SYSTEMIC: Information System and Informatics Journal

ISSN: 2460-8092, 2548-6551 (e)

Vol 7 No 2 - Desember 2021

A Survey of Social Network - Word Embedding Approach for Hate Speeches Detection

Bayu Adhi Nugroho

Inforrmation System Department, UIN Sunan Ampel Surabaya

bayu@uinsby.ac.id

Kata Kunci

Word embedding, ujaran kebencian, deteksi online.

Abstrak

Word embedding merupakan teknik representasi kata atau kalimat dalam ruang vektor. Representasi tersebut ditujukan untuk membangun sebuah model yang sesuai untuk tugas khusus terkait penggunaan kata atau kalimat tersebut, contohnya, sebuah model kemiripan antar kata atau kalimat, sebuah model dari hubungan antar pengguna Twitter. Penggunaan word embedding sangat bermanfaat dalam proses riset analisa sentimen karena membantu dalam pembentukan model matematika dari kalimat, selain itu word embedding juga bermanfaat untuk proses komputasi yang lebih lanjut.

Keywords

Word embedding, hate speeches, online detection.

Abstract

Word embedding is a technique to represent sentences in vector space. The representation itself is carried-out to build a model that would suffice in representing a particular task related to the use of the sentence itself, for example, a model of similarity among sentences/words, a model of Twitter user connectivity, and demographics of tweets model. The use of word embedding is a handful to the sentiment analysis research because it helps build a mathematical-friendly model from sentences. The model then will be suitable as feeds for the other computational process.

1. Background

Current research in sentiment detection that focuses on racism and hate speeches can be categorized into two categories, namely: (1) feature engineering with traditional classifiers using word embeddings architectures that classify where word embeddings are part of the feature set and (2) end-to-end deep learning architecture that classifies the text while learning word embedding as a byproduct.

Feature engineering includes capturing linguistic, textual, topical features, and semantic information. For example, Hasanuzzaman et al. [2], Mondal et al. [9], and Tulkens et al. [10] use linguistic-template and demographics discourses-dictionaries to define "racism" then utilize classic classifiers such as SVM to detect "racisms."

In the case of deep neural network approaches, we specifically focus on integrated approaches that do not rely on hand-crafted features and external classifiers, for example, Huang et al. [11], Kim (2014) [12] [13] and Vosoughi et al. (2016) use Neural Network models to generate word embeddings specifically for sentiment detection tasks.

2. Feature Engineering and Classifiers

The research by Hasanuzzaman et al. [2] uses demographic embodied with pre-trained word embedding. The dataset contains three months of Twitter messages dated from February 5, 2015, to May 5, 2015. Despite the use of slur databases [14] [15] as references, it was clearly stated in their publication [2] that the existence of slurs does not define the offensiveness of a tweet. Demographic features: age, gender, and location are incorporated into the embedding during training.

The data were annotated through the crowd flower crowdsourcing platform [16]. Each tweet was labeled by three raters who were asked the labeling of a tweet can be considered racist or not based on age, gender, and location of the tweet's owner. The label came in three choices: Yes (racist), No, and Unsure. Like Tulkens et al. [10], Hasanuzzaman et al. [2] rely on SVM for the classification model. The slight difference was that Hasanuzzaman et al. [2] use linear SVM with a different setup of input configurations implemented in Weka [17]. The input configuration varies from n = 1 to 4 grams and

DOI : 10.29080/systemic.v7i2.1771

word2vec, independently or combined with all demographic embeddings. Hasanuzzaman et al. [2] demonstrated that word2vec, combined with all demographic embeddings (age + gender + location), outperformed any other input model.

Mondal et al. [9] focus on measuring the large scale of hate speeches inside social media. The approach is an unsupervised method relying on sentence templates and Hatebase [18] ("world's largest online crowdsourced repository of structured, multilingual, usage-based hate words" [9]). The datasets come from two social media: Twitter and Whisper messaging platforms. In both of the two sources, the data are taken from June 2014 to June 2015. The aspects of hatred were based on: race, religion, disability, sexual orientation, ethnicity, and gender [19]. These aspects are used as baseline categories. The limitation is that the hate speeches that do not conform to the sentence template are not detectable. The primary research output is exploratory statistics of hate speeches based on the anonymity of speech across different countries.

