# A DATA-DRIVEN STRATEGY TO COMBINE WORD EMBEDDINGS IN INFORMATION RETRIEVAL

Alfredo Silva[1, 2] and Marcelo Mendoza[1, 2]

[1]Department of Informatics, Universidad Técnica
Federico Santa María, Santiago, Chile
[2]Millenium Institute of Foundational Research on Data, Santiago, Chile

***ABSTRACT***

*Word embeddings are vital descriptors of words in unigram representations of documents for many tasks in natural language processing and information retrieval. The representation of queries has been one of the most critical challenges in this area because it consists of a few terms and has little descriptive capacity. Strategies such as average word embeddings can enrich the queries' descriptive capacity since they favor the identification of related terms from the continuous vector representations that characterize these approaches. We propose a data-driven strategy to combine word embeddings. We use Idf combinations of embeddings to represent queries, showing that these representations outperform the average word embeddings recently proposed in the literature. Experimental results on benchmark data show that our proposal performs well, suggesting that data-driven combinations of word embeddings are a promising line of research in ad-hoc information retrieval.*

***KEYWORDS***

*Word embeddings, information retrieval, query representation.*

## 1. INTRODUCTION

Information retrieval is an essential area in information systems and document engineering. Its goal is to leverage query-document matching methods, from which a user can retrieve documents relevant to a query. These systems are organized into two components: a) Document retrieval, in which a set of documents potentially related to a query are retrieved, 2) Ranking: the retrieved documents are ordered using a ranking strategy. Usually, the document retrieval stage is performed by evaluating a bag-of-words query on an inverted index, from which the collection of documents containing the query words are retrieved. Then, the ranking phase is conducted using a word scoring function, which measures the relationship between the terms of the query and the terms of the document. Classic information retrieval schemes that follow this procedure are Tf-Idf [1] and BM25 [2].

These types of methods have limitations. One of the most important is its inability to identify homonyms relationships between the words of the queries and the documents. Since the query-document matching scheme depends on a lexical match, it is not possible to retrieve documents that contain semantically related terms to the query if they do not have a lexical match. To address this limitation, many researchers have proposed automatic query expansion (AQE) strategies [3, 4], adding related words to the query to retrieve more relevant documents.

David C. Wyld et al. (Eds): AIAP, SIGML, CNSA, NIAI - 2021
pp. 99-111, 2021. CS & IT - CSCP 2021 DOI: 10.5121/csit.2021.110107

AQE strategies define a candidate feature extraction method from a data source. Then by specifying a feature selection method, they select some potentially relevant terms to expand the query and reevaluate it. Usually, the data sources used for this purpose correspond to external sources, such as WordNet [5] or Wikipedia [6], or a local corpus built from the top-k ranked documents retrieved using the original query [7]. A feature selection method is used to define the level of relevance of each candidate word in the corpus with the query words. These methods cover a wide range of techniques and models to determine the significance of a term to the query. Among them, the literature reports methods based on mutual information [8], Kullback-Leibler divergence [9], and probabilistic term-to-term associations [10], among others. Once the terms of the expansion are selected, the original query is combined with the new words to generate an expanded query. Different word-combining methods have been tried, such as boolean queries [11], unweighted term combination, or Rocchio term-weighting [12]. Once the expanded query representation is available, the expanded query is evaluated using a query-document matching scheme such as Tf-Idf [1] or BM25 [2].

Representation learning has been a very active area in NLP in the last decade. Driven by a growing interest in the use of deep neural networks, several text representations have been proposed to solve different tasks supported on deep learning architectures. These representations operate at varying levels of aggregation, such as at the level of words, sentences, or documents. Word-level representations, known as word embeddings, allow building dense and low-dimensional vector representations. These embeddings allow identifying semantic relationships between words that do not have a lexical match. There are word embeddings like Word2Vec [13] and GloVe [14] that provide one embedding for each word. This approach has limitations as it makes processing polysemic words difficult, and it is unable to construct representations for out-of-vocabulary words. The out-of-vocabulary problem has been tackled using subword-based methods such as FastText [15], which can generate representations of out-of-vocabularies words from partial lexical matches.

Word embeddings can be used to find term expansion candidates and then retrieving documents using the expanded query. Different strategies have been studied for estimating the relevance of candidate terms to the query comparing a single candidate term to every query term. The literature shows that word embedding based query expansion achieves competitive results when combined with strategies based on relevance feedback [16]. Like the classic AQE methods, the use of a local corpus based, for example, on the top-k documents retrieved from the original query, allows obtaining more relevant words for the meaning of the original query than if using a global vocabulary [17]. For this reason, query-specific feature techniques predominate in AQE techniques based on word embeddings.

In this study, we compare different word embeddings strategies in AQE. We propose a data-driven strategy to combine word embeddings based on the IDF weights calculated on the corpus. Our approach aims to give more relevance to more informative terms, avoiding the inclusion of marginally relevant terms in the query expansion process.

The work is organized in the following way. In Section 2, we present related work. In Section 3, we discuss background knowledge on AQE. Our AQE strategy based on word embeddings is introduced in Section 4. Section 5 presents the experiments, the discussion of results, and the limitations of this work. Finally, we conclude in Section 6 providing concluding remarks and outlining future work.

## 2. RELATED WORK

The use of neural word embeddings in information retrieval is recent. This line of research has had increasing interest from researchers and practitioners in the last five years. One of the first works in this area evaluated the effectiveness of skip-grams and CBOW in the translation language model [18]. Both skip-grams and CBOW proved to be useful in ad-hoc information retrieval on TREC corpus. This success has prompted research on different IR tasks, such as sponsored search [19] and web recommendations [20].

