Phrase2VecGLM: Neural generalized language model–based semantic tagging for complex query reformulation in medical IR

Manirupa Das[†], **Eric Fosler-Lussier**[†], **Simon Lin**[‡], **Soheil Moosavinasab**[‡], **David Chen**[‡], **Steve Rust**[‡], **Yungui Huang**[‡] & **Rajiv Ramnath**[†]
The Ohio State University[†] & Nationwide Children's Hospital[‡]
{das.65, fosler.1, ramnath.6}@osu.edu
{Simon.Lin, SeyedSoheil.Moosavinasab, David.Chen3, Steve.Rust, Yungui.Huang}@nationwidechildrens.org

Abstract

In fact-based information retrieval, state-of-the-art performance is traditionally achieved by knowledge graphs driven by knowledge bases, as they can represent facts about and capture relationships between *entities* very well. However, in domains such as medical information retrieval, where addressing specific information needs of complex queries may require understanding query intent by capturing novel associations between potentially *latent concepts*, these systems can fall short. In this work, we develop a novel, completely unsupervised, neural language model–based ranking approach for semantic tagging of documents, using the document to be tagged as a query into the model to retrieve candidate phrases from top–ranked related documents, thus associating every document with *novel related concepts* extracted from the text. For this we extend the word embedding–based generalized language model (GLM) due to (Ganguly et al., 2015), to employ phrasal embeddings, and use the semantic tags thus obtained for downstream query expansion, both directly and in feedback loop settings. Our method, evaluated using the TREC 2016 clinical decision support challenge dataset, shows statistically significant improvement not only over various baselines that use standard MeSH terms and UMLS concepts for query expansion, but also over baselines using human expert–assigned concept tags for the queries, on top of a standard Okapi BM25–based document retrieval system.

1 Introduction

Existing state-of-the-art information retrieval (IR) systems such as knowledge graphs (Su et al., 2015; Sun et al., 2015), or information extraction techniques centered around entity relationships (Ritter et al., 2013), that often rely on some form of weak supervision from ontological or knowledgebase (KB) sources, tend to perform quite reliably on fact-based information retrieval and factoid question answering tasks. However, such systems may be limited in their ability to address the complex information needs of specific types of queries (Roberts et al., 2016; Diekema et al., 2003) in domains such as clinical decision support (Luo et al., 2008) or guided product search (Teo et al., 2016; McAuley and Yang, 2016), due to: 1) complex and subjective, or lengthy nature of the query containing multiple topics, 2) vocabulary mismatch between the query expression and knowledge representations in the document collection, and 3) lack of sufficiently complete knowledge bases of "related concepts", covering *all possible relations* between candidate concepts that may exist in a collection, essential for effectively addressing these types of queries (Hendrickx et al., 2009).

We hypothesize, that similar to human experts who can determine the *aboutness* of an unseen document by recalling meaningful concepts gleaned from similar past experiences via *shared contexts*, a completely unsupervised machine learning model could be trained to associate documents within a large collection with meaningful concepts *discovered* by fully leveraging *shared contexts* within and between documents, thus surfacing "related" concepts specific to the current context (Lin and Pantel, 2002; Halpin et al., 2007; Xu et al., 2014; Kholghi et al., 2015a; Turney and Pantel, 2010; Pantel et al., 2007; Bhagat and Hovy, 2013; Hendrickx et al., 2009). As a trivial example, ordinarily unrelated concepts (noun phrases,

Proceedings of the BioNLP 2018 workshop, pages 118–128
Melbourne, Australia, July 19, 2018.

in this work) such as "scarlet macaw" and "raccoon" occurring in separate documents d_1 and d_2 may become related by a novel context such as "exotic pets" that may occur as terms in a query or as a phrase in a document d_p which could be related to both d_1 and d_2. If by some means, documents d_1 and d_2 were semantically tagged with the phrase "exotic pets" via d_p, those documents would surface in the event of such a query (Hendrickx et al., 2009; Bhagat and Ravichandran, 2008). This could thus help to better close the vocabulary gap between potential user queries and the documents. To our knowledge, ours is the first work that employs word and phrase-level embeddings for local context analysis in a pseudo-relevance feedback setting (Xu and Croft, 2000), using a *language model-based document ranking framework*, to semantically tag documents with appropriate concepts for use in downstream retrieval tasks (Kholghi et al., 2015a; De Vine et al., 2014; Sordoni et al., 2014; Zhang et al., 2016; Zuccon et al., 2015; Tuarob et al., 2013).

