Predictive Top-Down Knowledge Improves Neural Exploratory Bottom-Up Clustering

Chihli Hung[1,2], Stefan Wermter[1], Peter Smith[1]

[1] Centre for Hybrid Intelligent Systems
School of Computing and Technology
University of Sunderland
Sunderland SR6 0DD, UK
http://www.his.sunderland.ac.uk
{chihli.hung, stefan.wermter, peter.smith}@sunderland.ac.uk
[2] De Lin Institute of Technology
Taipei, Taiwan
chihli@dlit.edu.tw

Abstract. In this paper, we explore the hypothesis that integrating symbolic top-down knowledge into text vector representations can improve neural exploratory bottom-up representations for text clustering. By extracting semantic rules from WordNet, terms with similar concepts are substituted with a more general term, the hypernym. This hypernym semantic relationship supplements the neural model in document clustering. The neural model is based on the extended significance vector representation approach into which predictive top-down knowledge is embedded. When we examine our hypothesis by six competitive neural models, the results are consistent and demonstrate that our robust hybrid neural approach is able to improve classification accuracy and reduce the average quantization error on 100,000 full-text articles.

1 Introduction

Document clustering is often performed under the assumption that predefined classification information is not available. Thus, the accuracy of clustering is mostly dependent on the definitions of cluster features and similarities since most clustering approaches organise documents into groups based on similarity measures. If the results of document clustering are compared with human classification knowledge, the accuracy depends on the difference between implicit factors of human classification assignment and explicit definitions of cluster features and similarities. However, pure unsupervised document clustering methods are sometimes unable to discern document classification knowledge hidden in the document corpus. One possible reason is that documents are classified not only on the basis of feature representation but also on the basis of human subjective concepts.

Clustering and classification are treated as methods to organise documents, and thus are helpful to access information [11]. Classification is supervised categorisation when classes are known; clustering is unsupervised categorisation when classes are not known. However, when different pre-assigned categories of documents contain many of the same features, i.e. words, it is not easy for traditional unsupervised clustering methods to organise documents based on their pre-classified categories [1].

An example of different decisions by document clustering and classification is illustrated in Fig. 1. There are nine documents which are pre-classified as two categories. Documents pre-classified as one category are represented as black circles and documents pre-classified as the other category are represented as white circles. However, based on mutual similarities of document vectors, nine documents form two clusters in Fig. 1A. The distance from document 1 to document 2 is shorter than that to document 5, so document 1 is in the same cluster as document 2 (Fig. 1B). Without embedding classification knowledge in the clustering approach, it is hard for document 1 to be grouped with document 5 (Fig. 1C).

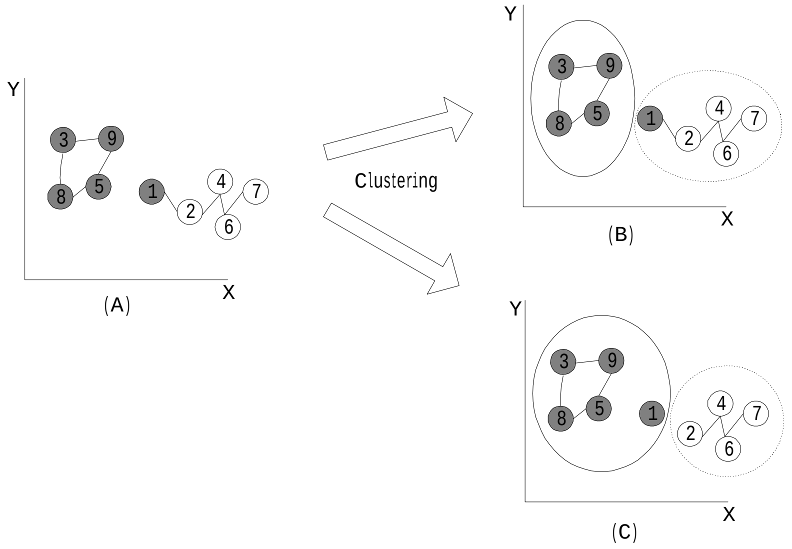

Fig. 1. An example of different decisions for document clustering and classification. Documents are represented as circles with numbers. Circles with the same filled colour are pre-assigned to the same category