The literature shows recent work in AQE based on word embeddings [21, 22, 17, 16, 23]. Usually, these works use a pseudo-relevant feedback scenario (PRF), which is useful for restricting the term search space. Query expansion based on one-to-one relationships using word embeddings was studied by Almasriet al. [23]. The authors used cosine similarity to find terms related to each query term. Then a new query was built joining the identified words. Ad-hoc IR was evaluated using the query likelihood model on the expanded query. Results on CLEF data showed that their effectiveness was better than one-to-one AQE methods based on mutual information and PRF.

Other works build a representation of the query based on word embeddings. The terms for the expansion are identified using a similarity function between the vector representation of the query and the word embeddings of the terms that make up the (pseudo)-relevant documents. To construct the query vector representation, these methods average the word embeddings of the query terms, an approach known as average word embeddings (AWE). Both Roy et al. [22] and Kuziet al. [21] studied the effectiveness of AWE vectors on AQE using cosine similarity, showing good performance when used in combination with PRF. Recently, Imani et al. [24] showed that cosine similarity can be replaced by a siamese network trained to detect candidate words.

Diaz et al. [17] incorporate relevance feedback in the process of learning the term embeddings. The idea is to retrieve a set of documents for the query to learn a query-specific term embedding model. Experimental results show that the terms of the query expansion identified using this procedure are more specific than those obtained by using embeddings trained on a global corpus. Along these same lines, Zamani and Croft [16] proposed a query likelihood model in which query embeddings are estimated in a local corpus. Using these data-driven query embeddings improves the effectiveness of the query expansion process. Driven by these promising results, Zamani and Croft [25] proposed relevance-based word embeddings, a word embedding strategy that learns word representations based on query-document relevance information. The authors showed that these word representations performed better on AQE than those based on word2vec or GloVe.

## 3. BACKGROUND

The AQE strategy studied in this work is based on pseudo relevance feedback. Consequently, the AQE strategy considers two phases. In the first phase, the top-k documents are retrieved using an information retrieval model. In the second phase, we use the top-k documents retrieved as a local corpus. This corpus allows us to define the extraction phase of characteristics of the query in a query-specific context, focusing the AQE technique on a local vocabulary conditioned on the query. The assumption of relevance, or pseudo relevance feedback, is used by the AQE strategy to focus on the domain in which the candidate terms for expansion are sought.

Let q be a bag-of-words query (BOW) composed by an arbitrary amount of terms that belong to a vocabulary V. Let $D=\{d_{1,}...,d_N\}$ be a collection of documents, or global corpus, in which the query q is evaluated. A query engine retrieves the posting lists of the inverted index of D for each word in q. After retrieving the posting lists, a set of documents $D'$ is available. Then, for each document $d \in D'$, a term-scoring function $\mathrm{Term}-\mathrm{score}(w_i,d)$ is used to rank the documents according to its relevance to q. Then, we calculate the ranking of d by adding the term-scoring obtained for each word of q:

$$\mathrm{Score}(q,d) = \sum_{w_i \in q} \mathrm{Term}-\mathrm{score}(w_i,d).$$

Note that the term-scoring function may includes the IDF function. The literature shows many ways to calculate this function. We choose an expression that performs well in collections where the co-occurrence between terms and documents is sparse. Accordingly, IDF is given by the following expression:

$$\mathrm{IDF}(w_i) = \log\left(\frac{N-n(w_i)+0.5}{N+0.5}\right),$$

Where N is the number of documents in D and $n(w_i)$ corresponds to the number of documents in which $w_i$ occurs.

The $\mathrm{Score}(q,d)$ function allows sorting the documents in $D'$ according to their relevance to q. To build the local corpus on which the expansion of the q query is made, the list of documents is truncated. Accordingly, the top-k documents in $D'$ correspond to the local corpus from which the expansion terms will be searched. In this study, we work with a local corpus made up of the top-10 documents. Working with a small corpus allows us to focus the AQE strategy on a specific vocabulary, avoiding terms that are marginally relevant to q can be considered in the expansion process. Using a small local corpus also avoids introducing high computational costs during the expansion process.

Our AQE strategy uses a query re-weighting model from which the terms of the query expansion can be identified. To do this, we build a query representation based on the word embeddings of the terms that compose it. This query representation is used to identify related terms. We use pre-trained word embeddings on large text collections to fulfill this purpose.

## 4. PROPOSAL

### 4.1. IDF-AWE

To identify the terms of the expansion, we created a representation of the query based on word embeddings. Both [17] and [21] use a similar approach named average word embeddings (AWE), which gives the same weight to each term of the query:

$$\mathrm{AWE}(q) = \frac{1}{n}\sum_{w_i \in q} \overrightarrow{w_i}, \quad (1)$$

Where $\overrightarrow{w_i}$ is the word embedding of the query term $w_i$, and n is the number of terms in q. Note that Equation (1) can also be used to build representations of documents based on word embeddings.

Kuziet al. [21] proposes the use of the closest term, measured by cosine similarity between the AWE vector and the word embedding of each candidate word, to determine which words will be incorporated in the query expansion. The authors define a scoring function for this purpose:

$$S(w,q) = e^{\cos(\overrightarrow{w},AWE(q))}, \quad (2)$$

Where w is a candidate word to be included in the query expansion process, and $\overrightarrow{w}$ is its word embedding.