The research by Tulkens et al. [10] approach was developing "discourses dictionaries" using Facebook's comments page as the primary input. Two pages of themes were selected; those were the anti-Islam theme and the right-wing theme. Comments were classified as racist or not by three annotators. Three types of dictionaries were built: the first was based on Dutch Linguistic and Inquiry Word Count (LIWC) [20], utilizing its word categories as a baseline for a discourse dictionary. The second dictionary was built from words classified as racist in the training data. The third dictionary was an expansion from each word in the second dictionary by adding the five closest words based on word2vec (Mikolov et al., 2013) [21] using the cosine similarity method. This dictionary-based approach produced an n-dimensional vector containing numbers that had been normalized and scaled. The numbers resulted from dividing words' frequencies in each category with the total of words in comments. The vector was used as input for nonlinear SVM with RBF kernel implemented in scikit-learn [22]. The overall performance of classifications concluded that both the original and the word2vec expansion discourse dictionaries from training data outperformed the result from the LIWC dictionary.

3. Feature Extraction and Deep Learning

Another version of the word embeddings approach was introduced by Huang et al. [11]. The research aimed to capture the semantics of words in the vector space model. The primary output from this research was an improvement of word embeddings which can predict the next word by using joint training of local and global context within a neural language model. The dataset chosen was the April 2010 Wikipedia corpus. Ten windows (5 words after and before) were used as local context with 50-dimensional embeddings and 100 hidden units. The IDF-weighting was used as a weighting function. Hence the weighting function captures the importance of a word within the document (global context). In order to maximize the semantic similarity, the researchers made use of Wordnet (Miller, 1995) [23] and benchmarked with WordSim-353 (Finkelstein et al.,2001) [24]. The final result of this research outperformed the result from similar research by Mnih and Hinton (2008) [25], both duplicating on Wikipedia dataset or using the research's [25] original dataset of one year Reuters English newswire.

Convolutional Neural Network was used for Sentence Classification by Kim (2014) [12]. The research used pre-trained word vectors; if a word were missing from pre-trained vectors, that word would be generated randomly. The research was a feature selection process over CNN. This research by Kim (2014) [12] was a duplicate of similar research by Razavian et al. [26] but used text datasets; in contrast, Razavian et al. [26] used imagery datasets for the experiment. Hence it was evidence that" feature extractors from pre-trained deep learning models perform well on a variety of tasks" [12]. The CNN model was borrowed from the work of Collobert et al. [27] with slight multichannel modification, two channels of word vectors: the first channel kept static throughout training, and the second one is fine-tuned via backpropagation. The datasets in word2vec embeddings (Mikolov et al.,2013) [21] came from Movie reviews (Pang and Lee, 2005) [28], Stanford Sentiment Treebank (Socher et al.,2013) [13], Subjectivity (Pang and Lee, 2004) [29], TREC question dataset (Li and Roth, 2002) [30], Costumer reviews (Hu and Liu, 2004) [31]. The final benchmarked outcomes suggested that pre-trained word vectors are suitable. Thus,' universal' feature extractors can be utilized across datasets.

Recursive Neural Tensor Network research by Socher et al. [13] introduced a model for predicting sentiments over treebanks; the main challenge of this research is that the sentiment must be predicted over compositional phrases. The dataset was taken from Stanford Sentiment Treebank. The basic concept of a treebank is that each phrase has been labeled in its compositional form. The main advantage of using the treebank over the bag-of-words approach is that a treebank does not ignore word order. The Stanford Sentiment Treebank uses movie reviews from rottentomatoes.com as the primary source and parses the data using The Standford Parser (Klein and Manning, 2003) [32], then had each comment labeled with a sentiment. The graphical form of Recursive Neural Models will parse n-gram inputs into a binary tree and computes in a bottom-up