Kohonen et al. [18] summarise the main purpose of neural text clustering as "*the goal is to organize a given data set into a structure from which the documents can be retrieved easily … this task is different from the pattern recognition task, in which the goal is to classify new items and, after learning, no attention is paid to the training data.*" Thus the main aim of the Self-Organising Map (SOM) [19] is to organise the given document set. They [18] point out that "*Obviously, one should provide the different words with such weights that reflect their significance or power of discrimination between the topics.*" They suggest using the vector space model (VSM) [28] to transform documents to vectors if no category information is provided. However, they also state that "*If, however, the documents have some topic classification which contains relevant information, the words can also be weighted according to their Shannon entropy over the set of document classes.*" A modified VSM which includes category information is used in their WebSOM project [18, 13].

In other words, an integration of clustering and classification knowledge may take advantage from an explicit mathematical definition of clustering similarity for the classification decision and achieve a higher accuracy for clustering using classification knowledge. Consequently, a *guided* neural network based on predictive top-down classification information offers the opportunity to exploit domain knowledge, which is able to bridge the gap of inconsistency between classification knowledge and clustering decisions.

In this paper, we explore the hypothesis whether integrating linguistic top-down knowledge from WordNet [24] into a text vector representation can improve neural exploratory bottom-up clustering based on classification knowledge. By extracting semantic rules from WordNet, terms with similar concepts are substituted with a more general term. To achieve our objectives, a series of experiments will be described using several unsupervised competitive learning approaches, such as pure Competitive Learning (CL) [19, 10], Self-Organising Map (SOM) [18], Neural Gas (NG) [22], Growing Grid (GG) [8], Growing Cell Structure (GCS) [7] and Growing Neural Gas (GNG) [6]. Our experiments show that hypernyms in WordNet successfully complement neural techniques in document clustering.

2 Current Reuters Corpus of news articles

We work with the new version of the Reuters corpus, RCV1 (It can be found at http://about.reuters.com/researchandstandards/corpus/), which consists of 806,791 news articles. There are 126 topics in this new corpus but 23 of them contain no articles. All articles except 10,186 are classified as at least one topic. In this paper, we concentrate on the most prominent 8 topics (Table 1) for our data set.

Table 1. The description of chosen topics and their distribution over the whole new Reuters corpus

Topic	Description	Distribution	
		no.	%
C15	Performance	149,359	5.84
C151	Accounts/Earnings	81,201	3.17
C152	Comment/Forecasts	72,910	2,85
CCAT	Corporate/Industrial	372,099	14.54
ECAT	Economics	116,207	4.54
GCAT	Government/Social	232,032	9.07
M14	Commodity markets	84,085	3.29
MCAT	Markets	197,813	7.73

We use the first 100,000 full-text news articles which are pre-classified according to the Reuters corpus. Because a news article can be pre-classified as more than one topic, we consider the multi-topic as a new combination topic in our task. Thus the 8 chosen topics are expanded to 54 topics (Table 2).

Table 2. The distribution of topic composition

No	Topic composition	Distribution	
		no.	%
1	ECAT/MCAT	1,034	1.03
2	CCAT	20,660	20.66
3	C15/C151/CCAT/ECAT/GCAT	32	0.03
4	C15/C151/CCAT	6,530	6.53
5	M14/MCAT	8,197	8.20
6	ECAT	7,368	7.37
7	CCAT/GCAT	3,557	3.56
8	CCAT/ECAT/GCAT	1,842	1.84
9	MCAT	11,202	11.20
10	GCAT	22,337	22.34
……			
53	C15/C151/CCAT/GCAT/M14/MCAT	1	0.00
54	C15/C151/C152/CCAT/ECAT	3	0.00
Total number of news articles		100,000	100.00

3 Extended Significance Vector Presentation

For clustering, each document must be transformed into a numeric vector. One candidate, the traditional Vector Space Model (VSM) [28] based on a bag-of-words approach is probably the most common approach. However, this model suffers from the curse of dimensionality while dealing with a large document collection because the dimensionality of document vectors is based on the total number of the different terms in the document collection. In our experiments, there are 7,223 words belonging to open-class words, i.e. nouns, verbs, adjectives, and adverbs, from 1,000 full-text news articles. In the 100,000 full-text news article task, there are 28,687 different words. Thus, some dimensionality reduction technique for a large scale document set is useful.