We extend AWE, introducing a data-driven strategy to combine the word embeddings. Our vector representation, IDF-AWE, is a representation of the query generated from the linear combination of word embeddings according to their IDF weights:

$$IDF - AWE(q) = \frac{1}{\sum_{w_i \in q} IDF(w_i)} \sum_{w_i \in q} IDF(w_i) \cdot \overrightarrow{w_i}, \quad (3)$$

Where IDF is calculated with respect to the whole corpus. IDF-AWE will give more importance in representing q to terms that are more specific in the corpus. To keep the magnitude of the word embeddings, the sum of the vectors is scaled by the sum of the factors IDF of the terms that belong to the query. In this way, the IDF-AWE vector gets the same magnitude as the word embeddings. Note that Equation (3) can also be used to build representations of documents based on word embeddings.

Once the IDF-AWE vector is computed, we can use it to score the candidate terms to be used in the expansion. For this purpose, we use the score function defined in Equation (2). The expanded query will include the top-T terms closest to the IDF-AWE vector representation of q. The number of terms of the expansion, the value of T, corresponds to a parameter of the AQE strategy. These terms define a new query, denoted as $q_{exp}$, that will be re-evaluated in the query engine.

## 4.2. ElMo

The definition of the IDF-AWE vector for a context-dependent word embedding is different. In our study, we examined the effectiveness of ElMo [26], which defines a context-conditioned word embedding. To make use of this particularity that ElMo offers, we search for the occurrences of each query word in each document of D′. Then, we use the surrounding text detected around each occurrence of $w_i$ as a context. Let $w_i$ be a word from the query q and let $S_j(w_i,d) = \langle w_{i-N},...,w_i,...,w_{i+N} \rangle$ be a surrounding text of $w_i \in d$, whose length is 2N + 1. By providing ElMo with the context $S_j(w_i,d)$, we obtain a context-conditioned word embedding, which we will denote $\overrightarrow{w_{i,j}}$. Let's assume that $w_i$has M matches in d and therefore, ElMo returns M word embeddings$\overrightarrow{w_{i,j}}$, $j \in \{1,M\}$. We obtain a word embedding for $w_i$conditioned on d by averaging the ElMo word embeddings obtained in d:

$$\overrightarrow{w_i} = \frac{1}{M} \sum_{j=1}^{M} \overrightarrow{w_{i,j}}. \quad (4)$$

Once the ElMo word embedding for $w_i$ conditioned on d is obtained, we can compute the IDF-AWE vector using the Equation (3). The length of the context window was set to 5 (N=2), as usual in word embeddings [13].

The retrieval phase of relevant documents from the expanded query $q_{exp}$ is implemented using the disjunction of the terms included in the expansion. This variant that incorporates our AQE strategy retrieves the posting list of the inverted index of D for each word of $q_{exp}$. Instead of intersecting the postings lists, it retrieves all the documents addressed in these lists. Since the IDF-AWE vector gives more relevance to the more specific terms in D, the words of the expansion should be strongly related to these specific words. We hypothesize that by giving more relevance to specific words in the representation of q, the words in $q_{exp}$ will also be more specific. If this is effective, the AQE strategy is expected to improve the effectiveness of the system by joining the posting lists of these terms.

### 4.3. Re-ranking

Let $\text{top} - T(q) = \{t_1,...,t_T\}$be the terms of the expansion of q. Let $p(t_i)$be the posting list of each term $t_i$in $\text{top} - T(q)$. A new set of documents is defined by $D^{exp} = \{Up(t_i) \vee \forall t_i \in \text{top} - T(q)\}$, that is, joining the posting lists of the terms of the expansion. Our AQE strategy ranks $D^{exp}$, applying a term-scoring function to each term of $q_{exp}$and each document in $D^{exp}$. Once the documents in $D^{exp}$are ranked, a global ranking of the documents in $D'UD^{exp}$is obtained, applying a re-ranking function. Our re-ranking function combines the rankings of both ordered lists of relevant documents, using a linear combination of the rankings obtained in each of the stages of the AQE strategy.

Let d be a document in $D'UD^{exp}$. The scoring function for d in our AQE system is given by:

$$\text{Score}\left(\left\langle q,q_{exp}\right\rangle,d\right) = (1-\alpha)\cdot \text{Score}(q,d)+\alpha\cdot \text{Score}\left(q_{exp},d\right), \quad (4)$$

Where $\alpha$ controls the relative importance of q and $q_{exp}$in the global ranking. Note that $\text{Score}(q,d) = 0$ if $d \notin D'$and $\text{Score}\left(q_{exp},d\right) = 0$if $d \notin D^{exp}$.

We introduce a variant of this re-ranking function, which, instead of combining the document-level rankings, modifies the BM25 ranking function according to the linear combination factors. To do this, each document is evaluated in a new query that has the terms of the original query and those of the expanded query. If the query term belongs to the original query, its BM25 term-scoring is weighted according to $1-\alpha$. If the query term belongs to the expanded query, its BM25 term scoring is weighted according to the $\alpha$ value. The following expression gives the re-ranking function:

$$\text{Score}\left(qUq_{exp}\right) = (1-\alpha)\cdot \sum_{w_i\in q} \text{BM}\,25(w_i,d)+\alpha\cdot \sum_{w_i\in q_{exp}} \text{BM}\,25(w_i,d). \quad (5)$$

## 5. EXPERIMENTS

### 5.1. Datasets

The data available to evaluate the proposed strategies correspond to two partitions of the Tipster corpus, also known as the Text Research Collection Volume (TREC). The two Tipster partitions studied in this section are: Associated Press (AP) and Wall Street Journal (WSJ). Table 1 shows some basic statistics of these collections.

