Semantic Word Cloud Generation Based on Word Embeddings

Jin Xu* Yubo Tao† Hai Lin‡

State Key Laboratory of CAD&CG
Zhejiang University

ABSTRACT

Word clouds have been widely used to present the contents and themes in the text for summary and visualization. In this paper, we propose a new semantic word cloud taking into account the word semantic meanings. Distributed word representation is applied to accurately describe the semantic meaning of words, and a word similarity graph is constructed based on the semantic distance between words to lay out words in a more compact and aesthetic manner. Word-related interactions are introduced to guide users fast read and understand the text. We apply the proposed word cloud to user generated reviews in different fields to demonstrate the effectiveness of our method.

Index Terms: I.3.6 [Computer Graphics]: Methodology and Techniques—Interaction techniques

1 INTRODUCTION

Word clouds leverage different font sizes and colors to display important words in the text. They are useful to visually summarize the main contents of the text and intuitively guide users to explore the themes in the text. Most researches on word clouds focus on aesthetic design on word layout [24]. However, words are not independent, but have semantic meanings and dependencies. The random layout of words ignores these semantic meanings and relations between words. In order to make word clouds more readable for themes in the text, several improvements have been proposed to incorporate the semantic meanings of words [4, 26], which have a higher user satisfaction compared to other random layouts [20, 5].

The similarity between words is usually represented by the co-occurrence matrix in the previous word clouds. The semantic meaning in the word-word co-occurrence matrix is represented by a sparse semantic vector, whose length is the number of words in the dictionary. It is very memory intensive for large documents. Although LSA [7] or other dimensionality reduction techniques can apply to the word-document co-occurrence matrix to generate dense and short vectors, it only can represent the syntagmatic relation. In recent years, neural nets, such as the popular skip-gram [12] or CBOW [13] approaches, have been proposed to generate dense, short and semantically-meaningful vectors, called word embeddings. The basic idea under these word embedding techniques is that two words are similar if they tend to occur in similar contexts. They have improved performance in many natural language processing tasks. For example, the vector ("king") − vector ("man") + vector ("woman") is a vector very close to vector ("queen") in the word analogy task. Thus, this paper uses the word embeddings learned from large related corpora to measure the similarity between words.

In this paper, we present a new semantic word cloud based on word embeddings. A word similarity graph is first constructed from important words according to their semantic similarities, and then a force-directed layout is applied to generate a more compact and aesthetic word layout. Word-related interactions are provided to highlight the context of the word in both the word cloud and text for intuitively exploring the contents and themes in the text. Compared to previous methods, our method can capture the semantic meaning of words more accurate and effectiveness, and optimize the word layout to enhance the readability and aesthetic.

*e-mail: jinxcoder@gmail.com

†e-mail:taoyubo@cad.zju.edu.cn Corresponding Author

‡e-mail:lin@cad.zju.edu.cn

IEEE Pacific Visualization Symposium 2016
19–22 April, Taipei, Taiwan
978-1-5090-1451-4/16/$31.00 ©2016 IEEE

2 RELATED WORK

Many word cloud methods have been proposed, and some of them focus on improving the aesthetic on word layout. Kaser et al. [9] presented a method to reduce and balance the white space in word clouds. Seifert et al. [21] proposed a different layout strategy to cope with convex polygons as boundaries.

Semantic word clouds use clustered layout to indicate the word relation by the spatial distance. The similarity between words is largely estimated by the co-occurrence matrix in the previous methods. Cui et al. [4] proposed a semantic word cloud by computing the similarity between words using the co-occurrence counts. Wu et al. [26] created semantic-preserving word clouds with the co-occurrence matrix, and used the force-directed algorithm and seam carving technique to keep semantically similar words close to each other. Paulovich et al. [18] used the term frequency matrix and the covariance matrix to construct a semantic-preserving graph, and then employed spectral sorting to maintain the semantic relation among words. Barth et al. [2] constructed the co-occurrence matrix to calculate the cosine similarity coefficient between words as the word similarity. The semantic vector in these methods are sparse but memory intensive, as the length depends on the number of words in the dictionary. Furthermore, the captured semantic meaning is relatively weak compared to word embeddings.