The most common way is leaving out the most common stop words, the most rare words and stemming a word to its base form. However, Riloff [26] suggests that these "unimportant words" will make a big difference for text classification. Only choosing the most frequent words from the whole specific word master list is also common [3]. However, there is no general heuristic to determine the threshold of the frequency. Some researchers consider this problem from a document structure viewpoint. They stress that only choosing the news headline, title, the first sentence of the first paragraph, the last sentence of the last paragraph, the first several lines or any combination above is meaningful enough for the full-text articles, e.g. [17]. However, this is decided by the information providers and therefore very subjective. Henderson et al. [12] choose so-called head of nouns and verbs using the Natural Language Processing (NLP) parser technique instead of full-text. This approach still depends on the text structure.

Another group of researchers uses vector representations and train them by clustering techniques, e.g. SOM. This cluster information from raw data is treated as input for other clustering or classification algorithms to produce a 2-stage clustering or classification model. The original version of WebSOM is one of them [13]. It consists of a word-topic SOM in its first stage and document SOM in its second stage. Pullwitt [25] proposes that the concept of a document comes from the concepts of sentences. He produces a sentence SOM and uses it to build a document SOM. Other researchers consider the dimensionality reduction problem from a mathematical view, such as Principal Component Analysis (PCA), Multi-Dimensional Scaling (MDS), Singular Value Decomposition (SVD), etc. [5]. Generally speaking, these approaches suffer from three shortcomings, which are computational complexity, information loss and difficult interpretation.

In our work, we propose another vector representation approach, which is called the extended significance vector representation. Dimensionality reduction is one major reason for using a different vector representation and another reason is the extraction of important features and filtering noise to improve the clustering performance. We do not remove common and rare words because of the evidence by Riloff [26] that these words are important. For the consistency, we restrict our experiments to those words found in WordNet, which only contains open-class words that are believed to be able to convey enough information of document concepts. The extend significance vector representation approach is started with the word-topic occurrence matrix, which is described as:

$$\begin{bmatrix} o_{11} \quad o_{12} \quad o_{13} \ldots\ldots o_{1M} \\ o_{21} \quad o_{22} \quad o_{23} \ldots\ldots o_{2M} \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ o_{N1} \; o_{N2} \; o_{N3} \ldots\ldots o_{NM} \end{bmatrix} . \qquad (1)$$

where o_{ij} is the occurrence of word i in topic j, M is the total number of topics and N is the total number of different words. An element of a significance word vector for a word i in topic j is represented as w_{ij} and is obtained using the following equation:

$$w_{ij} = \frac{o_{ij}}{\sum_{\tilde{j}=1}^{M} o_{i\tilde{j}}} \; . \qquad (2)$$

Equation 2 can be influenced by the different number of news documents observed in each topic. When a specific topic *j* contains much more articles than others, a word *i* may contain much more occurrences in topic *j* than in other topics. Therefore, most words may have the same significance weights in topic *j* and lose the discriminatory power to topics. Equation 3 is defined as the extended significance vector, which uses the logarithmic weights of the total number of word occurrences in the data set divided by the total number of word occurrences in a specific semantic topic to alleviate skewed distributions in Equation 2. A more prominent topic which contains more word occurrences will have smaller logarithmic values. Thus, the definition of an element in a word for word *i* for topic *j* is:

$$w_{ij} = \frac{o_{ij}}{\sum_{\tilde{j}=1}^{M} o_{i\tilde{j}}} \times \log \frac{\sum_{\tilde{i}=1}^{N}\sum_{\tilde{j}=1}^{M} o_{\tilde{i}\tilde{j}}}{\sum_{\tilde{i}=1}^{N} o_{\tilde{i}j}} \ . \qquad (3)$$

Then ,the news document vector $\vec{d}$ is defined as a summation of significance word vectors $\vec{w}_i = \left(o_{i1}\ o_{i2}\ ......o_{iM}\right)$ divided by the number of words in a document, which is defined as:

$$\vec{d} = \frac{1}{n}\sum \vec{w}\text{, where } n \text{ is the number of words in news document } d. \qquad (4)$$

4 Extracting Top-Down Semantic Concepts from WordNet

WordNet [24] is rich of semantic relationships of synset, which is a set of synonyms representing a distinct concept. In this work, we adopt the hypernym-hyponymy relationship from WordNet to get more general concepts and thus to improve the classification ability of the SOM. A hypernym of a term is a more general term and a hyponym is more specific. We use this relationship because its gist is similar to the definition of news cluster in that the concept of a cluster of news is more general than each distinct news article. News articles with a similar concept will be grouped in the same cluster.

