, and that relation is thereby modelled using annotated corpora. On the other hand, regarding negated expressions or underlying meanings, this approach still does not behave inefficient terms because it maintains the process at the word level. Moreover, the relation between common terms and sentiments are seen as being context dependent, which raises severe issues when developing reusable and context-independent solutions.

Statistical A machine learning algorithm is fed with a large corpus of expertly annotated text. This approach classifies texts and detects emotions using Support Vector Machines (SVM), Bayesian Inference or Neural Networks, by identifying the affect keywords, the related common words that change the intensity or direction of expression and the appropriate punctuation and co-occurrence frequency. Because it is based on a statistical model, it has low semantic information (aside from the affect keywords), which gets the best results when classifying pages or paragraphs of texts, rather than short or few sentences. Concept-Based This approach relies on semantic networks and web ontologies to classify affective information expressed on texts. It can identify direct expressions in the text but also subtle sentiments expressed in multi-word expressions or even in articulated sentences and concepts. This technique depends in-depth and breadth knowledge preexistent resources. In other terms, the inference capability of the system is directly proportional to the richness or completeness of the knowledge database.

III. PROPOSAL

Opinions presented in reviews are commonly not restricted to one item. Instead, they refer to multiple levels or components of the item. For example, a restaurant review commonly includes the user opinion on a specific recipe, the waiter's sympathy or even the location of the restaurant that serves the recipe, and a webcam review might refer to the design, the image quality, the size or the cost, for example. Thus, each review can include detailed opinions on many facts that affected the user's global sentiment.

Hu et al. [10] introduced a feature-based opinion mining technique where each specific feature in a review is identified and the user opinion on that feature clas-sified as positive or negative. The focus of the approach is only on the features that the users had commented, not the item itself. The authors proposed a two-phase approach: start with a POS tagging process that identifies the main features and opinion parts, then classify each tuple feature-opinion as positive or nega-tive. Following a similar approach, Freitas et al. [11] proposed an ontology-based process that also tries to identify all the known features referred in a user review, and then calculate the global review polarity. The authors work focused Movie and Hotel domains for Brasilian Portuguese texts. They used three main sets of resources: the TreeTagger [www.cis.uni-muenchen.de/schmid/tools/TreeTagger] for Part-of-Speech processing, the OpLexicon [on-tolp.inf.pucrs.br/Recursos/downloads-OpLexicon.php] for polarity identification and the Hontology [ontolp.inf.pucrs.br/Recursos/downloads-Hontology.php] and Movie Ontology [www.movieontology.org] as for domain ontologies. Both Hu et al.[12] and Freitas et al. [11] approaches can be generally illustrated in Figure 1, adapted from [11]. The high interconnection between emotion analysis and polarity detection mo-tivated Cambria et al. to propose a new approach Sentic Computing- that merges Artificial Intelligence, Linguistics and Psychology [13]. This approach explores the knowledge about linguistics and statistical methods. The process flow largely depends on SenticNet [14], a semantic and affective labelled resource where 30.000 single or multi-word expressions are classified. The author underlined that the proposed approach firstly uses linguistics and affective knowledge to represent emotions and emotional flows in human interactions, while machine learning algorithms are used as backup methods when there is no previous knowledge or exact representation on a specific object. The approach result is a polarity score calculated within the range [−1;1]. On Affective Computing and Sentiment Analysis [15], Erik Cambria describes how the such a hybrid ap-proach might be successfully applied to multiple contexts, from marketing and strategy evaluation to public or private intelligence and decision support sys-tems.