Gansner et al. [8] measured the semantic similarity by LDA [3], and then transferred the semantic-preserving word cloud problem to classic graph layout problems. LDA is able to reduce the word representation dimension, but the similarity is in the topic level, not the word level. Wang et al. [25] applied the grammatical dependency graph to preserve the semantic information and used the force-directed graph drawing algorithm to arrange words. This method is computation intensive and tends to create a sparse layout. In this paper, we apply word embeddings as the semantic vector, and produces a more compact word layout.

3 SEMANTIC WORD CLOUD

Word clouds mainly provide the fragmentary information of the contents in the text. The motivation of our semantic word cloud is to preserve the semantic meanings of words to intuitively reveal general themes of texts for summary and exploration. As shown in Fig. 1, our method consists of four steps: semantic word representation, word similarity graph construction, force-directed word layout, and word cloud visualization.

3.1 Semantic Word Representation

The input data of our method is the texts and the pre-trained word embeddings.

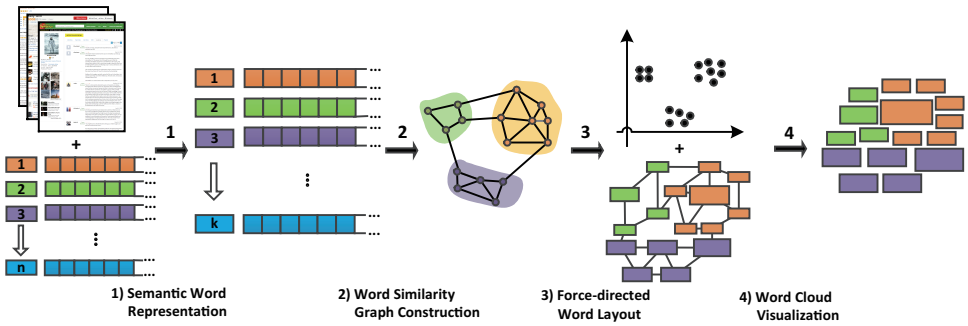

Figure 1: Workflow. Texts and pre-trained word embeddings are input data and the sematic word cloud generation consists of four steps.

For the texts, we first extract important key words from the documents. The OpenNLP toolkits are used to break down each document into a collection of sentences, and tokenize them into a collection of words. The stop words are filtered out, since they are mostly common but unimportant words. Meanwhile, due to different rules and special cases in the English language, we stem each word with WordNet [14] to reduce each word to its base. For instance, "writing", "wrote", and "written" can be reduced to "write". After these processing, remains words are key words.

The importance of key words is measured by the TF-IDF value. As a document may contain many themes and a theme may be described in many documents, we consider a sentence as a unit in the TF-IDF computation. We then select the top-k important key words as our words in the word cloud (We set k to 100).

The semantic meanings of words are provided by word embeddings in this paper, which is first used in the semantic word cloud generation to the best of our knowledge. We prepare the related text corpus and then train our word embeddings by using the continuous bag-of-words (CBOW) model [13], which is implemented in the open-source toolkit word2vec. CBOW is orders of magnitude faster than the others for training datasets and yields significant gains for dependency parsing [1]. After training, we extract the semantic meanings of all important key words from the word embeddings. By using the pre-trained word embeddings, each word corresponds a vector in the low dimensional space, typically 50-500.

3.2 Word Similarity Graph Construction

Important key words are represented as points in a high-dimensional space which is hard to understand and difficult to visualize. The goal is to analyze and visualize the relations of them and gain insight of clusters.

Dimensionality reduction techniques [23] can be directly applied to project points to a lower dimensional subspace, such as MDS [10]. However, they tend to create a sparse layout. In order to create a sematic word layout with a pleasing layout, we construct a word similarity graph, and then use the graph related algorithms to generate a semantic-preserving and aesthetic word layout.

The word similarity graph contains a node set and an edge set with from and to vertex tuples. The node set is the collection of the selected important key words. An edge is constructed between two words if the semantic similarity between two words is above the user specified threshold.