The vocabulary problem describes that a term can be present in several synsets. Thus, a word in different synsets may be placed in a different hypernym hierarchy (Fig. 2). It is hard to determine the right concept for an ambiguous word from several synsets and it is hard to decide the concept of a document that contains several ambiguous terms. Brezeale [2] directly uses the first synset on WordNet because of the greatest frequency of occurrence in WordNet. Voorhess [30] proposes a method called *hood* to resolve this difficulty. An ambiguous word looks for its some level hypernym until finding the same hypernym in each hypernym tree. A hood is defined as the direct descendent of this same hypernym which is shared by different concepts of a term. The meaning of ambiguous words can be decided by counting the number of other words in the text that occur in each of the different sense's hoods. Then the specific hood with the largest number is represented as the sense of ambiguous words. Scott and Matwin [29] used *hypernym density* to decide which synset is more likely than others to represent the document. The hypernym density is defined as the number of occurrences of the synset within the document divided by the number of words in the document. The synset with higher density value is more suitable to represent the document.

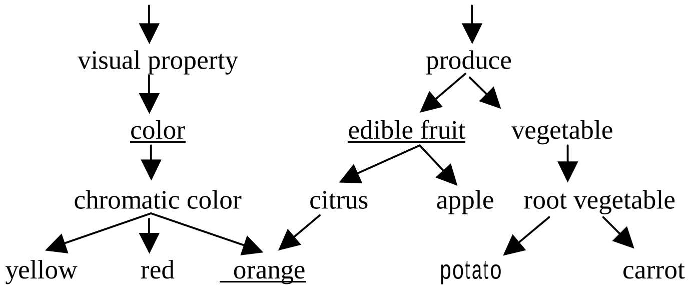

Fig. 2. An example of two hypernym trees for the term *orange*. The 2-level hypernym for orange with the colour concept is *color* but with the fruit concept is *edible fruit*

We do not use the synset directly but take advantage of the synset's gloss because two synonyms may not co-occur in a document, for example, color and colour, and orange and orangeness. The synset's gloss contains an explanation of the meaning and an example sentence of each concept. For example, the gloss of the word, *orange*, with the fruit concept is "*round yellow to orange fruit of any of several citrus trees*" and with the colour concept is "*any of a range of colors between red and yellow*". In contrast to synonyms, words in the gloss and their target word may be more likely to co-occur.

First, we have to convert each term in our semantic lexicon into its hypernym version for every topic. We treat each gloss as a small piece of the document with a core concept and transform the gloss using the extended significance vector representation. To decide the possible gloss for an ambiguous word, the specific element weights of each gloss vector in the specific topic of the original semantic lexicon is compared. The gloss vector with the highest weights in the specific element to represent the original word is chosen. For example, a comparison is made for the first element weight only when the ambiguous word occurs in topic 1. The second element weight is compared when the ambiguous word occurs in topic 2 and so on. Then going up 2-levels in the hypernym tree, we can use this hypernym to build our hypernym version of a semantic lexicon for all terms and all categories.

To describe this approach more clearly, the following example is given. Assume that the extended significance vector of the word *orange* in the semantic lexicon is [0.234 0.033 0.502 … 0.002] and its two gloss vectors with colour concept and with fruit concept are [0.101 0.203 0.302 … 0.031] and [0.201 0.103 0.222 … 0.021], respectively. When *orange* in topic 1 is converted to its hypernym, only the first element is compared for two gloss vectors. Thus, the gloss with fruit concept is chosen for *orange* in topic 1 since the first element in the gloss vector with fruit concept is greater than that with colour concept (0.201>0.101). When *orange* in topic 2 is converted to its hypernym, the colour concept is chosen (0.203>0.103). Therefore, the same word in the same topic has only one hypernym tree and different words in different topics may share the same hypernym tree (Fig. 3). Please note that to define the true meaning of an ambiguous word is not our purpose and this research rather bridges the gap of inconsistent decisions from the automated clustering technique and human classification.