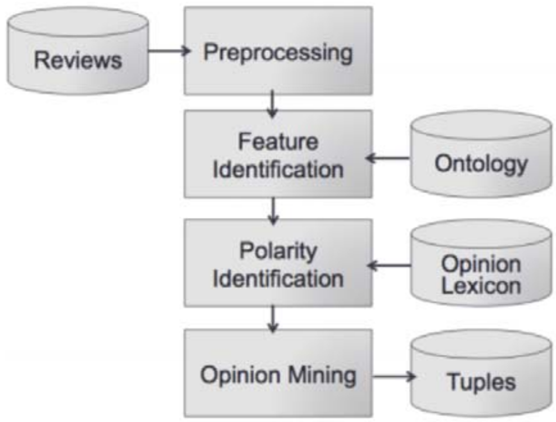

Figure 1:Approach implemented

Zhang et al. considered that some features that affect user's opinion are not always directly related to the product or service being analyzed [16]. The authors considered that these features - the implicit features - might play a relevant part in global user opinion and can be extracted from review texts. Their matrix-based algorithm leverages co-occurrence and association rules to uncover hidden fea-tures. Consider the example "No electricity after a few phone calls" in a user re-view about a mobile phone. The implicit feature Electricity could identify a rele-vant phone part - the battery - that was hidden in the implicit Electricity. Some authors as Riloff et al. [17] try to describe a user's opinion based on a sentence level text analysis, calculating the polarity at a sentence level. This level of ap-proach shows to be insufficient to address situations where the users express opinions in more than one target in the same sentence: "I like the restaurant, but the chips were terrible!". The opinion on the restaurant is positive and different from the negative opinion on the chips. This kind of mining can draw a structured and concise opinion representation [12], [18].

Word Vectors

In [4] Mikolov et al. proposed two new efficient architectures for words representation from large datasets on a continuous vector space model. The authors stated that the similarity of word vector representations goes beyond the syntactic regularities as it can also describe semantic relations. Therefore, a word vector model delivers the possibility to infer knowledge using algebraic operations, like the one that can be observed in the operation vector(King) + vector(Woman) − vector(Man) which would result in a vector similar to vector(Queen). With the newly proposed architectures the author's tried to reduce the extensive computational resources involved in neural network training tasks, while improving systems accuracy using computing high dimensional word vectors from large datasets (they used Google News corpus, which contains 6 billion of tokens). Mikolov et al. started with a Neural Network Language Model (NNLM) architecture, a multiple layer neural network. They worked on a simple method for training the NNLM in two steps: learn a word vector representation using a simple model and then train the n-gram NNLM. Their work resulted in two different models: the Continuous Bag-of-Words (CBOW) and the Continuous SkipGram. The Continuous Bag-of-Words Model represents words independently of their order in the text. The authors found that the best performance in word vectors representation task was achieved when considering a window of 1 + 8 words centred on the considered word. Figure 2 describes the model with a window of size 1 + 4.

The Skip-Gram model is similar to the CBOW, but it uses each word as input to a log-linear classifier in order to predict a range of words before and after the word (vector) used as input. The model tries to learn vector representations that can efficiently support the prediction of surrounding words, i.e., words that are inside the analysis window and appear before or after the considered word. Figure 2 shows an example with a window size of 5. Considering a bigger window will result in a more accurate word vector representation but also will raise computational complexity. To authors propose a window of the size of 1 + 10 which, in a single machine implementation can train 100 billion words in one day. In [19] the authors continue their work on Skip-Gram with further developments on the training algorithms, namely by introducing Hierarchical Softmax and Negative Sampling. Subsampling of frequent words was introduced as a mean to improve the vectors accuracy and provide a faster training model, since words that occur very frequently give less information that the less frequents.

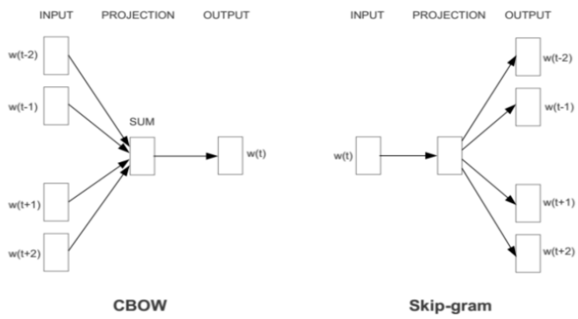

Figure1. Two different models for word vector representation proposed by Mikolov et al. The word (t) represented in position t considering a window of size 1+ 4 in CBOW model and the words in the window deduced from the word w(t) in Skip-Gram.