The similarity between two vectors can be measured by many metrics, such as the cosine and the jacard similarity. Since the semantic meaning of a word is described by the direction of the vector in word embeddings, the cosine similarity is widely used for the NLP task. Thus, we use the cosine similarity to measure the similarity between two words represented in word embeddings.

The higher the value of the cosine similarity, the more similar the two corresponding words in semantics. The cosine similarity value is also used as the weight of the edge.

We then conduct a clustering algorithm based Newman and Girvan's modularity [16] on the graph, which uses modularity to evaluate the strength of division of a graph into clusters. These clusters can better present the themes in the text.

3.3 Word Cloud Layout

After constructing the word similarity graph, we need to project the graph to a 2D space. We first apply MDS to initialize the nodes of the graph in 2D and then apply a force-directed graph layout with the energy model [17] to further optimize the positions of words to generate an aesthetically pleasing semantic word cloud.

MDS approximately preserves the distance between nodes, i.e., the semantic similarity between words, which can accelerate the convergence of the force-directed graph drawing algorithm by 4% - 7%. The energy model is based on the constructed similarity graph to perform the force-directed graph layout. It assigns the energy among the set of nodes and the set of edges with three kinds of energy, including the attraction energy, the repulsion energy and the gravitation energy. The algorithm iteratively searches for all possible solutions and the minimal energy value corresponds a good layout.

3.4 Word Cloud Visualization

With the initial positions of words, we can visualize words without overlapping. The words are positioned by the cluster. In the same cluster, words are rendered in the order of their importance. If the overlap or collision occurs, the word follows the Archimedean spiral to find a new position to render.

Clusters are encoded by the colors to help users understand different themes in the texts, which also indicate that the force-directed algorithm satisfies layout requirements. The font size is proportional to the importance value of the word.

As a word cloud should guide users fast read and understand the text, the framework supports the exploration of a specific word. When we click an interested word in the word cloud, the texts are sorted by the frequency of the word in the text, and the word is also highlighted in the texts. When we hover the mouse over an interested word, the contexts of the word in the document collection, which are defined as the words appearing in the same sentence with it, are highlighted and other words are faded, with colored by the number of co-occurrence from cool to warm.

4 RESULTS

In this section, we apply the proposed semantic word cloud to user generated reviews in different fields. In order to visually explore the large reviews and verify the effectiveness of our method, we have developed a system including two linked juxtaposed views: the semantic word cloud view in Fig. 2(a) and the text view in Fig. 2(b).

Figure 2: Overview. Our framework includes (a) the semantic word cloud view and (b) the text view. When we hover the mouse over the word "busy" and "sashimi" in (a), the contexts of "busy" and "sashimi" are represented in (c) and (d), respectively, with colored by the number of co-occurrence from cool to warm.

Figure 3: The word clouds of the reviews of a Japanese cuisine restaurant on Yelp with four different word embedding sets.

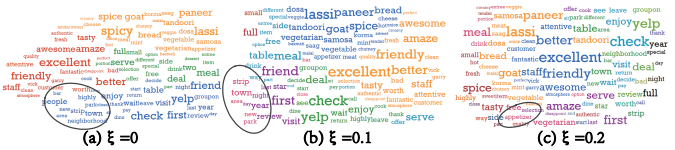

Figure 4: An Indian cuisine restaurant's reviews are visualized with different similarity threshold values.

4.1 Different Word Embedding Sets

We first analyse the influence of different word embedding sets on the result. We trained three word embedding sets from three different fields based on word2vec for experiments.

(i) Restaurant word embedding set. 99, 431 words, 400 dimensions; training corpus: the yelp academic dataset with 1, 569, 264 reviews.

(ii) Movie word embedding set. 17, 290 words, 400 dimensions; training corpus: 27 movies' reviews on Rotten Tomatoes with 27, 537 reviews.

(iii) Product word embedding set. 749,731 words, and 400 dimensions; training corpus: Amazon Electronics data released by McAuley et al. [11] with 3, 663, 769 reviews.

Besides the above trained word embedding sets, we also use two pre-trained word embedding sets for evaluation.