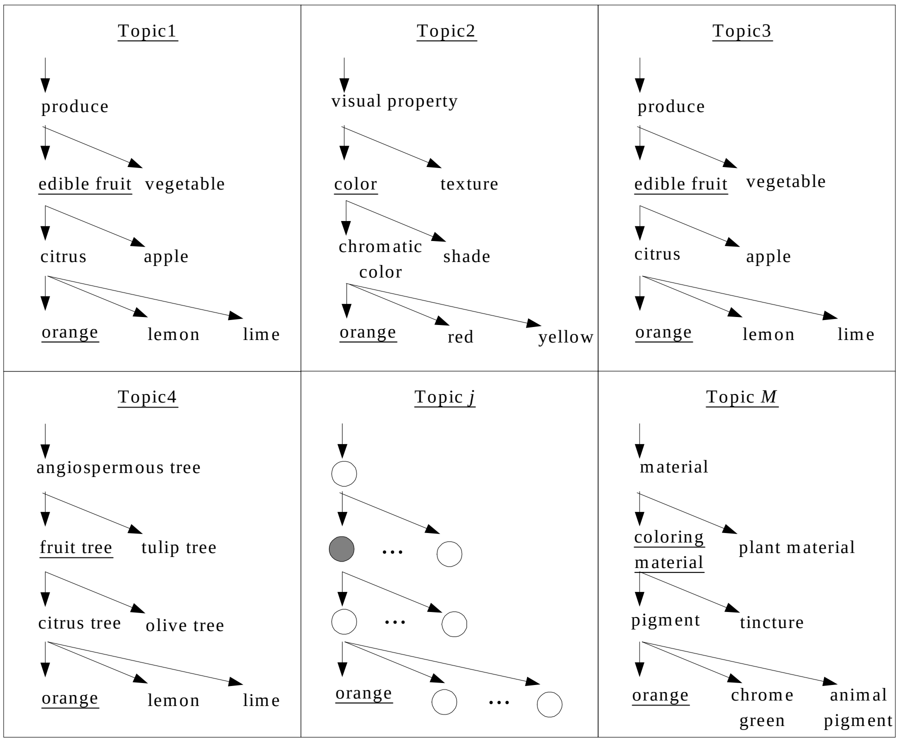

Fig. 3. An example of different hypernym trees for the term *orange*. There are four hypernym trees for this word in WordNet. Several different topics may contain the same trees, e.g. topics 1 and 3

Second, we convert each news article from its original version to its 2-level hypernym one. Because a look-up table of topic-word-hypernym has been built in stage one, we convert each word in each news document based on its classification topic and transform the 2-level hypernym data set to vectors by using the extended significance vector representation. This approach is successful to reduce the total number of distinct words from 28,687 to 11,239 for our 100,000 full-text test bed, and even improves the classification performance for several SOM-like models as we will show below.

5 Experiments with Six Competitive Learning Methods

There are several models which adopt the competitive learning principle [19, 10]. A common goal of those algorithms is to map a data set from a high-dimensional space onto a low-dimensional space, and keep its internal structure as faithful as possible. We divide these algorithms into 2 groups, i.e. static models and dynamic models, depending on whether the number of output units is static or not.

5.1 Static Models

We test our approach with three static competitive learning models, i.e. pure Competitive Learning (CL) [10, 19], Self-Organising Map (SOM) [18] and Neural Gas (NG) [22]. The main difference between them is the way they update their cluster centres. CL is a neural version of *k*-means [21], which always organises its *k* cluster centres based on the arithmetic mean of the input vectors. CL enforces the *winner-take-all* function so only the best matching unit (BMU) of the input vector is updated. SOM is a model which mimics the self-organising feature in the brain and maps the high dimensional input into a low dimensional space, usually 2. SOM defines its own neighbouring boundary and relation in a grid. Unit centres which are inside the neighbouring boundary are updated according to the distance to the input vector. The topographic map of SOM with WordNet is shown in Fig. 4. NG is a SOM-like model without the relations between its clusters, so the clusters are treated as the gas, which can spread in the input space. All unit centres are updated based on the distance to the input vector.