fashion. RNTN by Socher et al. [13] was an improvement of the Recursive Neural Network by Socher et al. [33] and Matrix-Vector - RNN also by Socher et al. [34]. All of those approaches [13] [33] [34] used softmax; the main difference was the representation form of input phrases: RNN [33] used vector, MV-RNN [34] used vector-matrix, RNTN [13] used tensor. The fine-grained sentiments used were: very negative, negative, somewhat negative, neutral, somewhat positive, positive, and very positive. These sentiments were baselines for the sentiment of each phrase. There were some tests made to provide the assessment of RNTN model performances, and RNTN was benchmarked with other algorithms: MV-RNN, RNN, VecAvg2, Binary Na¨ıve Bayes, SVM, and Na¨ıve Bayes. RNTN outperformed those algorithms by up to 85.4 % for full-sentence binary classification. Contrastive conjunction sentences in the form' X but Y,' for example: "There are slow and repetitive parts. However, it has just enough spice to keep it interesting," for this type of dataset, RNTN obtains 41 % accuracy compared to MV-RNN, RNN, and Binary Na¨ıve Bayes. RNTN also outperformed other algorithms on negated sentences: "I liked a single minute of this film" is a positive sentence, "I did not like a single minute of this film" is a negated positive sentence, "It is just incredibly dull," is a negative sentence, "It is definitely not dull" is a negated negative sentence, for those types of sentences RNTN captured better performances. The last test was on the most positive and negative phrases, such as: "one of the best films of the year" and "best worst special-effects creation of the year," RNTN also resulted in better performances than RNN and MV-RNN.

Tweet2vec by Vosoughi et al. (2016) [7] was an effort to capture a vector representation of tweets that can be used for any classification task. The method was CNN - LSTM encoder-decoder model, which operates at the character level to learn and then vectorize the representation of tweets. The dataset was taken from 3 million randomly selected English tweets. The input features from a tweet were 70 characters x 150 matrices. The number 150 was taken from 140 characters plus ten paddings. The 70 x 150 matrix input matrix was given to CNN - LSTM model. After passing through the encoder section, the output representation of a tweet will be a 256-sized vector. The encoded representation will be fed to the decoder section. The resulting decoded matrix representation will be the final result (no exact matrix size is mentioned in the paper). After semantic relatedness and sentiment classification tests compared to Paragraph2Vec by Mikolov and Le, 2014 [35], this research's result outperformed Paragraph2Vec.

4. Deep Learning for Capturing Structure

The structure of the Twitter user, which represent connectivity among them, could be significant for racism detection. In real life, species with high feature similarities tend to live and build communities. The feature learning research by Vu and Parker (2015) [3] introduced a distributed representation of nodes in social network analysis as node embeddings. The node embedding learning model was adapted from word2vec (Mikolov et al., 2013) [21]. The main aim of the research was to introduce a generic method for learning embeddings from nodes in a social network based on their connectivity and attributes. The three aspects of mining on social networks used in the research are :

1. Community Homogeneity, the degree of closeness among its members
2. Community Distance, the average total distance for all members
3. Community Connectors Identification, identification of inter-community outliers who are not necessarily well-known but play significant roles in the social Network, also not influencers nor leaders

The dataset was taken from DBLP September 2013 citation network compiled by aminer.org [36]. The training process used 24 Na¨ıve Bayes classifiers over 24 fields of Computer Science, resulting in each field counts of members, authors, and papers (an author could be a member of a particular field community in Computer Science or not). The citation-based author embedding was a vector of 200 real numbers. There were two types of authors: the paper authors (citers) and the authors of citing papers (citees). Inbound citations can determine the homogeneity scores to a research community (IC) and outbound citations to a community outside (OOC). The research did provide the scores as a descriptive result. The last result of this research was determining community connectors, the authors who were considered outliers from the Data Mining community to the extent that they worked between two communities or that their work was related to multiple research fields. The paper stated those outliers sorted by their scores in descending order.

Perrozi et al.'s (2014) [4] research aimed to label graph-based text datasets using a combination of SkipGram dan Hierarchical Softmax; the technique was called DeepWalk. The concept plays an essential role in this algorithm. The datasets being used were: BLOGCATALOG [37] for interest labeling, FLICKR [37] for group labeling, and YOUTUBE [38] for group labeling. The baseline methods for comparison were: SpectralClustering [39], Modularity [37], EdgeCluster [38], wvRN [40], and Majority. The final comparison is 10% - 90% BLOGCATALOG

training data and 1% - 10% FLICKR and YOUTUBE training data of the DeepWalk algorithm. This results in better performance than other baseline algorithms.