(i) HLBL [15, 22]. 246, 122 words, 100 dimensions; training corpus: RCV1 corpus, one year of Reuters English newswire from August 1996 to August 1997.

(ii) GloVe [19]. 1, 193, 514 words, 300 dimensions; training corpus: 42 billion tokens of web data.

The reviews of a Japanese cuisine restaurant on Yelp are used to generate four word clouds with four different word embedding sets. The word cloud using the restaurant word embedding set in Fig. 3(a) clearly shows four themes of the restaurant: comprehensive evaluation (pink), location (red), recommended dishes (blue), time information (dark green), and other information (grass green). The word cloud using the field-unrelated word embedding set in Fig. 3(b) arranges "waitress", "sashimi" and "strip" into the grass green group, which is semantically incorrect obviously. The word cloud using the GloVe word embedding set in Fig. 3(c) arranges "waitress" and "sashimi" into the red group. The word cloud using the HLBL word embedding set in Fig. 3(d) mainly consists of one grass green group, and we are not able to find other information.

Based on the results, we can conclude that it is more accurate to capture the semantic meanings of words using the word embedding set trained by the related text corpus, since the same word may have different meanings in the different fields [6]. In the following experiments, we use the field-related word embedding set to extract the semantic meanings of words.

4.2 Different Similarity Thresholds

In the word similarity graph construction, we filter out the less important edges below the similarity threshold ξ to improve the computational efficiency of the force-directed layout and the performance of the clustering algorithm.

An Indian cuisine restaurant's reviews are visualized based on the restaurant word embedding set. Fig. 4 shows the results with different similarity thresholds. The number of preserved edges is 6580, 1962, and 558, and the computation time is 5.22s, 4.67s and 4.45s for the threshold 0, 0.1, and 0.2, respectively. Comparing Fig. 4(a) with Fig. 4(b), we can see that the words associated with locations, such as "strip", "town", and "area", are clustered into another group in Fig. 4(b). There are too many clusters in Fig. 4(c), since it only preserves partial information about the graph. For instance, "appetizer" is not clustered into the orange group, which is about recommended dishes. Although the edge weight between words is the similarity value, edges with dissimilar words are still added and these edges can be considered as noises, which greatly affect the word layout. In the following experiments, the similarity threshold is set to 0.1.

4.3 Use Cases

We have applied our method to three domains as follows: movie domain, restaurant domain and product domain.

As Interstellar may be the hottest movie in 2014, we generate a word cloud from its reviews on Rotten Tomatoes with the movie word embedding set (Fig. 5(a)). The word cloud mainly consists of three groups. In the red group, as "director", "actor" and "excellent" stand out, we can infer this group is the comprehensive comments on the director, actors and movie. Interestingly, the word "dark" in the red is very strange. As the word "knight" is in the upper left of "dark", we can infer this may discuss another movie called The Dark Knight directed by the same director with Interstellar and reviewers tend to think they are both excellent, which is proven in the text view when clicking on "dark". The yellow group mainly discusses the movie plot including "mission", "father", "astronaut" and "life". By clicking on these words, we are able to learn that this movie features an astronaut, named Damon Matt, who has to execute a mission searching for a new home for the whole human life in the face of the separation with his daughter. The words, "two"

Figure 5: Web reviews from three domains a) movie: Interstellar, b) restaurant: a Japanese cuisine restaurant, and c) product: a type of Bose headphone. Our method summarizes and visualizes general themes with a clustered layout.

Figure 6: We visualize an Italian cuisine restaurant's reviews with four different word clouds.

and "three", denote that reviewers tend to compare movies with the similar plot. Since this movie involves a lot of physical theories, the blue group is about this content, and reviewers may get confused and want to explain what happened.