Table 1. Collections used in this study.

| Corpus | Years | Documents | Tokens | Vocabulary |
|---|---|---|---|---|
| WSJ | 1987 - 1992 | 173252 | $81 \cdot 10^6$ | 410578 |
| AP | 1988 - 1990 | 239302 | $114 \cdot 10^6$ | 417490 |

The queries used in this study correspond to the enumerated TREC queries 51-200, of which there are also qrels that allow evaluating the effectiveness of the proposed strategies. The 150 test queries were evaluated in the three datasets for each of the experimental configurations studied in this section. All corpora in Table 1 were stopped using the SMART stopword list and stemmed using the Krovetz algorithm.

## 5.2. Word embeddings used in this study

We study five strategies of word embeddings. Two variants of word2vec [13], Skip-grams, and CBOW, were studied in this work. Both strategies were applied to a corpus that joins the two TREC datasets evaluated (WSJ + AP). Both CBOW and Skip-grams were trained using sliding windows with five terms, using an FFNN with a hidden layer of 300 units.

For GloVe, FastText, and ElMo, we use pre-trained vectors. GloVe vectors [14] were pre-trained on Wikipedia 2014 + Gigaword 5, two large-scale text collections. GloVe vectors also have 300 dimensions and it has a vocabulary of 400,000 terms. GloVe's match on each dataset is 89,142 terms in SJMN, 103,700 terms in WSJ, and 120,218 in AP.

FastText vectors [15] were pre-trained in a corpus of over 2.5 million materials science articles. FastText vectors have 100 dimensions. Since they are generated from subwords, they produce a full match with the TREC dataset vocabularies.

ElMo vectors [26] were pre-trained in a corpus called 1 Billion Word Language Model Benchmark. ElMo vectors have 1024 dimensions. These vectors also generate a full match with the TREC dataset vocabularies.

## 5.3. Experimental setting

We study different variants of query-document matching methods that can be implemented using the concepts discussed in Sections 3 and 4. These variants allow us to identify the impact of each element of the proposal on the effectiveness of the AQE strategy. To illustrate the usefulness of each building block of the AQE strategy, we defined five variants, two of which do not implement AQE, and three of which implement different variants of the AQE strategy. Each of these variants is detailed below:

AWE-VS: In this variant of the proposal, called AWE-based vector space, the documents in $D'$ are ranked using the cosine similarity between the AWE vector of the query and the AWE vector of each document in $D'$. AWE-VS does not consider query expansion. AWE-VS is evaluated using four word embedding strategies; these are, Skip-grams, CBOW, GloVe, and FastText. ElMo is not evaluated in this variant of the model since the construction of the vector representation of documents based on context-dependent encoding is very costly in computational time.

IDF-AWE-VS: In this variant of the proposal, called IDF-AWE-based vector space, the documents in $D'$ are ranked using the cosine similarity between the IDF-AWE vector of the query

and the IDF-AWE vector of each document of D′. IDF-AWE-VS does not consider query expansion. It is evaluated in CBOW, Skip-grams, GloVe, FastText, and ElMo. In the case of ELMo, it is more difficult to obtain a vector representation of the document since ElMo is context-dependent. In this configuration, all the occurrences on the document of each query term were searched. The AWE vector of the document context was obtained from them, using a term window of 5 words around the query term. Then, the context AWE vectors for each query term were averaged to generate one AWE vector per query term. The IDF-AWE vector of the document was obtained by combining the AWE vectors per word according to their Idf weights.

AQE-Cent: AQE-Cent uses the AWE vector of q to identify the terms of the expansion using the term-scoring function defined in Equation (2). Five variants of word embeddings are evaluated to construct the representation of the query (CBOW, Skip-grams, GloVe, FastText, and ElMo). The variants based on CBOW and Skip-grams correspond exactly to those studied by Kuziet al. [21]. The candidate terms of the expansion correspond to the terms of the top-10 documents of D′ according to the BM25 score obtained for the original query. The term-scoring function allows defining T terms related to q, which make up a new query. This new query is evaluated in D, and its results are ranked according to BM25. The results of both rankings were consolidated in a global ranking using the re-ranking function defined in Equation (4).

IDF-AWE-VS + AQE-Cent: This variant also uses AQE-Cent to represent the query during the expansion process. However, instead of searching for the terms in the top-10 ranked documents of D′ according to BM25, it ranks them using IDF-AWE-VS. In this way, the search space for terms is different, giving more prominence to documents that are close to the IDF-AWE query vector. IDF-AWE-VS + AQE-Cent is evaluated using Skip-grams, CBOW, GloVe, and FastText. The terms of the expansion are determined using the term-scoring function defined in Equation (2). Then, the expanded query is evaluated in D, and its documents are ranked using BM25. The results of both rankings are consolidated in a global ranking using the re-ranking function defined in Equation (4).

IDF-AWE-VS + AQE-IDF-Cent: This variant uses IDF-AWE-VS to rank the documents in D′. The query representation is built using IDF-AWE, and the terms of the expansion are determined using the term-scoring function defined in Equation (1). Five variants of word embeddings are evaluated to construct the representation of the query (CBOW, Skip-grams, GloVe, FastText, and ElMo). The expanded query is evaluated in D, and its documents are ranked using IDF-AWE. The results of both rankings are consolidated in a global ranking using the re-ranking function defined in Equation (4). Another variant of this strategy that we study is to use the re-ranking function defined in Equation (5). This variant is indicated as IDF-AWE-VS + AQE-IDF-Cent$^{+}$.

