ABSTRACT
This paper investigates the popular neural word embedding method Word2vec as a source of evidence in document ranking. In contrast to NLP applications of word2vec, which tend to use only the input embeddings, we retain both the input and the output embeddings, allowing us to calculate a different word similarity that may be more suitable for document ranking. We map the query words into the input space and the document words into the output space, and compute a relevance score by aggregating the cosine similarities across all the query-document word pairs. We postulate that the proposed Dual Embedding Space Model (DESM) provides evidence that a document is about a query term, in addition to and complementing the traditional term frequency based approach.

Categories and Subject Descriptors
H.3 [Information Storage and Retrieval]: H.3.3 Information Search and Retrieval

Keywords: Document ranking; Word embeddings; Word2vec

1. INTRODUCTION
A key challenge for information retrieval (IR) is to distinguish whether a document merely references a term or is about that entity. The traditional approach is to count repetitions of query terms in the document. However, as previously noted [5], the probabilistic model of IR can also consider additional terms that correlate with relevance. The two passages in Fig. 1 are indistinguishable for the query term Albuquerque under term counting but the presence of related terms like "population" and "metropolitan" points to the passage about the city. We propose to identify such related terms using word2vec.

Word2vec [2, 3] learns word embeddings via maximizing the log conditional probability of the word given the context word(s) occurring within a fixed-sized window. Therefore the learnt embeddings contain useful knowledge about word co-occurrence. A crucial detail often overlooked is that two different sets of vectors are learnt by the model corresponding to the input and the output words, henceforth referred to as the IN and OUT embeddings. By default, word2Vec discards the OUT vectors at the end of training. However, for certain IR tasks we postulate that we should use both the IN and the OUT embeddings jointly. Table 1 shows that the nearest neighbours of the word "yale" using IN-OUT vector cosine similarity produces words that often co-occur with "yale" (e.g., "faculty" and "alumni") as opposed to the IN-IN similarity which gives functionally similar words (e.g., "harvard" and "nyu"). We use this property of the IN-OUT embeddings to propose a novel Dual Embedding Space Model (DESM) for document ranking.

2. DUAL EMBEDDING SPACE MODEL
Given $q_i$ and $d_j$ as the embedding vectors for the $i^{th}$ and the $j^{th}$ term of the query and the document, respectively, we define the Dual Embedding Space Model as:

$$DESM(Q, D) = \frac{1}{|Q|} \sum_{q_i \in Q} q_i^T \frac{D}{\|D\|},$$

(1)

where,

$$D = \frac{1}{|D|} \sum_{d_j \in D} d_j / \|d_j\|.$$

(2)

$D$ is the centroid of all the normalized document word vectors serving as a single embedding for the whole document. Note that taking the centroid of the document word vectors is equivalent to computing the similarity between all query-document word pairs.

Albuquerque is the most populous city in the U.S. state of New Mexico. The high-altitude city serves as the county seat of Bernalillo County, and it is situated in the central part of the state, straddling the Rio Grande. The city population is 557,169 as of the July 1, 2014, population estimate from the United States Census Bureau, and ranks as the 32nd-largest city in the U.S. The Metropolitan Statistical Area (or MSA) has a population of 902,797 according to the United States Census Bureau’s most recently available estimate for July 1, 2013.

(a)

Allen suggested that they could program a BASIC interpreter for the device; after a call from Gates claiming to have a working interpreter, MITS requested a demonstration. Since they didn’t actually have one, Allen worked on a simulator for the Altair while Gates developed the interpreter. Although they developed the interpreter on a simulator and not the actual device, the interpreter worked flawlessly when they demonstrated the interpreter to MITS in Albuquerque, New Mexico in March 1975; MITS agreed to distribute it, marketing it as Altair BASIC.

(b)
Table 1: The nearest neighbors for the words "yale", "sea-
hawks" and "eminem" based on the IN-IN and the IN-OUT
vector cosine similarities. The IN-IN cosine similarities are high
for words that are similar by function or type (typical), and the
IN-OUT similarities are high between words that co-occur in
the same query or document frequently (topical).