We visualize a Japanese cuisine's reviews on Yelp with the restaurant word embedding set (Fig. 5(b)). There are five groups on the word cloud. The grass green group, including "friend", "dinner" and "night", indicates that people tend to come here to have dinner with friends. The red group describes the restaurant location. Significantly, the word "Japanese" is colored red due to its expression of location, while it is obvious that "Japanese" mean Japanese cuisine in the reviews. Therefore, we can find the word cloud has placed "Japanese" close to the blue group, which is about the recommended dishes. The pink block is a comprehensive evaluation on the restaurant that consists of the words "friendly" and "nice". The dark green group mainly complains that the restaurant is busy and people have to wait. Fig. 2(c) displays the context of the word "busy" when we hover the mouse on it. We can see that the restaurant tend to be busy at night considering that "busy" most often appears with "night", "wait" and "dinner". Fig. 2(d) displays the context of the word "sashimi". The reviews on "sashimi" are positive and you can choose to try.

The reviews of a type of Bose headphone on Amazon are visualized using our word cloud with the product word embedding set (Fig. 5(c)). The word cloud mainly has three groups. The red group is about its functionality with the words "low", "frequency" and "reduction" and indicates that the functionality of this type of headphone is noise reduction especially when noise occurs at low frequencies. The orange group, including "airplane", "plane" and "fly", shows that people tend to use this headphone on the airplane thanks to its ability of cancelling noise. The blue group is still the comprehensive evaluation on the product with the words "recommend", "love" and "expensive". Thus, if you want to purchase a headphone to use on the plane for the noise reduction functionality and do not mind the price, you can choose it.

4.4 Comparison

Since our method uses word embeddings to measure the semantic similarity and uses the force-directed layout to arrange words, we compare our approach to three other word cloud methods, namely, the co-occurrence matrix semantic extraction with force-directed algorithm and seam carving technique (SCW) [26], grammatical dependency extraction with force-directed layout (GDW) [25], and MDS layout with word embeddings (MDSW). The reviews are from an Italian cuisine restaurant on Yelp, which have 167 sentences and 2149 words. Fig. 6 shows the word clouds generated by SCM, GDW, MDSW, and our method.

SCW has summarized the main contents but the accuracy of semantic information can not satisfy requirements. GDW uses grammatical dependency as semantic information, which focuses on capturing semantic relatedness. Due to this feature, GDW has to compute all the words' positions in the process of force-directed layout and only top-k words are displayed in the word cloud, which cause low computational efficiency and low space efficiency. MDSW creates a sparse layout. In our approach, general themes are clearly displayed including a comprehensive evaluation (purple), locations (blue), recommended dishes (red), and other information (grass green). The computation time of the four word cloud generation is 14.00, 20.49, 4.13, and 4.75 seconds on the Intel Core I5-4460 3.2 GHz CPU with 16 GB RAM, respectively. Our method is much faster than SCM and GDW and creates a more pleasing word cloud than MDSW.

5 Conclusions

In this paper, we have proposed a new semantic word cloud generation method. The semantic similarity between words is better captured in word embeddings, and the word similarity graph is constructed to improve the performance and aesthetically pleasing layout. Case studies on three review domains have demonstrated the effectiveness of our method in guiding users fast understanding the text. As the future work, sentiment information can be integrated for further analyzing the themes in the text.

Acknowledgements

The authors would like to thank the anonymous reviewers for their valuable comments and thank Yelp Data Challenge for making the data available. This work was partially supported by 863 Program Project 2012AA12A404 and National Natural Science Foundation of China No. 61472354.

REFERENCES

[1] M. Bansal, K. Gimpel, and K. Livescu. Tailoring continuous word representations for dependency parsing. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 809–815, Baltimore, Maryland, June 2014. Association for Computational Linguistics.

[2] L. Barth, S. G. Kobourov, and S. Pupyrev. *Experimental Algorithms: 13th International Symposium, SEA 2014, Copenhagen, Denmark, June 29 – July 1, 2014. Proceedings*, chapter Experimental Comparison of Semantic Word Clouds, pages 247–258. Springer International Publishing, Cham, 2014.

[3] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *the Journal of machine Learning research*, 3:993–1022, 2003.

[4] W. Cui, Y. Wu, S. Liu, F. Wei, M. X. Zhou, and H. Qu. Context preserving dynamic word cloud visualization. In *Pacific Visualization Symposium (PacificVis), 2010 IEEE*, pages 121–128. IEEE, 2010.