The AQE strategies were studied using expansions with five terms. Top-k document retrieval on D′ was conducted using the top-10 highly ranked documents in each experimental setting. We tested many values for α, but α at 0.3 was the one with the best results. We included two baselines to carry out the evaluations. These are BM25 for the methods without expansion and BM25-AQE for the methods with query expansion. BM25-AQE uses the cosine similarity of each query term and each word embedding in the top-k documents to determine the terms of the expanded query. Once the expanded query has been evaluated, its results are ordered using BM25. We compare these results with a state-of-the-art method from the literature, the query-likelihood model with query expansion (QLM) introduced by Diaz et al. [17] and discussed in Section 3.

The methods were evaluated using Mean Average Precision (MAP@10), Recall (R@10), and NDCG (NDCG@10). The evaluation @10 is usual in the validation of AQE strategies as it is

expected that the effectiveness of the technique will be shown in the top-k documents of the ranking. The reported results correspond to averages across queries, for each corpus.

## 5.4. Results

The experimental results for WSJ and AP are shown in Table 2. MAP and recall are shown in percentages and NDCG in the interval [0,1].

The results in Table 2 show that the strategies that use IDF-AWE-VS to determine the term search space consistently perform well. By combining this technique with the IDF-AWE query representation, the results are improved in terms of MAP. The best result obtained by this combination, indicated as IDF-AWE-VS + AQE-IDF-Cent, is obtained using FastText. This result is consistent in the three corpus studied. The MAP of IDF-AWE-VS + AQE-IDF-Cent$^+$ is the best observed in each corpus, surpassing all its competitors. The margin with which IDF-AWE-VS + AQE-IDF-Cent$^+$ outperforms AQE-Cent is remarkable, exceeding 30 percentage points of MAP in various configurations. This finding shows that the IDF-AWE query vector is very useful in query expansion tasks.

Table 2. Experimental results in WSJ and AP.

| | WSJ | | | AP | | |
|---|---|---|---|---|---|---|
| Method | MAP@10 | R@10 | NDCG@10 | MAP@10 | R@10 | NDCG@10 |
| BM25 RobertsonJ76 | 21,79 | 1,58 | 0,54 | 22,67 | 1,50 | 0,59 |
| AWE-VS + CBOW | 25,59 | 1,85 | 0,65 | 23,48 | 1,68 | 0,60 |
| AWE-VS + SG | 25,87 | 2,15 | **0,70** | 23,73 | 1,90 | 0,62 |
| AWE-VS + GloVe | 21,93 | 1,46 | 0,54 | 23,31 | 1,27 | 0,50 |
| AWE-VS + FastText | 21,77 | 1,20 | 0,52 | 23,26 | 1,29 | 0,51 |
| IDF-AWE-VS + CBOW | 27,70 | 2,04 | 0,67 | 22,16 | 1,79 | 0,62 |
| IDF-AWE-VS + SG | 27,09 | 2,16 | 0,68 | 23,37 | **1,93** | **0,63** |
| IDF-AWE-VS + GloVe | 23,65 | 1,44 | 0,54 | 23,54 | 1,33 | 0,49 |
| IDF-AWE-VS + FastText | 21,31 | 1,34 | 0,54 | 20,88 | 1,32 | 0,52 |
| IDF-AWE-VS + ElMo | 38,96 | 1,82 | 0,46 | 37,95 | 1,56 | 0,43 |
| BM25 + AQE | 21,79 | 1,58 | 0,54 | 22,67 | 1,50 | 0,59 |
| QLM Diaz:16 | 38,46 | 1,86 | 0,57 | 34,32 | 1.86 | 0,59 |
| AQE-Cent + CBOW KuziSK16 | 24,42 | 1,51 | 0,55 | 26,68 | 1,55 | 0,61 |
| AQE-Cent + SG KuziSK16 | 22,52 | 1,58 | 0,59 | 24,18 | 1,66 | 0,61 |
| AQE-Cent + GloVe | 21,07 | 1,50 | 0,59 | 22,73 | 1,44 | 0,58 |
| AQE-Cent + FastText | 20,97 | 1,44 | 0,55 | 22,19 | 1,54 | 0,58 |
| AQE-Cent + ElMo | 24,42 | 1,51 | 0,55 | 23,17 | 1,63 | 0,60 |
| IDF-AWE-VS + AQE-Cent + CBOW | 49,35 | 1,67 | 0,49 | 49,59 | 1,65 | 0,43 |
| IDF-AWE-VS + AQE-Cent + SG | 58,25 | 1,86 | 0,5 | 59,72 | 1,78 | 0,43 |
| IDF-AWE-VS + AQE-Cent + GloVe | 39,30 | 1,03 | 0,29 | 62,93 | 1,01 | 0,26 |
| IDF-AWE-VS + AQE-Cent + FastText | 23,58 | 0,91 | 0,41 | 26,21 | 0,63 | 0,29 |
| IDF-AWE-VS + AQE-IDF-Cent + CBOW | 48,71 | 1,20 | 0,38 | 54,44 | 1,15 | 0,36 |
| IDF-AWE-VS + AQE-IDF-Cent + SG | 48,87 | 1,22 | 0,38 | 55,38 | 1,13 | 0,37 |
| IDF-AWE-VS + AQE-IDF-Cent + GloVe | 48,09 | 1,19 | 0,38 | 54,33 | 1,12 | 0,37 |
| IDF-AWE-VS + AQE-IDF-Cent + FastText | 49,30 | 1,22 | 0,38 | 58,65 | 1,14 | 0,37 |
| IDF-AWE-VS + AQE-IDF-Cent$^+$+ CBOW | 63,77 | 2,01 | 0,46 | 67,28 | 1,75 | 0,39 |
| IDF-AWE-VS + AQE-IDF-Cent$^+$+ SG | 64,01 | 2,00 | 0,46 | 66,79 | 1,76 | 0,40 |
| IDF-AWE-VS + AQE-IDF-Cent$^+$+ GloVe | 64,90 | 2,09 | 0,46 | 68,46 | 1,73 | 0,39 |
| IDF-AWE-VS + AQE-IDF-Cent$^+$+ FastText | **68,10** | **2,19** | 0,48 | **69,92** | 1,78 | 0,38 |
| IDF-AWE-VS + AQE-IDF-Cent$^+$+ ElMo | 64,07 | 2,09 | 0,45 | 66,40 | 1,76 | 0,38 |