The main contributions of our work, are as follows: 1) We present a novel use for a neural language modeling approach that leverages shared context between documents within a collection via phrase-based embeddings (1, 2, and 3-grams), finding the right trade-off between the local context around each term versus its global context within the collection, incorporating a local context analysis-based pseudo-relevance feedback mechanism(Xu and Croft, 2000) for concept extraction. 2) Our method is fully unsupervised, i.e. it includes no outside sources of knowledge in the training, leveraging instead the *shared contexts* within the document collection itself, via word and phrasal embeddings, mimicking a human that potentially reads through the documents in the collection and uses the seen information to make relevant concept tag judgments on unseen documents. 3) Our method presents a black-box approach for tagging any corpus of documents with meaningful concepts, treating it as a closed system. Thus the concept associations can be pre-computed offline or periodically, as new documents are added to the collection and can reside outside of the document retrieval system, allowing for it to be plugged into any such system, or for the underlying retrieval system to be changed. It is also in contrast to previous approaches to document categorization for retrieval, such as those based on clustering, e.g. clustering by committee (Lin and Pantel, 2002) or semantic class induction as in (Lin and Pantel, 2001b), LDA-based topic modeling (Blei et al., 2003; Griffiths and Steyvers, 2004; Tuarob et al., 2013) and supervised or active learning approaches (Kholghi et al., 2015a) for concept extraction in information retrieval.

2 Background and Motivation

The problem of *vocabulary mismatch* in information retrieval where *semantic overlap* may exist while there is no *lexical overlap*, can be greatly alleviated by the use of query expansion (QE) techniques; whereby a query is reformulated to improve retrieval performance and obtain *additional relevant documents* by expanding the original query with additional relevant terms, and reweighting the terms in the expanded query (Xu and Croft, 2000; Rivas et al., 2014). This can also be done by learning *semantic classes* or *related candidate concepts* in the text and subsequently tagging documents or content with these semantic concept tags, that could then serve as a means for either query-document keyword matching, or for query expansion, to facilitate downstream retrieval or question answering tasks (Lin and Pantel, 2002; Xu and Croft, 2000; Lin and Pantel, 2001b; Xu et al., 2014; Bhagat and Ravichandran, 2008; Li et al., 2011; Tuarob et al., 2013; Halpin et al., 2007; Lin and Pantel, 2001a; McAuley and Yang, 2016). This is exactly the approach we adopt in order to achieve query expansion in an automated, fully unsupervised fashion, using a neural language model for local relevance feedback (Xu and Croft, 2000).

A major problem of approaches like LSA (Deerwester et al., 1990) and LDA–based topic modeling (Blei et al., 2003; Griffiths and Steyvers, 2004) is that they only consider word co-occurrences at the level of documents to model term associations, which may not always be reliable. Furthermore, these are parameterized approaches, where the number of topics K is fixed; and the final topics learnt are available as bags of words or n-grams from which topic labels must yet be inferred by an expert. In contrast, word and phrasal embeddings take into account *local co-occurrence information* of terms in the top ranked documents retrieved in response to a query (corresponding to the relevance feedback step in IR). This leads to a better modeling of query ver-

sus document term dependencies (Ganguly et al., 2015; Xu and Croft, 2000) lending itself to direct unsupervised extraction of meaningful terms related to a document, and eventually to the query.