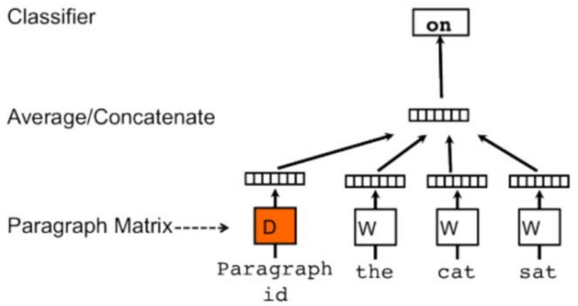

Figure2. Using the Paragraph Vector Distributed Memory modelin order to predict a fourth word given a paragraph vector and three-word vectors

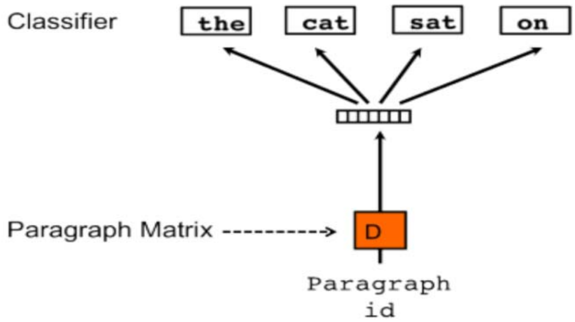

Figure3. Using the Paragraph Vector Distributed Bag-of-wordsmodel in order to predict four words given only a paragraph vector.

Paragraph Vectors

Quoc Le and Tomas Mikolov [9] extended the Word Vectors work of Mikolov et al. for sentence and paragraphed continuously distributed vector representa-tion. The authors claimed that their work support general and efficient vector modelling for pieces of text of any length as it does not rely on text parsing either on domain-specific word dataset configuration and weighting. Le and Mikolov proposed two different approaches: a Paragraph Vector Distributed Memory (PV-DM) model and a Paragraph Vector Distributed Bag-of-Words (PV-DBOW).The Distributed Memory (PV-DM) approach introduces a paragraph vector that rep-resents unequivocally a set of words in a specific order as a column in a matrix D. This vector is concatenated with word vectors represented as columns in a matrix W in order to find a final vector representation that can predict the next expected word in the considered phrase, sentence or paragraph context. Thereby, this approach tries to introduce the information that is missing in the given win-dow of analysis by adding paragraph information or memory (see Figure 3). The algorithm has two phases. First, the calculation of word vectors and paragraph vectors. The second step is the inference of new (unseen) paragraph vectors. A fixed length window is sampled over each paragraph or piece of text. Each of the paragraph and word vectors is trained using Stochastic Gradient Descend, having the gradient calculated by backpropagation. This process doesn't rely on Figure 4: Using the Paragraph Vector Distributed Bag-of-words model in order to predict four words given only a paragraph vector. on any text parsing or even labelling. The second approach proposed by Le and Mikolov in [9] is the Paragraph Vec-tors Distributed Bag-of-Words (PV-DBOW). It ignores the context words at the input moment and relies only on a paragraph vector to predict words in a small window (see Figure 4for a window of 4 words). This model is lightest than the PV-DM since it doesn't need to calculate or use the word vectors. The authors referred that more consistent results were achieved when the paragraph represen-tation was made by a combination of a PV-DM vector with a PV-DBOW vector. Thereby they suggest (strongly recommend) that this approach is preferable when compared with the single use of PV-DM or PV-DBOW.