Table 3 shows some highlighted words that we found interesting. These words have the particularity of identifying new terms related to the original query that expands its meaning. Some words are more specific, while others incorporate related senses. The ability of these embeddings to identify collocations strongly related to the query, such as nuclear-weapon or computer-design, is illustrated. The presence of these words is more significant in FastText than in Skip-grams (see, for example, insider-trad, environmental satellite, anti-submarine, anti-tank, among others). It is also observed that AQE-Cent identifies more specific words when using

Skip-grams, which is an effect attributable to the training of this strategy in the local corpus of TREC. Although FastText is trained in an external corpus, it manages to identify several words relevant to the original queries. Its ability to generalize in conjunction with its coding based on sub-words helps the AQE strategy.

Table 3. Query expansion terms detected using Skip-grams and FastText.

| | Skip-grams | | |
|---|---|---|---|
| qid | query | AWE (AQE-Cent) | IDF-AWE (IDF-AWE-VS + AQE-IDF-Cent$^{+}$) |
| 101 | design, star, war, anti-missil, defens, system | space-bas, sdi, antimissil, spacebas, effort-system | space-bas, sdi, missile-defens, antimissil, earlier-than-contempl |
| 102 | laser, research, applic, u.s., strateg, defens | spaced-bas, early-deploy, 200-person-strong, defense-research, far-term | spaced-bas, early-deploy, directed-energi, defense-research, sdi |
| 103 | welfar, reform | alongget, job-support, social, income-mainten, educ | alongget, social, job-support, income-mainten, educ |
| 104 | catastroph, health, insur | higher-priv, health-car, policyholdi, care, health-insur | medicar, higher-priv, catastrophic-il, health-car, catastrophic-car |
| 105 | black, monday | tuesday, wednesday, thursday, friday, today | tuesday, friday, today, white, hispan |
| 106 | u.s., control, insid, trade | also, u., public-investor, howev, although | outsid, also, although, howev, moreov |
| 107 | japanes, regul, insid, trade | public-investor, mega-play, also, howev, seijiro | public-investor, semi-insid, mega-play, seijiro, outsid |
| 108 | japanes, protectionist, measur | protection, brazilian-u.s., japan, nontariff, gatt-ord | protection, brazilian-u.s., free-trad, japan, super-301 |
| 109 | find, innov, compani | indc, corporate-awar, howev, instanc, also | idea, indc, success, corporate-awar, creativ |
| 110 | black, resist, against, south, african, govern | political-reconstruct, africa, taxi-own, udf-link, anc-link | africa, political-reconstruct, taxi-own, udf-link, pretoria |
| 111 | nuclear, prolifer | weapon, weapons-cap, warhead, weapon-fre, reactor | weapon, weapons-cap, nuclear-weapon, weapon-fre, warhead |
| 112 | fund, biotechnolog | single-sector, +107.21, technolog, invest, 10th-best-perform | technolog, amgen, pharmaceut, sunagra, repligen |
| 113 | new, space, satellit, applic | 44p, intelsat-5, 111c, reboost, orbit | 44p, orbit, payload, intelsat-5, 111c |
| 114 | non-commerci, satellit, launch | 44p, intelsat-5, earth-map, 4,580-pound, rol | intelsat-5, 44p, military-spac, direct-broadcast, military-satellit |
| 115 | impact, 1986, immigr, law | farm-wag, 377-a-year, joint-petit, immigration-reform, cuban/haitian | 377-a-year, farm-wag, joint-petit, immigration-reform, alien |
| 116 | generic, drug, substitut | macrocryst, anti-epilepsi, macrocrystallin, bio-equival, theopyllin | macrocryst, anti-epilepsi, macrocrystallin, brand-nam, bio-eq |
| 117 | capac, u.s., cellular, telephon, network | call-handl, vehicle-bas, areturn, phone, microcel | phone, vehicle-bas, call-handl, areturn, microcel |
| 118 | intern, terrorist | terror, osunu, terrorist-track, iraq-inspir, nidal | terror, nidal, terrorist-track, lebanon-bas, louzaki |
| 119 | action, against, intern, terrorist | terror, lybyan, respons, retali, contingency-measur | terror, nidal, retali, libya, bomb |
| 120 | econom, impact, intern, terror | consequ, 90-nation, mcreov, wage-set, economi | consequ, humanwav, terrorist, indc, iraq-inspir |
| | FastText | | |
| qid | query | AWE (AQE-Cent) | IDF-AWE (IDF-AWE-VS + AQE-IDF-Cent$^{+}$) |
| 101 | design, star, war, anti-missil, defens, system | bookstor, armament, staunch, singer, troop | antimissil, anti-lock, warhead, anti-submarin, anti-tank |
| 102 | laser, research, applic, u.s., strateg, defens | us, military-tech, research-and-dev, manufacturing, businessmen | computer-design, lasertech, prospect, technic, practic |
| 103 | welfar, reform | welfare-reform, welfare-to-work, price-reform, bureau, work-for | welfare-reform, welfare-to-work, work-for-welfar, bureau, reagan |
| 104 | catastroph, health, insur | health-insur, health-car, health-polici, catastrophic-il, disease-associ | fatal, disease-associ, accid, diseas, stroke |
| 105 | black, monday | black-and-whit, black-and-blu, sunday, friday, bush | black-and-whit, gray, sunday, black-and-blu, friday |
| 106 | u.s., control, insid, trade | offic, government-procur, french, prosecut, shipment | insider-trad, offic, expart-control, incent, outsid |
| 107 | japanes, regul, insid, trade | trade-regulatori, regulatori, american-control, trading-surveil, polit | grensid, insider-trad, regulatori, offic, trade-regulatori |
| 108 | japanes, protectionist, measur | protest, japanese-own, american-design, internation, american | protest, internation, american-design, assault, assail |
| 109 | find, innov, compani | discoveri, procur, u.s-develop, develop, disclos | non-innov, invis, indirect, discoveri, energ |
| 110 | black, resist, against, south, african, govern | southern, west, eastern, north, oppress | southern, west, eastern, north, northern |
| 111 | nuclear, prolifer | non-nuclear, protein, cancer, geneticist, cooper | cancer, protein, geneticist, cancer-rel, regulatori |
| 112 | fund, biotechnolog | project, swedish, nation, research, ventur | bio-technolog, technolog, biopharmaceut, biolog, pharmaceut |
| 113 | new, space, satellit, applic | promis, design, sophist, continu, real | military-satellit, environmental-satellit, spy-satellit, space-st, sat |
| 114 | non-commerci, satellit, launch | satellite-launch, relaunch, military-satellit, launcher, satellit | commerci, commercial-satellit, commerc, satellite-launch, launcher |
| 115 | impact, 1986, immigr, law | lawyer, heard, 1979, ferguson, 1948 | emigr, ancestri, countri, confus, democraci |
| 116 | generic, drug, substitut | drug-us, prescription-drug, therapeut, drugstor, medication-substitut | medication-substitut, generic-drug, prescription-drug, latest, antibiot |
| 117 | capac, u.s., cellular, telephon, network | cellular-servic, portug, consortium, servic, offic | cellular-servic, metrophon, creation, portug, so-cal |
| 118 | intern, terrorist | terrorism-watch, surrend, contemporari, couting, indign | terrorism-watch, terrorist-track, anti-terrorist, tourist, terror |
| 119 | action, against, intern, terrorist | internecin, sanction, intervent, fervent, incent | terrorism-watch, terrorist-track, anti-terrorist, sanction, vehement |
| 120 | econom, impact, intern, terror | econometr, economist, incent, extern, profit | incent, profit, economist, powerti, surrend |