Automatic query expansion techniques can be further categorized as either *global* or *local*. While global techniques rely on analysis of a whole collection to discover word relationships, local techniques emphasize analysis of the top-ranked documents retrieved for a query (Xu and Croft, 2000; Manning et al., 2009). Global methods include: (a) query expansion/reformulation with a thesaurus or ontology, e.g. WordNet, UMLS (b) query expansion via automatic thesaurus generation, and (c) techniques like spelling correction (Manning et al., 2009). Local methods adjust a query relative to the documents that initially appear to match the query, which is the basic idea behind our language modeling approach to semantic tagging. Basic local methods comprise: (a) relevance feedback, (b) pseudo-relevance feedback, (or blind relevance feedback), and (c) (global) indirect relevance feedback (Manning et al., 2009). *Pseudo-relevance feedback* automates the manual part of relevance feedback, so that the user gets improved retrieval performance without an extended interaction.

Here, we find an initial set of most relevant documents, then assuming that the top k ranked documents are relevant, relevance feedback is done as before under this assumption. Our proposed method tries to exactly mimic the human user behavior via pseudo-relevance feedback to semantically pre-tag documents that can later aid downstream novel retrieval for direct querying or refined querying. Thus, in our work we combine this local feedback approach with our neural language model, Phrase2VecGLM, as the query mechanism. Using a pseudo-document representation of *top-K TFIDF terms* for the document as a query into the GLM, we make novel use of Phrase2VecGLM, to semantically tag documents with phrases representative of latent concepts within those documents. This makes the collection more readily searchable by use of these tags for query expansion in downstream IR, particularly helpful in our specific use case of medical information retrieval (Luo et al., 2008; Kholghi et al., 2015b; De Vine et al., 2014; Halpin et al., 2007; Li et al., 2011; Zhang et al., 2016). Additionally, our method treats all queries in our dataset as unseen at test time, on which our actual results and gains are reported.

3 Dataset and Task

The TREC Clinical Decision Support (CDS) task track investigates techniques to evaluate biomedical literature retrieval systems for providing answers to generic clinical questions about patient cases (Roberts et al., 2016), with a goal toward making relevant biomedical information more discoverable for clinicians. For the 2016 TREC CDS challenge, actual electronic health records (EHR) of patients, in the form of case reports, typically describing a challenging medical case, as shown in Figure 1 are used. A case report is, for our purposes *a complex query* having a specific information need. There are 30 queries in the challenge dataset, corresponding to such case reports, divided into 3 topic types, at 3 levels of granularity **Note**, **Description** and **Summary** text.

The target document collection is the Open Access Subset of PubMed Central (PMC), containing 1.25 million articles consisting of $title$, $keywords$, $abstract$ and $body$ sections. In our work, we develop our query expansion method as a blackbox system using only a subset of 100K documents of the entire collection for which human judgments are made available by TREC. This allows us to derive "inferred measures" for *Normalized Discounted Cumulative Gain* (NDCG) and *Precision at 10* (P@10) scores for our evaluation (Voorhees, 2014). However, we evaluate our method on the entire collection of 1.25 million PMC articles on a separate search engine setup using an ElasticSearch (Gormley and Tong, 2015) instance, that indexes this entire set of articles on all available fields. Our unsupervised document tagging method as outlined in Section 4 employs only the $abstract$ field of the 100K PMC articles, for developing the Phrase2VecGLM language model–based document ranking subsequently used in query expansion.