IV. SETUP

In order to evaluate the Paragraph Vectors approach for a sentiment analyzes problem I followed the previous work of Le and Mikolov [5]. The authors have not released any code implementation of their work, although, other authors pro-vide libraries that implement word and paragraph representation as described by Le and Mikolov. For this work, I chose a python implementation of Paragraph Vectors [20] that have reasonable support and examples in order to reproduce the reference paper results. By reproducing the baseline [5] results, I expected to get a stable setup in order to test the same approach in another dataset, Yelp. I concentrated in the IMDB dataset experiment, which is the one that is most doc-umented and reviewed. Yelp, like IMDB, contains single sentence reviews and paragraph reviews. They are both in the English language, but IMDB concentrates only in movie reviews, while Yelp has multiple business types reviews.

4.**Baseline**: Sentiment Analysis of IMDB reviews using Paragraph Vectors

The IMDB dataset is a large movie review resource made available by Maas et al. [21]. It includes 100.000 movie reviews got from IMDB organized in three sets: 25.000 labelled training reviews, 25.000 labelled test reviews and 50.000 unlabeled develop reviews. The sentiment associated with each review is repre-sented with a label, Positive or Negative, having 25.000 Positive reviews (12.500 for training and 12.500 for testing) and 25.000 Negative reviews (again, 12.500 for training and 12.500 for testing). There is no reference on how the text prepro-cessing was made. Considering that the dataset was already preprocessed, this step is resumed to the lower-case text conversion and the disconnection of the punctuation symbols (0:0;0 "0;0 ;0 ;0 (0;0)0;0 !0;0 ?0;0 ;0 ;0 :0) from words, inserting a white space before and after the symbol. This preprocessing also in-cludes replacing the special symbol < =BR > with white space. This simple meth-od is followed by Tomas Mikolov on his own reproduction of the PV-DBOW approach, therefore, considered here as a minimum acceptable method. The word and paragraph vectors were learned by using the training set (25.000) and devel-opment set (50.000) reviews using Stochastic Gradient Descent (SGD) and back-propagation. To get the concatenated final paragraph vectors PV-DM and PV-DBOW, models were trained with a vector of 400 dimensions using a window of 10 words

Sentiment Analysis of Yelp using Paragraph Vectors

Yelp academic dataset is available by request and contains 891.250 reviews on multiple business types, including restaurants, auto repair or veterinarians, for example. The reviews can include multiple sentences (as in IMDB), and the re-view label consists in a value between 1 to 5 (see Figure 5). The reviews were preprocessed using the same routine as the one used with IMDB. For a closer comparison to the reference work, the yelp dataset ratings were converted to Pos-itive or Negative ones by considering Negative the reviews with ratings 1, 2 or 3, and Positive the reviews with ratings 4 or 5 (see Figure 6). Tests were also run considering Negative the reviews with ratings 1 or 2, and Positive the reviews with ratings 3, 4 or 5 (see Figure 6). This difference did not affect the global re-sults. In the end, the Positive and Negative Yelp rating distribution was unbal-anced when compared with the even distribution of IMDB reviews dataset.

V. RESULTS

The results obtained by applying the approach proposed by Le and Mikolov [5] are shown in Table 1. Fixing a verifiable baseline enables one to evaluate the possibility of applying the same approach to other domains, using the same or different hyperparameters. It is also desirable to leverage the understanding of each hyperparameter behaviour. Different paragraph vectors dimensions were tested (100 and 400) which seemed to have a limited positive impact on the re-sults. I also tested the of just one of the different models (PV-DM and PV-DBOW) in the paragraph vectors construction, and not the concatenation of the paragraphs got from the two models.

Figure4. Stars distribution on first 100.000 reviews of Yelp Academic Dataset.