Large-scale Information Network Embedding or abbreviated as LINE by Tang et al. (2015) [5], introduced the concepts of first-order proximity and second-order proximity and argued that the truncated random walk method as DeepWalk [4] only covered the second-order proximity. LINE aimed to introduce a large-scale network embedding model that preserves first- and second-order proximity. The LINE model was tested using three types of network domain corpora; the first came from the language domain and was an entire set of English WIKIPEDIA pages. The second domain came from Social Networks. There were two datasets, FLICKR and YOUTUBE [41]. The third domain came from the citation network, and the DBLP dataset [36] was used. Baseline methods for comparison were: Graph factorization [42], DeepWalk [4], and variations of LINE algorithms. The result for the language network domain was evaluated using two applications, word analogy [21] and document classification. On the first application, LINE with second-order proximity optimization outperformed all algorithms.

In contrast, with the second application, the concatenation of first-order proximity + second-order proximity outperformed all algorithms. The result for the social network domain was quite similar. The concatenation of first-order proximity + second-order proximity outperformed all algorithms; hence it can be concluded that the approach was quite effective and efficient for Network embedding regardless of the sparsity of the Network. The result for the citation network domain was slightly different. since the experiment only involved a directed graph, only second-order proximity was involved. The baseline method for comparison only involved DeepWalk [4], and the final result was that the second-order proximity outperformed DeepWalk both on the Author-Citation Network and Paper-Citation network.

Node2vec by Grover and Leskovec (2016) [8] can learn continuous feature representation for nodes. The research argues that classic approaches such as Principal Component Analysis, Multi-Dimensional Scaling, and those descendants [43] [44] [45] [46] are computationally expensive for large-world networks. The objective of node2vec is to preserve the neighborhood of nodes based on word2vec (Mikolov et al.,2013) representation. That objective was achieved using stochastic gradient-descent backpropagation on a single hidden-layer feed-forward neural Network. The performances were evaluated against Spectral Clustering [39], DeepWalk [4], and LINE [5], and there were increasing AUC scores results over Facebook, PPI, and arXiv datasets.

5. Tweets Detection for Capturing Time

Guille and Favre's (2014) [47] research incorporates time attributes for Twitter event detection. The research used two corpora: English and French. The text preprocessing only included removing trivial words and the stop-words. The input was partitions of n time-sliced tweets. The output was a list of events with the k-highest magnitudes. The approach was capturing bursty words' magnitude during a particular time and detecting special occurred events in the real world from those times. The event detection technique used graph approaches topic graph and redundancy graph for merging duplicated-redundant events. Then the actual real-world events were supervised by two annotators. The primary contribution of this research to the field of Twitter analytics was the capability to filter out non-related tweets for particular events.

REFERENCES

[1] Ed Mazza. https://www.huffingtonpost.com.au/entry/twitterracism-study n 4786283 racism on Twitter, by the numbers, 2014.

[2] Mohammed Hasanuzzaman, Ga¨el Dias, and Andy Way. Demographic word embeddings for racism detection on Twitter. In Proceedings of the Eighth International Joint Conference on Natural Language Processing, IJCNLP 2017, Taipei, Taiwan, November 27 – December 1, 2017 - Volume 1: Long Papers, pages 926–936, 2017.

[3] T. Vu and D. S. Parker. Node embeddings in social network analysis. In 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM), pages 326–329, Aug 2015.

[4] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. Deepwalk: Online learning of social representations. In Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '14, pages 701–710, New York, NY, USA, 2014. ACM.

[5] Jian Tang, Meng Qu, Mingzhe Wang, Ming Zhang, Jun Yan, and Qiaozhu Mei. Line: Large-scale information network embedding. In Proceedings of the 24th International Conference on World Wide Web, WWW '15, pages 1067–1077, Republic and Canton of Geneva, Switzerland, 2015. International World Wide Web Conferences Steering Committee.

[6] Long Jin, Yang Chen, Tianyi Wang, Pan Hui, and A.V.Vasilakos. Understanding user behavior in online social networks: a survey. Communications Magazine, IEEE, 51(9):144–150, September 2013.