4 Methodology

In our work, a *concept* is defined as a "candidate term" or "noun phrase" scored by a chosen metric e.g. top-K TFIDF, for downstream use in our algorithm (see Section 4.1 & Algorithm 1). They are used in both, training, as building blocks for unsupervised model creation by first learning a phrasal embedding space on the document collection and

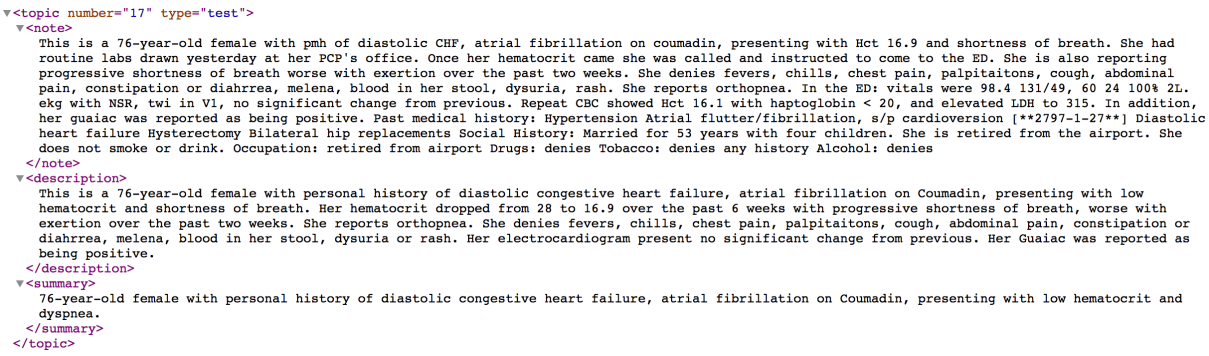

Figure 1: Sample query from the TREC 2016 challenge dataset, representing a clinical note with patient history, at **Note**, **Description** and **Summary** granularity levels.

subsequent construction of the GLM (Section 4.2), and at inference, for semantically concept–tagging documents. At the time of evaluation, concepts refer to either query terms representing a query document (Yang et al., 2009), or concept tags for target documents. Thus our concepts, predominantly noun phrases, vary from a single unigram term to consisting of up to three terms as employed by our phrase-embedding based language model (Section 4.1). Word embedding techniques use the information around the local context of each word to derive the embeddings. We therefore hypothesize that using these embeddings within a language model (LM) could help to derive terms or concepts that may be closely associated with a given document (Ganguly et al., 2015). Then further extending the model to use embeddings of candidate noun phrases, we could leverage such shared contexts for query expansion, despite no lexical overlap between the query and a given document. This could potentially help both: 1) the global context analysis for IR leading to better downstream retrieval performance from direct query expansion, and, 2) the local context analysis from top-ranked documents aiding query refinement for complex query reformulation within a relevance feedback loop (Su et al., 2015; Xu and Croft, 2000).

Thus, using our phrasal embedding based general language model, Phrase2VecGLM, described in Section 4.2 we generate top-ranked document sets for each document in the collection, treating each document as a query. We subsequently select concepts to tag query documents with, from the top-ranked documents sets for each query. We apply our language model-based concept discovery to query expansion (QE) both *directly* on the challenge dataset queries, as well as via *relevance feedback*, using the concept tags for the top-ranked documents as QE terms. We evaluate the expanded queries on a separate ElasticSearch–based search engine setup, showing improvement in both methods of query expansion (Gormley and Tong, 2015; Chen et al., 2016).

4.1 Pre-processing corpus for Phrasal GLM

We first pre-process the documents in our collection by lower-casing the text, removing most punctuation, like commas, periods, ampersands etc. keeping however, the hyphens, in order to retain hyphenated unigrams, also keeping semicolons and colons for context. We use regular expressions to retain periods that occur within a decimal value replacing these with the string *decimal* that then gets its own vector representation.

Since we implement both *unigram* and *phrasal embedding–based* GLMs, we process the same document collection accordingly, for each. For the unigram model, our tokens are single or hyphenated words in the corpus. For the phrasal model, we do an additional step of iterating through each document in the corpus, extracting the noun phrases in each using the *textblob* (Loria, 2014) toolkit. This at times gave phrases of up to a length of six, so we only admit ones of size up to three which may include some hyphenated words, to avoid tiny frequency counts. We then plug these extracted phrases back into the documents to obtain a "phrase-based corpus" for training, that has both unigrams and variable-length phrases upto 3-grams, with no tokens repeated for the n-gram processed corpus.