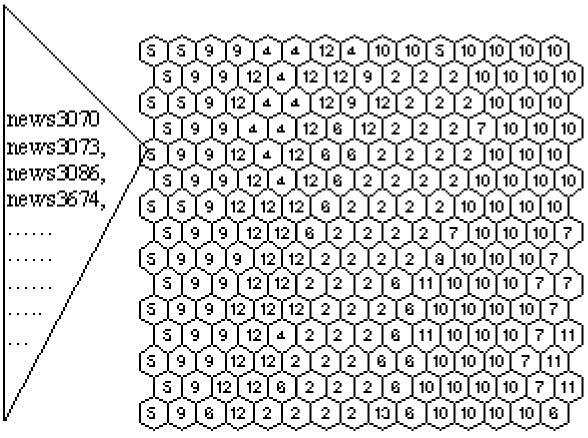

Fig. 4. The static model, SOM, with 15*15 units. Reuters topic codes are shown as numbers (Table 2)

5.2 Dynamic Models

Apart from the different definition of neighbourhood, dynamic models have variant dynamic representations. In this group of competitive learning algorithms, there is no need to define the number of units before training. These models will decide the number of units automatically. According to the work of Fritzke [9], a SOM model may have a good representation on the input vectors with uniform probability density but may not be suitable for complex clustering from the viewpoint of topology preservation.

In this work, Growing Grid (GG) [8], Growing Cell Structure (GCS) [7] and Growing Neural Gas (GNG) [6] are used to test our hypothesis which integrating symbolic top-down knowledge into vector representations can enhance text clustering. GG is an incremental variant of a SOM in terms of the model topology. It contains 2 stages, i.e. a growing stage, and a fine-tuning stage. Its update rule is the same in these 2 stages but the learning rate is fixed in the growing stage and is decayed in the fine-tuning stage to ensure the convergence. It starts with 2x2 units in a grid architecture which is able to keep the relative topographic relationships among units and represent input samples on a 2-dimensional map. Then GG develops the grid in column or row direction according to the position of the unit with the highest frequency of the BMU and the farthest direct neighbour of this highest BMU frequency unit.

GCS and GNG have a unit growing feature and a unit pruning feature as well. GCS is a dynamic neural model which always keeps its units with the triangle connectivity. GCS starts with 3 units and a new unit is inserted by splitting the farthest unit from the unit with the biggest error. A unit with a very low probability density, which means few input vectors are mapped to this unit, will be removed together with its direct neighbours of the corresponding triangle. GNG is a neural model applying GCS growth mechanism for the competitive Hebbian learning topology [23].

GNG starts with 2 units and connects an input sample's BMU to the second match unit as direct neighbours. A new unit is inserted by splitting the unit with the highest error in the direct neighbourhood from the unit with the highest error in the whole structure. Units will be pruned if their connections are not strong enough. Both GCS and GNG have 2 learning rates, which are applied to BMU and BMU's direct neighbours, respectively.

5.3 A Comparison of Performance between Six Competitive Learning Models

The evaluation of SOM-like models needs more careful analysis. The unsupervised feature of SOM usually needs the inclusion of the subjective judgements of domain experts [27]. Even though it is possible to see clusters in the SOM-like maps, human qualitative judgements should not be the only evaluation criterion. The main reason is that human judgements are subjective and different assessments can be made by the same person at a different time or different process.

Unlike qualitative assessment, quantitative criteria or cluster validity can be divided into two types: internal and external [16]. *Internal validity* criteria are data-driven and the average quantization error (AQE) is applied in this research. The AQE tests the distortion of the representation for the model and is defined by Kohonen [20] in Equation 5. *External validity* criteria evaluate how well the clustering model matches some prior knowledge which is usually specified by humans. The most common form of such external information is human manual classification knowledge so the classification accuracy is used in this research. These two evaluation criteria have been also used by several researchers, e.g. [18, 4, 31, 14, 15].

$$AQE = \frac{1}{N}\sum_{i=1}^{N}\left\|\vec{d}_i - \vec{w}_i\right\|, \text{ where } w_i \text{ is the weight vector of BMU for input} \quad (5)$$

sample *i* and *N* is the total number of input vectors.