[7] Soroush Vosoughi, Prashanth Vijayaraghavan,

and DebRoy. Tweet2vec: Learning tweet embeddings using character-level CNN-LSTM Encoder-Decoder. In Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '16, pages 1041–1044, New York, NY, USA, 2016. ACM.

[8] Aditya Grover and Jure Leskovec. Node2vec: Scalable feature learning for networks. In Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '16, pages 855–864, New York, NY, USA, 2016. ACM.

[9] Mainack Mondal, Leandro Ara´ujo Silva, and Fabr´ıcio Benevenuto. A measurement study of hate speech in social media. In Proceedings of the 28th ACM Conference on Hypertext and Social Media, HT '17, pages 85–94, New York, NY, USA, 2017. ACM.

[10] St´ephan Tulkens, Lisa Hilte, Elise Lodewyckx, Ben Verhoeven, and Walter Daelemans. A dictionary-based approach to racism detection in dutch social media. CoRR, abs/1608.08738, 2016.

[11] Eric H. Huang, Richard Socher, Christopher D. Manning, and Andrew Y. Ng. Improving word representations via global context and multiple word prototypes. In Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers - Volume 1, ACL '12, pages 873–882, Stroudsburg, PA, USA, 2012—Association for Computational Linguistics.

[12] Yoon Kim. Convolutional neural networks for sentence classification. CoRR, abs/1408.5882, 2014.

[13] Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Y. Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, pages 1631–1642, Stroudsburg, PA, October 2013. Association for Computational Linguistics.

[14] https://en.wikipedia.org/wiki/list of ethnic slurs list of ethnic slurs, 2018.

[15] http://rsdb.org/ racial slur database, 1999.

[16] https://www.crowdflower.com training data, machine learning and human-in-the-loop for a.i., 2018.

[17]https://www.cs.waikato.ac.nz/ml/weka/ weka the university of waikato, 2018.

[18] Timothy Quinn. https://www.hatebase.org/ world's largest online repository of structured, multilingual, usage-based hate speech, 2018.

[19]FBI.https://www.fbi.gov/investigate/civilrights/hate-crimes hate crimes – FBI, 2018.

[20]liwc.http://dx.doi.org/10.1075/dujal.6.1.04boo the dutch translation of the linguistic inquiry and word count (liwc) 2007 dictionary, 2017.

[21] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. CoRR, abs/1301.3781, 2013.

[22] http://scikit-learn.org machine learning in python, 2018.

[23] George A. Miller. Wordnet: A lexical database for English. Commun. ACM, 38(11):39–41, November 1995.

[24] Lev Finkelstein, Evgeniy Gabrilovich, Yossi Matias, Ehud Rivlin, Zach Solan, Gadi Wolfman, and Eytan Ruppin. Placing search in context: The concept revisited. In Proceedings of the 10th International Conference on World Wide Web, WWW '01, pages 406–414, New York, NY, USA, 2001. ACM.

[25] Andriy Mnih and Geoffrey Hinton. A scalable hierarchical distributed language model. In Proceedings of the 21st International Conference on Neural Information Processing Systems, NIPS'08, pages 1081–1088, USA, 2008. Curran Associates Inc.

[26] Ali Sharif Razavian, Hossein Azizpour, Josephine Sullivan, and Stefan Carlsson. Cnn features off-the-shelf: An astounding baseline for recognition. In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPRW '14, pages 512–519, Washington, DC, USA, 2014. IEEE Computer Society.

[27] Ronan Collobert, Jason Weston, L'eon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. Natural language processing (almost) from scratch. J. Mach. Learn. Res., 12:2493–2537, November 2011.

[28] Bo Pang and Lillian Lee. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics, ACL '05, pages 115–124, Stroudsburg, PA, USA, 2005. Association for Computational Linguistics.

[29] Bo Pang and Lillian Lee. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In Proceedings of the 42Nd Annual Meeting on Association for Computational Linguistics, ACL '04, Stroudsburg, PA, USA, 2004. Association for Computational Linguistics.

[30] Xin Li and Dan Roth. Learning question classifiers. In Proceedings of the 19th International Conference on Computational Linguistics - Volume 1, COLING '02, pages 1–7, Stroudsburg, PA, USA, 2002—Association for Computational Linguistics.