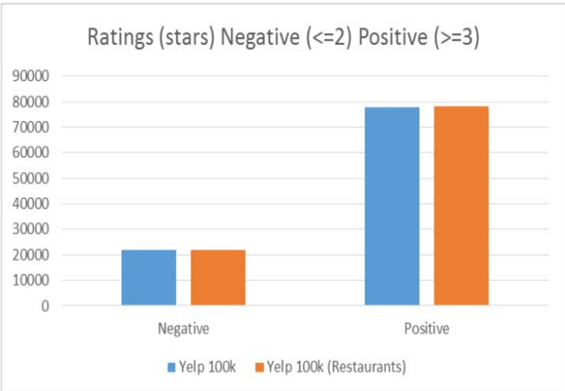

Figure 6. Stars distribution on first 100.000 reviews of Yelp Academic Dataset after the conversion to Positive or Negative rating

Table 1. Results using PV-DBOW concatenated with PVDM(concatenated or mean), with 100-dimensional vectors, negative sample of 5 noise words, and word minimum count of 2 (words that appear on one single review are discarded)

	dbow+dmm,d100, n5,mc2,t4	dbow+dmc,d100, n5,mc2,t4
IMDB	0.102720	0.104200
Yelp 100K		0.091640

Table 2. Results using PV-DBOW with 100 or 400-dimensional vectors, with or without Hierarchical Softmax (hs) and word minimum count of 2 (words that appear on one single review are discarded)

	dbow,d100,n5, mc2,t4	dbow,d400,n5 ,t4	dbow,d100,n5, hs,t4
IMDB	0.103640	0.106520	
Yelp 100K	0.092920	0.090240	
Yelp 100K Neg3			0.137900

Table 3. Results using PV-DM - concatenated vectors- with 100-dimensional vectors, with or without Hierarchical Softmax (hs) and with or without word minimum count of 2 (mc2). Words that appear on one single review are discarded). The window is set to 5 words or 10 words and a sample to be used in the higher-frequency words downsampling.

	dm/c,d100,n5, w5,mc2,t4	dm/c,d100,n5,hs, w10,s0.001,t4	dm/c,d100,n5,w10, mc2,s0.001,t4
IMDB	0.183320		
Yelp 100K		0.133180	0.132480

Table 4. Results using PV-DM - mean vector - with 100 or 400-dimensional vectors, with or without Hierarchical Softmax (hs) and with or without word minimum count of 2 (mc2). Words that appear on one single review are discarded). The window is set to 5 words or 10 words and a sample to be used in the higher-frequency words downsampling.

	dm/m,d100,n5, w10,mc2,t4	dm/m,d400,n5,hs, w10,s0.001,t4
IMDB	0.135360	
Yelp 100K		0.116700

I. CONCLUSIONS

Word and paragraph vectors can be very useful on a considerable number of NLP applications. Although, when referring to Paragraph Vectors, it is not clear how word semantic similarity can be calculated when referring to informal knowledge or when the words have different use depending on the domain. Even though, it is easy to consider that F rance is similar to Spain when considering the relation between countries and continents. The Paragraph Vectors approach relies on the choice of a training algorithm and the definition of a set of optimal hyperparameters, all being a domain or problem dependent. As referred by Mikolov et al. in [13], some crucial (and unclear) decisions must be made about the adopted training model, vectors size, subsampling rate and word window size. It is difficult to analyze these key choices that affect the obtained results when authors present hyperparameter values without any correct test result. For example, Le and Mikolov suggest the use of a window size of 10 words [9] arguing that it is the optimal size, although the tests run on this work do not confirm that. Moreover, PV-DM model needs to hold the vector representation of all possible known words, which might imply a real-world issue when dealing with unseen paragraphs which might contain new unseen words. In the same way, building the paragraph vector as the concatenation of two paragraph vectors PV-DM and PV-DBOW or the better PV-DM results against PV-DBOW, is not confirmed by multiple experiments. On the contrary, it seems that the results of PV-DBOWare generally better and only improves a little with the concatenation of the PV-DM, as can be seen on the results Table ??. This might indicate high variability in what concerns to the correct use of language syntax.