[5] S. Deutsch, J. Schrammel, and M. Tscheligi. *Human Aspects of Visualization: Second IFIP WG 13.7 Workshop on Human-Computer Interaction and Visualization, HCIV (INTERACT) 2009, Uppsala, Sweden, August 24, 2009, Revised Selected Papers*, chapter Comparing Different Layouts of Tag Clouds: Findings on Visual Perception, pages 23–37. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011.

[6] N. Diakopoulos, D. Elgesem, A. Salway, A. Zhang, and K. Hofland. Compare clouds: Visualizing text corpora to compare media frames. In *Proc. of IUI Workshop on Visual Text Analytics*, 2015.

[7] S. T. Dumais. Latent semantic analysis. *Annual review of information science and technology*, 38(1):188–230, 2004.

[8] E. R. Gansner, Y. Hu, and S. C. North. Interactive visualization of streaming text data with dynamic maps. *J. Graph Algorithms Appl.*, 17(4):515–540, 2013.

[9] O. Kaser and D. Lemire. Tag-cloud drawing: Algorithms for cloud visualization. *eprint arXiv:cs/0703109*, 2007.

[10] J. B. Kruskal. Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. *Psychometrika*, 29(1):1–27, 1964.

[11] J. McAuley, R. Pandey, and J. Leskovec. Inferring networks of substitutable and complementary products. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 785–794. ACM, 2015.

[12] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. In *Proceedings of Workshop at ICLR*, 2013.

[13] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, pages 3111–3119, 2013.

[14] G. A. Miller. Wordnet: A lexical database for english. *Commun. ACM*, 38(11):39–41, Nov. 1995.

[15] A. Mnih and G. E. Hinton. A scalable hierarchical distributed language model. *Advances in neural information processing systems*, pages 1081–1088, 2009.

[16] M. E. Newman. Modularity and community structure in networks. *Proceedings of the National Academy of Sciences*, 103(23):8577–8582, 2006.

[17] A. Noack. Modularity clustering is force-directed layout. *Physical Review E*, 79(2):026102, 2009.

[18] F. V. Paulovich, F. Toledo, G. P. Telles, R. Minghim, and L. G. Nonato. Semantic wordification of document collections. *Comput. Graph. Forum*, 31(3):1145–1153, 2012.

[19] J. Pennington, R. Socher, and C. D. Manning. Glove: Global vectors for word representation. *Proceedings of the Empiricial Methods in Natural Language Processing (EMNLP 2014)*, 12:1532–1543, 2014.

[20] J. Schrammel, M. Leitner, and M. Tscheligi. Semantically structured tag clouds: an empirical evaluation of clustered presentation approaches. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 2037–2040. ACM, 2009.

[21] C. Seifert, B. Kump, W. Kienreich, G. Granitzer, and M. Granitzer. On the beauty and usability of tag clouds. In *Information Visualisation, 2008. IV'08. 12th International Conference*, pages 17–25. IEEE, 2008.

[22] J. Turian, L. Ratinov, and Y. Bengio. Word representations: a simple and general method for semi-supervised learning. In *Proceedings of the 48th annual meeting of the association for computational linguistics*, pages 384–394. Association for Computational Linguistics, 2010.

[23] L. J. van der Maaten, E. O. Postma, and H. J. van den Herik. Dimensionality reduction: A comparative review. *Journal of Machine Learning Research*, 10(1-41):66–71, 2009.

[24] F. B. Viegas, M. Wattenberg, and J. Feinberg. Participatory visualization with wordle. *Visualization and Computer Graphics, IEEE Transactions on*, 15(6):1137–1144, 2009.

[25] J. Wang, J. Zhao, S. Guo, C. North, and N. Ramakrishnan. Recloud: semantics-based word cloud visualization of user reviews. In *Proceedings of the 2014 Graphics Interface Conference*, pages 151–158. Canadian Information Processing Society, 2014.

[26] Y. Wu, T. Provan, F. Wei, S. Liu, and K.-L. Ma. Semantic-preserving word clouds by seam carving. *Comput. Graph. Forum*, 30(3):741–750, 2011.