We use 15x15 (225) units for each model and some other architectures have been tried with similar results. According to our experiments, if we use these models alone, we reach a classification accuracy between 54.60% and 61.35% for 100,000 full-text documents and an AQE between 2.721 and 2.434. Except GG, dynamic models are better in both evaluation criteria. This is because GNG and GCS contain the unit-pruning and unit-growing functions, which are able to adapt per se to input samples but GG only contains the unit-growing function and is confined its architecture to a grid, which may not reflect input samples faithfully.

Table 3. Classification accuracy and AQE without and with integration of WordNet 2-level hypernym for 100,000 full-text documents

Without WordNet	CL	NG	SOM	GG	GCS	GNG
Accuracy	54.60%	58.06%	58.22%	54.91%	57.55%	61.35%
AQE	2.437	2.444	2.708	2.721	2.492	2.434
With WordNet	CL	NG	SOM	GG	GCS	GNG
Accuracy	75.64%	80.90%	74.46%	74.60%	80.87%	86.60%
AQE	2.318	2.325	2.611	2.636	2.383	2.295
Improvement	CL	NG	SOM	GG	GCS	GNG
Accuracy	21.04%	22.84%	16.24%	19.69%	23.32%	25.25%
AQE	4.88%	4.87%	3.58%	3.12%	4.37%	5.71%

We achieve much better performance by integrating top-down knowledge from WordNet in all six algorithms based on two evaluation criteria. This hybrid approach achieves an improvement of classification accuracy from 16.24% to 25.25% and accomplishes between 74.46% and 86.60% accuracy. The AQE improvement varies from 3.12% to 5.71% and has smaller values between 2.295 and 2.636 for 100,000 full-text documents (Table 3).

6 Conclusion

In our work, we integrate symbolic top-down knowledge from WordNet into text vector representation using the extended significance vector representation technique. We examine the three static unsupervised models, Competitive Learning (CL), Neural Gas (NG) and, Self-Organizing Map (SOM) and three dynamic unsupervised models, Growing Grid (NG), Growing Cell Structure (GCS), and Growing Neural Gas (GNG) to test our hypothesis and approach. All results demonstrate that an integration of top-down symbolic information based on WordNet improves the bottom up significance vector representations in all six different approaches. Finally dynamic approaches, which determine their architecture during learning the task perform slightly better in average than static approaches. This is significant because it can avoid testing many static architectures. Our results demonstrate that our hybrid and dynamic neural model has a large potential for learning automatic text clustering.

References

1. Aggarwal, C.C., Gates, S.C., Yu, P.S.: On the Merits of Building Categorization Systems by Supervised Clustering. Proceedings of the fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, (1999) 352-356
2. Brezeale, D.: The Organization of Internet Web Pages Using WordNet and Self-Organizing Maps. Masters thesis, University of Texas at Arlington (1999)
3. Chen, H, Schuffels, C., Orwig, R.: Internet Categorization and Search: a Self-Organizing Approach. Journal of Visual Communication and Image Representation, Vol. 7. No. 1 (1996) 88-102
4. Choudhary, R., Bhattacharyya, P.: Text Clustering Using Semantics. The 11th International World Wide Web Conference, WWW2002, Honolulu, Hawaii, USA (2002)
5. Duda, R.O., Hart, P.E., Stork, D.G.: Pattern Classification, second edition. John Wiley & Sons, A Wiley-Interscience Publication (2001).
6. Fritzke, B.: A Growing Neural Gas Network Learns Topologies. In: Tesauro, G, Touretzky, D.S., Leen, T.K. (eds.): Advances in Neural Information Processing Systems 7, MIT Press, Cambridge MA (1995) 625-632
7. Fritzke, B.: Growing Cell Structures – a Self-Organizing Network for Unsupervised and Supervised Learning. Neural Networks, Vol. 7, No. 9 (1994) 1441-1460
8. Fritzke, B.: Growing Grid-a Self-Organizing Network with Constant Neighborhood Range and Adaptation Strength. Neural Processing Letters. Vol. 2, No. 5 (1995) 9-13
9. Fritzke, B.: Kohonen Feature Maps and Growing Cell Structures – a Performance Comparison. In: Giles, C.L, Hanson, S.J., Cowan, J.D. (eds.): Neural Information Processing Systems 5. Morgan Kaufmann, San Meteo, CA (1993)
10. Grossberg, S.: Competitive Learning: from Interactive Activation to Adaptive Resonance. Cognitive Science, vol. 11 (1987) 23-63
11. Hearst, M. A.: The Use of Categorise and Clusters for Organizing Retrieval Results. In: Strzalkowski, T. (ed.): Natural Language Information Retrieval, Kluwer Academic Publishers, Netherlands (1999) 333-374
12. Henderson, J., Merlo, P., Petroff, I. and Schneider, G.: Using Syntactic Analysis to Increase Efficiency in Visualizing Text Collections. 19th International Conference on Computational Linguistics (COLING 2002), Taipei, Taiwan (2002) 335-341