[31] Minqing Hu and Bing Liu. Mining and summarizing customer reviews. In Proceedings of the Tenth ACM SIGKDD

International Conference on Knowledge Discovery and Data Mining, KDD '04, pages 168–177, New York, NY, USA, 2004. ACM.

[32] Dan Klein and Christopher D. Manning. Accurate unlexicalized parsing. In Proceedings of the 41st Annual Meeting on Association for Computational Linguistics - Volume 1, ACL '03, pages 423–430, Stroudsburg, PA, USA, 2003—Association for Computational Linguistics.

[33] Richard Socher, Cliff Chiung-Yu Lin, Andrew Y. Ng, and Christopher D. Manning. Parsing natural scenes and natural language with recursive neural networks. In Proceedings of the 28th International Conference on International Conference on Machine Learning, ICML'11, pages 129–136, USA, 2011. Omnipress.

[34] Richard Socher, Brody Huval, Christopher D. Manning, and Andrew Y. Ng. Semantic compositionality through recursive matrix-vector spaces. In Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, EMNLP-CoNLL '12, pages 1201–1211, Stroudsburg, PA, USA, 2012. Association for Computational Linguistics.

[35] Quoc V. Le and Tomas Mikolov. Distributed representations of sentences and documents. In Proceedings of the 31st International Conference on Machine Learning, ICML 2014, Beijing, China, 21-26 June 2014, pages 1188–1196, 2014.

[36] Jie Tang, Jing Zhang, Limin Yao, Juanzi Li, Li Zhang, and Zhong Su. Arnetminer: Extraction and mining of academic social networks. In Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '08, pages 990–998, New York, NY, USA, 2008. ACM.

[37] Lei Tang and Huan Liu. Relational learning via latent social dimensions. In Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '09, pages 817–826, New York, NY, USA, 2009. ACM.

[38] Lei Tang and Huan Liu. Scalable learning of collective behavior based on sparse social dimensions. In Proceedings of the 18th ACM Conference on Information and Knowledge Management, CIKM '09, pages 1107–1116, New York, NY, USA, 2009. ACM.

[39] Lei Tang and Huan Liu. Leveraging social media networks for classification. Data Min. Knowl. Discov., 23(3):447–478, November 2011.

[40] Sofus A. Macskassy and Foster Provost. A simple relational classifier. In Proceedings of the Second Workshop on Multi-Relational Data Mining (MRDM-2003) at KDD-2003, pages 64–76, 2003.

[41] Alan Mislove, Massimiliano Marcon, Krishna P. Gummadi, Peter Druschel, and Bobby Bhattacharjee. Measurement and Analysis of Online Social Networks. In Proceedings of the 5th ACM/Usenix Internet Measurement Conference (IMC'07), San Diego, CA, October 2007.

[42] Amr Ahmed, Nino Shervashidze, Shravan Narayanamurthy, Vanja Josifovski, and Alexander J. Smola. Distributed large-scale natural graph factorization. In Proceedings of the 22Nd International Conference on World Wide Web, WWW '13, pages 37–48, New York, NY, USA, 2013. ACM.

[43] Mikhail Belkin and Partha Niyogi. Laplacian eigenmaps and spectral techniques for embedding and clustering. In Proceedings of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic, NIPS'01, pages 585–591, Cambridge, MA, USA, 2001. MIT Press.

[44] Sam T. Roweis and Lawrence K. Saul. Nonlinear dimensionality reduction by locally linear embedding. SCIENCE, 290:2323–2326, 2000.

[45] Joshua B. Tenenbaum, Vin de Silva, and John C. Langford. A global geometric framework for nonlinear dimensionality reduction. Science, 290(5500):2319–2323, 2000.

[46] S. Yan, D. Xu, B. Zhang, H. j. Zhang, Q. Yang, and S. Lin. Graph embedding and extensions: A general framework for dimensionality reduction. IEEE Transactions on Pattern Analysis and Machine Intelligence, 29(1):40–51, Jan 2007.

[47] A. Guille and C. Favre. Mention-anomaly-based event detection and tracking in twitter. In 2014 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2014), pages 375–382, Aug 2014.