REFERENCES

[1] S. Rendle. Factorization machines. In Proceedings of the IEEE International Conference on Data Mining, 2010.

[2] Steffen Rendle. 2012. Factorization Machines with libFM. ACM Trans. Intell. Syst. Technol. 3, 3, Article 57 (May 2012), 22 pages.DOI=http://dx.doi.org/10.1145/2168752.2168771

[3] S. Rendle, Z. Gantner, C. Freudenthaler, and L. Schmidt-Thieme. Fast context-aware recommendations with factorization machines. In Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval, 2011.

[4] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781, 2013.

[5] Q. V. Le and T. Mikolov. Distributed representations of sentences and documents.

[6] E. Cambria, B. Schuller, Y. Xia, and C. Havasi. New avenues in opinion mining and sentiment analysis. IEEE Intelligent Systems, (2):15–21, 2013.

[7] C. D. Elliott. The Affective Reasoner: A Process Model of Emotions in a Multi-agent System. PhD thesis, Evanston, IL, USA, 1992. UMI Order No. GAX92-29901.

[8] A. Ortony, G. L. Clore, and A. Collins. The cognitive structure of emotions. Cambridge university press, 1990.

[9] J. Wiebe, T. Wilson, and C. Cardie. Annotating expressions of opinions and emotions in language. Language resources and evaluation, 39(2-3):165–210, 2005.

[10] M. Hu and B. Liu. Mining opinion features in customer reviews. In AAAI, volume 4, pages 755–760, 2004

[11] L. A. Freitas and R. Vieira. Ontology based feature level opinion mining for portuguese reviews. In Proceedings of the 22nd international conference on World Wide Web companion, pages 367–370. International World Wide Web Conferences Steering Committee, 2013.

[12] E. Cambria. Affective computing and sentiment analysis. IEEE Intelligent Systems, 31(2):102–107, 2016.

[13] E. Cambria, D. Olsher, and D. Rajagopal. Senticnet 3: a common and common-sense knowledge base for cognition-driven sentiment analysis. In Twenty-eighth AAAI conference on artificial intelligence, 2014.

[14]]E. Cambria and A. Hussain. Sentic computing: a common-sense-based framework for concept-level sentiment analysis, volume 1. Springer, 2015.

[15] Y. Zhang and W. Zhu. Extracting implicit features in online customer reviews for opinion mining. In Proceedings of the 22nd international conference on World Wide Web companion, pages 103–104. International World Wide Web Conferences Steering Committee, 2013

[16] E. Riloff and J. Wiebe. Learning extraction patterns for subjective expressions. In Proceedings of the 2003 conference on Empirical methods in natural language processing, pages 105–112. Association for Computational Linguistics, 2003.

[17] G. D. Fabbrizio, A. Aker, and R. Gaizauskas. Starlet: multi-document summarization of service and product reviews with balanced rating distributions. In Data Mining Workshops (ICDMW), 2011 IEEE 11th International Conference on, pages 67–74. IEEE, 2011.

[18] B. Lu, M. Ott, C. Cardie, and B. K. Tsou. Multi-aspect sentiment analysis with topic models. In Data Mining Workshops (ICDMW), 2011 IEEE 11th International Conference on, pages 81–88. IEEE, 2011.

[19] T. Mikolov and J. Dean. Distributed representations of words and phrases and their compositionality. Advances in neural information processing systems, 2013

[20] R. Reh˚u ˇ ˇ rek and P. Sojka. Software framework for topic modelling with large corpora. In Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks, pages 45–50, Valletta, Malta, May 2010. ELRA.http://is.muni.cz/publication/884893/en.

[21] A. L. Maas, R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng, and C. Potts. Learning word vectors for sentiment analysis. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, pages 142–150, Portland, Oregon, USA, June 2011. Association for Computational Linguistics

★★★