We then pre-compute various document and collection level statistics such as raw counts, term frequencies (phrase frequencies for phrasal cor-

pus), IDF and TF-IDF (Sparck Jones, 1972) for the terms and phrases. Following this, we proceed to generate various embedding models (Mikolov et al., 2013) for both our unigram and phrasal corpora having different length vector representations and context windows using the *gensim* (Řehůřek and Sojka, 2010) package, using the processed text. In particular we generate word embeddings trained with the skip-gram model with negative sampling (Mikolov et al., 2013) with vector length settings of 50 with a context window of 4, and also length 100 with a context window of 5. We also train with the CBOW learning model with negative sampling (Mikolov et al., 2013) for generating embeddings of length 200 with a context window of 7. But we report all of our results on experiments run off the models having an embedding length of 50. Our method is outlined in detail, in the pseudocode shown in Algorithm 1, and assumes that the document and collection statistics as well as the embedding models are already computed and available. We now describe how the processed corpus and the collection and document-level statistics are employed as building blocks to construct our phrasal embedding-based generalized language model, Phrase2VecGLM.

4.2 Phrasal Embedding-based GLM

Standard Jelinek–Mercer smoothing–based language models used for query–document matching can lead to poor probability estimation when query terms *do not* appear in the document due to a key *independence* assumption in these models, wherein query terms are sampled *independently* from *either* the document or the collection (Zhai and Lafferty, 2004). Thus given our goal of alleviating vocabulary mismatch to reformulate complex queries, we find that the word-embedding based generalized language model due to Ganguly et al. (2015), that models *term dependencies* using vector embeddings of terms, lends itself exactly for this purpose as it *relaxes* this independence assumption to incorporate term similarities via vector embeddings. This leads to better probability estimations in the event of semantic overlap between query terms and documents while no lexical overlap by proposing a generative process in which a "noisy channel" may *transform* a term t sampled from a document d or the collection C, with probabilities α and β respectively, into a query term q'. Thus, by this model we have:

$$\begin{aligned}\prod_{q' \in q} P(q'|d) = \prod_{q' \in q} [&\lambda P(q'|d) \\ &+ \alpha \sum_{t \in d} \hat{P}_{sim\_doc}(q', t|d) \\ &+ \beta \sum_{t \in d} \hat{P}_{sim\_Coll}(q', t|d) \\ &+ (1 - \lambda - \alpha - \beta) P(q'|C)]\end{aligned} \tag{1}$$

Here $P(q'|d)$ and $P(q'|C)$ are the same as direct term sampling without transformation, from either the document d or collection C, by a regular Jeliner-Mercer smoothing-based LM as in Equation (2), when $t = q'$:

$$\begin{aligned}P(d|q) &= \prod_{q' \in q} \lambda . \hat{P}(q'|d) + (1 - \lambda) . \hat{P}(q'|C) \\ &= \prod_{q' \in q} \lambda \frac{tf(q', d)}{|d|} + (1 - \lambda) . \frac{cf(q')}{|C|}\end{aligned} \tag{2}$$

However, when $t \neq q'$ we may sample the term t either from document d or collection C where the term t is *transformed* to q'. When t is sampled from d, since the probability of selecting a query term q', given the sampled term t, is *proportional* to the *similarity* of q' with t, where $sim(q', t)$ is the cosine similarity between the *vector representations* of q' and t, and $\sum(d)$ is the sum of the similarity values between *all term pairs* occurring in document d, the document term transformation probability can be estimated as:

$$\hat{P}_{sim\_doc}(q', t|d) = \frac{sim(q', t)}{\sum(d)} . \frac{tf(t, d)}{|d|} \tag{3}$$

Similarly when t is sampled from C, where for the normalization constant, instead of considering all (q', t) pairs in C, we restrict to a small neighbourhood of say 3 terms around the query term q', i.e. $N_{q'}$, to reduce the effect of noisy terms, then the collection term transformation probability can be estimated as:

$$\hat{P}_{sim\_Coll}(q', t|d) = \frac{sim(q', t)}{\sum N_{q'}} . \frac{cf(t)}{|C|} \tag{4}$$

Equation 1 combines all these term transformation events by denoting the probability of observing a query term q' without transformation (standard LM) as λ, that of document sampling–based transformation as α and the probability of collection sampling–based transformation as β.