## 6. CONCLUSIONS

We have introduced IDF-AWE, a query vector representation that is useful for AQE. Its effectiveness, in conjunction with FastText, shows that word embeddings are useful in AQE.

We are expanding our work to study its effectiveness using other word embeddings, such as BERT [27] or relevance-based word embeddings [25]. An alternative of particular interest is to learn a combination of weights that generates a better representation of the query. An approach based on machine learning could be beneficial in this line of research.

### ACKNOWLEDGEMENTS

Mr. Silva and Dr. Mendoza acknowledge funding support from the Millennium Institute for Foundational Research on Data. Dr Mendoza was funded by ANID PIA/APOYO AFB180002 and from ANID FONDECYT grant 1200211.

### REFERENCES

[1] G. Salton and C. Buckley, "Term-weighting approaches in automatic text retrieval," Inf. Process. Manag., vol. 24, no. 5, pp. 513–523, 1988.
[2] S. E. Robertson and K. S. Jones, "Relevance weighting of search terms", JASIS, vol. 27, no. 3, pp. 129–146, 1976.
[3] C. Carpineto and G. Romano, "A survey of automatic query expansion in information retrieval," ACM Comput. Surv., vol. 44, no. 1, pp. 1:1–1:50, 2012.
[4] H. K. Azad and A. Deepak, "Query expansion techniques for information retrieval: A survey," Inf. Process. Manag., vol. 56, no. 5, pp. 1698–1735, 2019.
[5] E. M. Voorhees, "Query expansion using lexical-semantic relations," in Proceedings of the 17th Annual International ACM-SIGIR Conference. Dublin, Ireland, 3-6 July 1994 (Special Issue of the SIGIR Forum), pp. 61–69, 1994.
[6] Y. Xu, G. J. F. Jones, and B. Wang, "Query dependent pseudo-relevance feedback based on wikipedia," in Proceedings of the 32nd Annual International ACM SIGIR Conference, SIGIR 2009, Boston, MA, USA, July 19-23, 2009, pp. 59–66, 2009.
[7] F. Diaz and D. Metzler, "Improving the estimation of relevance models using large external corpora," in SIGIR 2006: Proceedings of the 29th Annual International ACM SIGIR Conference, Seattle, Washington, USA, August 6-11, 2006, pp. 154–161, 2006.
[8] J. Hu, W. Deng, and J. Guo, "Improving retrieval performance by global analysis", in 18th International Conference on Pattern Recognition (ICPR 2006), 20-24 August 2006, Hong Kong, China, pp. 703–706, 2006.
[9] C. Carpineto, R. de Mori, G. Romano, and B. Bigi, "An information-theoretic approach to automatic query expansion," ACM Trans. Inf. Syst., vol. 19, no. 1, pp. 1–27, 2001.
[10] H. Cui, J. Wen, J. Nie, and W. Ma, "Query expansion by mining user logs," IEEE Trans. Knowl. Data Eng., vol. 15, no. 4, pp. 829–839, 2003.
[11] S. Liu, F. Liu, C. T. Yu, and W. Meng, "An effective approach to document retrieval via utilizing wordnet and recognizing phrases," in SIGIR 2004: Proceedings of the 27th Annual International ACM SIGIR Conference, Sheffield, UK, July 25-29, 2004, pp. 266–272, 2004.
[12] B. He and I. Ounis, "Combining fields for query expansion and adaptive query ex-pansion," Inf. Process. Manag., vol. 43, no. 5, pp. 1294–1307, 2007.
[13] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed repre-sentations of words and phrases and their compositionality," in Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. December 5-8, 2013, Lake Tahoe, Nevada, United States, pp. 3111–3119, 2013.
[14] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word rep-resentation," in Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, pp. 1532–1543, 2014.