13. Honkela, T., Kaski, S., Lagus, K., Kohonen, T.: Exploration of Full-Text Databases with Self-Organizing Maps. Proceedings of the International Conference on Neural Networks (ICNN'96), Washington (1996) 56-61
14. Hung, C. and Wermter, S.: A Self-Organising Hybrid Model for Dynamic Text Clustering. Proceedings of The Twenty-third SGAI International Conference on Innovative Techniques and Applications of Artificial Intelligence (AI-2003), Cambridge, UK (2003)
15. Hung, C., Wermter, S.: A Dynamic Adaptive Self-Organising Hybrid Model for Text Clustering. Proceedings of The Third IEEE International Conference on Data Mining (ICDM'03), Melbourne, USA (2003)
16. Jain, A.K., Dubes, R.C.: Algorithms for Clustering Data. Prentice-Hall advanced reference series. Prentice-Hall, Inc., Upper Saddle River, NJ (1988)
17. Ko, Y., Seo, J.: Text Categorization Using Feature Projections. 19th International Conference on Computational Linguistics (COLING 2002), Taipei, Taiwan (2002)
18. Kohonen, T., Kaski, S., Lagus, K., Salojärvi, J., Honkela, J., Paatero, V., Saarela, A.: Self Organization of a Massive Document Collection. IEEE Transactions on Neural Networks, Vol. 11, No. 3 (2000) 574-585
19. Kohonen, T.: Self-Organization and Associative Memory. Springer-Verlag, Berlin (1984)
20. Kohonen, T.: Self-Organizing Maps. 3rd edition. Springer-Verlag, Berline, Heidelberg, New York (2001)
21. MacQueen, J.: On Convergence of K-Means and Partitions with Minimum Average Variance. Ann. Math. Statist., Vol. 36 (1965) 1084
22. Martinetz, T.M., Berkovich, S.G., Schulten, K.J.: Neural-Gas Network for Vector Quantization and Its Application to Time-Series Predication. IEEE Transactions on Neural Networks, Vol. 4, No. 4 (1993) 558-569
23. Martinetz, T.M.: Competitive Hebbian Learning Rule Forms Perfectly Topology Preserving Maps. International Conference on Artificial Neural Networks, ICANN'93, Amsterdam (1993) 427-434
24. Miller, G.A.: WordNet: a Dictionary Browser. Proceedings of the First International Conference on Information in Data, University of Waterloo, Waterloo (1985)
25. Pullwitt, D.: Integrating Contextual Information to Enhance SOM-Based Text Document Clustering. Neural Networks, Vol. 15 (2002) 1099-1106
26. Riloff, E.: Little Words Can Make a Big Difference for Text Classification. Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (1995) 130-136
27. Roussinov, D.G., Chen, H.: Document Clustering for Electronic Meetings: an Experimental Comparison of Two Techniques. Decision Support Systems, Vol. 27, (1999) 67-79
28. Salton, G.: Automatic Text Processing: the Transformation, Analysis, and Retrieval of Information by Computer. Addison-Wesley. USA (1989)
29. Scott, S., Matwin, S.: Text Classification Using WordNet Hypernyms. Proceedings of the COLING/ACL Workshop on Usage of WordNet in Natural Language Processing Systems (1998) 38-44
30. Voorhees, E.M.: Using WordNet to Disambiguate Word Senses for Text Retrieval. Proceedings of the Sixteenth Annual International ACM SIGIR conference on Research and Development in Information Retrieval (1993) 171 – 180
31. Wermter, S., Hung, C.: Selforganising Classification on the Reuters News Corpus. The 19th International Conference on Computational Linguistics (COLING 2002), Taipei, Taiwan (2002) 1086-1092