<table>
<thead>
<tr>
<th>yale</th>
<th>seahawks</th>
<th>eminem</th>
</tr>
</thead>
<tbody>
<tr>
<td>yale</td>
<td>seahawks</td>
<td>eminem</td>
</tr>
<tr>
<td>harvard</td>
<td>alumni</td>
<td>orientation</td>
</tr>
<tr>
<td>49ers</td>
<td>broncos</td>
<td>packers</td>
</tr>
<tr>
<td>highlights</td>
<td>jerseys</td>
<td>tshirts</td>
</tr>
<tr>
<td>rihanna</td>
<td>ludacris</td>
<td>kanye</td>
</tr>
<tr>
<td>rap</td>
<td>featuring</td>
<td>tracklist</td>
</tr>
<tr>
<td>ntu</td>
<td>nfl</td>
<td>seattle</td>
</tr>
<tr>
<td>tufts</td>
<td>graduates</td>
<td>steellers</td>
</tr>
<tr>
<td>graduate</td>
<td>hats</td>
<td>2pac</td>
</tr>
<tr>
<td></td>
<td></td>
<td>performs</td>
</tr>
</tbody>
</table>

Also, the document embeddings can be pre-computed which is
important for runtime efficiency. We only need to sum the score
contributions across the query terms at the time of ranking.

As previously mentioned, the word2vec model contains two sepa-
rate embedding spaces (IN and OUT) which gives us at least two
variants of the DESM, corresponding to retrieval in the IN-OUT
space or the IN-IN space.

\[
DESM_{IN-OUT}(Q, D) = \frac{1}{|Q|} \sum_{q \in Q} \frac{q^T_{IN,i} D_{OUT}}{\|q_{IN,i}\| \|D_{OUT}\|} 
\]

\[
DESM_{IN-IN}(Q, D) = \frac{1}{|Q|} \sum_{q \in Q} \frac{q^T_{IN,i} D_{IN}}{\|q_{IN,i}\| \|D_{IN}\|} 
\]

We expect the \( DESM_{IN-OUT} \) to behave differently than the
\( DESM_{IN-IN} \) because of the difference in their notions of word
relatedness as shown in Table 1.

One of the challenges of the embedding models is that they can
only be applied to a fixed size vocabulary. We leave the exploration
of possible strategies to deal with out-of-vocab (OOV) words for
future investigation. In this paper, all the OOV words are ignored
for computing the DESM score, but not for computing the TF-IDF
feature, a potential advantage for the latter.

3. EXPERIMENTS

We train a Continuous Bag-of-Words (CBOW) model on a query
corpus consisting of 618,644,170 queries and a vocabulary size of
2,748,230 words. The queries are sampled from Bing’s large
scale search logs from the period of August 19, 2014 to August 25,
2014. We repeat all our experiments using another CBOW model
trained on a corpus of document body text with 341,787,174 distinct
sentences sampled from the Bing search index and a corresponding
vocabulary size of 5,108,278 words. Empirical results for both the
variants of the DESM, corresponding to retrieval in the IN-OUT
evaluation set consists of 7,741 queries randomly sampled from
Bing’s index with a vocabulary size of 480,608 words. The
queries are sampled from Bing’s large
corpus consisting of 618,644,170 queries and a vocabulary size
of 480,608 words. The queries are sampled from Bing’s large

discussions during a period of a few months. In total the final evaluation
set contains 171,302 unique documents across all queries which are
then judged by human evaluators on a five point relevance scale.

We report the normalized discounted cumulative gain (NDCG)
at different rank positions as a measure of performance for the
different models. The results show that the \( DESM_{IN-OUT} \)
outperforms both the BM25 and the LSA baselines, as well as the
\( DESM_{IN-IN} \) at all rank positions. The embeddings trained on
the query corpus also achieves better results than the embeddings
trained on body text. We provide additional analysis and experiment
results in [4].

4. DISCUSSION AND CONCLUSION

We formulated a Dual Embedding Space Model (DESM) that
leverages the often discarded output embeddings learned by the
word2vec model. Our model exploits both the input and the out-
put embeddings to capture topic-based semantic relationships. The
examples in Table 1 show that different nearest neighbours can be
found by using proximity in the IN-OUT vs the IN-IN spaces. In
our experiments ranking via proximity in the IN-OUT space per-
forms better for retrieval than the IN-IN based ranking. This finding
emphasizes that the performance of the word2vec model is applica-
tion dependent and that quantifying semantic relatedness via cosine
similarity in the IN space should not be a default practice.

References

and R. A. Harshman. Indexing by latent semantic analysis.

estimation of word representations in vector space.

Distributed representations of words and phrases and their

embedding space model for document ranking.

on theoretical arguments for idf. Journal of documentation, 60(5):