[15] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching word vectors with subword information," Transactions of the Association for Computational Linguistics, vol. 5, pp. 135–146, 2017.

[16] H. Zamani and W. B. Croft, "Estimating embedding vectors for queries," in Proceedings of the 2016 ACM on International Conference on the Theory of Information Retrieval, ICTIR 2016, Newark, DE, USA, September 12- 6, 2016, pp. 123–132, 2016.

[17] F. Diaz, B. Mitra, and N. Craswell, "Query expansion with locally-trained word embeddings," in Proceedings of the 54th Annual Meeting of the Association for Com-putational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, 2016.

[18] G. Zuccon, B. Koopman, P. Bruza, and L. Azzopardi, "Integrating and evaluating neural word embeddings in information retrieval," in Proceedings of the 20th Australasian Document Computing Symposium, ADCS 2015, Parramatta, NSW, Australia, December 8-9, 2015, pp. 12:1–12:8, 2015.

[19] S. Zhai, K. Chang, R. Zhang, and Z. Zhang, "Attention based recurrent neural networks for online advertising," in Proceedings of the 25th International Conference on World Wide Web, WWW 2016, Montreal, Canada, April 11-15, 2016, Companion Volume, pp. 141–142, 2016.

[20] J. Manotumruksa, C. Macdonald, and I. Ounis, "A contextual attention recurrent architecture for context-aware venue recommendation," in the 41st International ACM SIGIR Conference, Ann Arbor, MI, USA, July 08-12, 2018, pp. 555–564, ACM,2018.

[21] S. Kuzi, A. Shtok, and O. Kurland, "Query expansion using word embeddings," in Proceedings of the 25th ACM International Conference on Information andKnowledge Management, CIKM 2016, Indianapolis, IN, USA, October 24-28, 2016, pp. 1929–1932, 2016.

[22] D. Roy, D. Ganguly, S. Bhatia, S. Bedathur, and M. Mitra, "Using word embeddings for information retrieval: How collection and term normalization choices affect performance," in Proceedings of the 27th ACM International Conference on Informa-tion and Knowledge Management, CIKM 2018, Torino, Italy, October 22-26, 2018, pp. 1835–1838, 2018.

[23] M. Almasri, C. Berrut, and J. Chevallet, "A comparison of deep learning based query expansion with pseudo-relevance feedback and mutual information," in 38th European Conference on IR Research, ECIR 2016, Padua, Italy, March 20-23, 2016. Proceedings, pp. 709–715, 2016.

[24] A. Imani, A. Vakili, A. Montazer, and A. Shakery, "Deep neural networks for query expansion using word embeddings," in 41st European Conference on IR Research, ECIR 2019, Cologne, Germany, April 14-18, 2019, Proceedings, Part II, pp. 203–210, 2019.

[25] H. Zamani and W. B. Croft, "Relevance-based word embedding," in Proceedings of the 40th International ACM SIGIR Conference, Shinjuku, Tokyo, Japan, August 7-11, 2017, pp. 505–514, 2017.

[26] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettle-moyer, "Deep contextualized word representations," in Proceedings of the NAACL-HLT 2018 conference, New Orleans, Louisiana, USA, June 1-6, 2018, pp. 2227–2237, 2018.

[27] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," in Proceedings of the NAACL-HLT2019 conference, Minneapolis, MN, USA, June 2-7, 2019, pp. 4171–4186, 2019.

## AUTHORS

**Alfredo Silva** is Informatics Engineer of the Universidad Técnica Federico Santa María, Chile. Currently he is researcher at the Millennium Institute for Foundational Research on Data. His research interests are text mining and information retrieval.

**Marcelo Mendoza** is Electronic Engineer and Master in Informatics of the Universidad Técnica Federico Santa María, Chile. He received a Ph.D. in Computer Science from the Universidad de Chile. He did a Post Doc in Yahoo Research. Currently, he is a associate professor at the Department of Informatics, Universidad Técnica Federico Santa María and head of the Master in Informatics program at the same University. He is a founder and former President of the Chilean Association of Pattern Recognition. He is a researcher at the Valparaíso Center of Science and Technology and associate researcher at the Millennium Institute for Foundational Research on Data. His research interests are text mining, information retrieval, and data mining in social networks.