Table 1: Comparison of Document Representations on Different Benchmarks

Task	Benchmark	Bow	Ebow	Lda	Lsa	Para2vec
Adhoc Task	TripleTest	0.6135	**0.6205 (1.1%)**	0.5067 (-17%)	0.4691 (-24%)	0.5287 (-14%)
	KnnTest: $k = 5$	0.6222	0.6245 (0.4%)	0.5268 (-15%)	0.5845 (-6%)	0.5740 (-8%)
	KnnTest: $k = 20$	0.5380	0.5411 (0.6%)	0.4425 (-18%)	0.4724 (-12%)	0.4567 (-15%)
Diversity Task	TripleTest	0.4894	**0.5116 (4.5%)**	0.4271 (-13%)	0.4421 (-10%)	**0.5093 (4.1%)**
	KnnTest: $k = 5$	0.6458	0.6454 (-0.1%)	0.5776 (-11%)	0.6407 (-0.8%)	0.6274 (-2.9%)
	KnnTest: $k = 20$	0.5609	0.5604 (-0.1%)	0.5145 (-8%)	0.5415 (-3.5%)	0.5357 (-4.5%)

dicate the topic of a document, the bag-of-words vectorization is the default choice in existing works. Each document is represented as a sparse word vector, with components determined by tf-idf weighting. Since the term occurrences are assumed independent, their inter-relationship (e.g., synonymy) are neglected. **The bag-of-word sparse vector expanded with similarity among term embeddings.** (Ebow). To mitigate the sparsity of Bow, we further encode the term embeddings from word2vec [7] by expanding the Bow with similarity among term vectors. Inspired by recent word2vec [7] method in capturing the semantic similarity among terms, we expand the sparse document vector by multiplying each document vector with the term similarity matrix, thus effectively performing a document expansion. **Latent semantic analysis [4]** (Lsa) represents the documents into a latent topic space to overcome the sparsity in term space. In this paper, we show the results when 100 document dimensions are used. **Latent Dirichlet Allocation [5]** (Lda). Similar to Lsa, Lda conducts the dimension reduction with a generative model, mapping documents into a low-dimensional space. The topic number in Lda is set to 7. Both aforementioned parameter settings are based on our preliminary experiments. **Neural network based document vectorization [6]** (Para2vec). The recently proposed para2vec method co-trains the document vector together with the word vectors, capturing word co-occurrence information. As a variant of the word2vec [7] method, Para2vec can be regarded as a neural network based method to encode the word embedding information from word2vec [7] into document representations, whereas document expansion is used in Ebow.

3. EVALUATION

In this section, we describe the benchmarks and the comparison results on Trec Web Track[1] 2011–2014, based on ClueWeb 09 & 12 datasets from Lemur project[2], with 200 queries and 64k labeled documents (*qrel*) for adhoc and diversity tasks. In adhoc task, all 200 queries and all documents from *qrel* are used. In diversity task, 145 queries annotated with more than one subtopic and documents that are relevant to at least one subtopic are used. In Lsa, Lda and Para2vec, the document representation is computed separately for each query, given the size of the complete ClueWeb dataset. The results summarized in Table 1 are the average results among queries, with **bold numbers** indicating statistically significant improvements when compared against Bow. Intuitively, the comparisons among different document representations are in terms of their agreement degree to the desirable properties mentioned in the introduction, where cosine similarity is used. To measure this agreement, the following benchmarks are employed. **Direct comparison of similarity value (TripleTest).** To employ the document similarity in low-cost evaluation, the most important part is to distinguish document pairs that are both relevant to the query and those including one relevant and one non-relevant document. Thereby, we follow the document similarity benchmark used in [6]. In particular, for each query q, document triples (d_{r1},d_{r2}, d_n) are created from *qrel*, such that d_{r1} and d_{r2} are relevant to q, or relevant to same subtopic(s), and d_n is non-relevant, or is relevant to different subtopics from d_{r1} and d_{r2}. Similar to the metric used in [6], if d_{r1} and d_{r2} are more similar with each other than with d_n, the document triple is regarded correct. Different methods are compared based on the aggregated ratio between the correct triples and the total triples among queries. **Near-neighbor test (KnnTest).** Introduced in [8], the ratio of relevant documents among the k closed neighbors for each relevant document are examined. In this work, we examine this relevant document ratio for different k at 5, 20.

Table 1 shows that the agreement is not good enough in terms of absolute value, e.g., on TripleTest, 0.6 indicates that the boundary of relevant and non-relevant documents is blurred, and better representations are desirable to fulfill the low-cost evaluation task. Moreover, the results also indicate the introduction of word embedding to improve the document representation is non-trivial: Ebow improve Bow on TripleTest by **1.1%** and **4.5%** respectively, meanwhile Para2vec [6] performs worse on adhoc task.

[1]http://trec.nist.gov/tracks.html
[2]http://lemurproject.org/

4. REFERENCES

[1] B. Carterette and J. Allan. Semiautomatic evaluation of retrieval systems using document similarities. *CIKM* 2007.

[2] K. Hui and K. Berberich. Selective labeling and incomplete label mitigation for low-cost evaluation. *SPIRE* 2015.

[3] N. Jardine and C. J. van Rijsbergen. The use of hierarchic clustering in information retrieval. *Information storage and retrieval* 1971.

[4] T. K. Landauer, P. W. Foltz, and D. Laham. An introduction to latent semantic analysis. *Discourse processes* 1998.

[5] D. M. Blei, A. Y. Ng and M. I. Jordan. Latent dirichlet allocation. *JMLR* 2003.

[6] Q. V. Le and T. Mikolov. Distributed Representations of Sentences and Documents. *ICML* 2014.

[7] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. *NIPS* 2013.

[8] E. M. Voorhees. The cluster hypothesis revisited. *SIGIR* 1985.