Thus, per Equations (2) and (1), deriving the posterior probabilities $P(d|q)$ for ranking documents with respect to a query involves maximizing the conditional log likelihood of the query terms in a query q given the document d, as shown:

$$P(d|q) = -\sum_{q' \in q}[log(P(q'|d))] \quad (5)$$

We use their original word (uni-gram) embedding–based model as a *baseline* in our work. Our model, Phrase2VecGLM, further augments the original model using *variable length noun-phrases* in the vocabulary prior to learning the embedding space for the GLM. While the model by Ganguly et al, is designed as an IR matching function, we extend this model in our work to incorporate embeddings of *candidate noun phrases* from the collection, and re-purpose the model to be used as a *pseudo-relevance feedback function* to select new query expansion terms (Xu and Croft, 2000). Thus, working with the hypothesis that *concepts* in the form of "candidate noun-phrases" provide more *support for meaning*, we update the vocabulary to include noun-phrases of up to a length of three, extracted from the text. The vocabulary terms now consist of phrases, introducing more contextually meaningful terms into the set used in term similarity calculations (Equation 3). This improves concept matching, giving additional coverage toward final query term expansion via LM–based document ranking.

5 Algorithm

Our algorithm (Algorithm 1) works by intrinsically using the Phrase2VecGLM model (Section 4.2) for query expansion, to discover concepts that are similar in the shared local contexts that they occur in, within documents ranked as top-K relevant to a *query document*, and using one of two options for specified threshold criteria to tag the document, as described below. Thus our algorithm consists of two main parts: 1) A document scoring and ranking module applying directly the phrasal embeddings–based general language model described in sections 4.2, 5.1 & algorithm 1, and, 2) A concept selection module to tag the query document with, coming from the set of top ranked matching documents to a query document from step 1. There are a couple of different variations implemented for the concept selection scheme: (i) Selecting the top *TF-IDF* term from each of the top-K matching documents as the set of diverse concepts, representative of the query document, and (ii) Selecting the top-similar concept terms matching each of the representative query document terms, using word2vec/Phrase2Vec similarities on the top-ranked set of documents (Mikolov et al., 2013). The code for the corpus pre-processing, model building and inference (semantically tagging documents) is made available online [1] and the dataset is available publicly [2].

5.1 Implementation Details

In the pseudocode given by Algorithm 1, $< docStats >$ represents a set of tuples containing various pre-computed document level frequency and similarity statistics, having elements like $docTermsFreqsRawCounts$, $docTermsTFIDFs$, $docTermPairSimilaritySums$. $< collStats >$ represents a similar set for collection level frequency and similarity measures with elements like $collTermsFreqRawCountsIDFs$ and $collTermPairSimilaritySums$. The procedure also assumes available, the precomputed hashtable $dqTerms$, holding the top TF-IDF terms for each document d, used for querying into the GLM. We have excluded the implementation details for the methods $selectConceptsEmbeddingsModel$, $selectConceptsTFIDF$ and also the GLM method (which essentially computes Equations (1) and (5) for the query document to be tagged with concepts.

6 Experimental Setup

We run two different sets of experiments: (1) *Direct* query expansion of the 30 queries in the TREC dataset, using UMLS concepts (Manual, 2008) for our augmented baselines, and, (2) *Feedback loop–based* query expansion where we use the concept tags for a subset of the top returned articles for the Summary Text–based queries ran against an ElasticSearch index, as query expansion terms, (here MeSH terms-based QE (Adams and Bedrick, 2014) is an augmented baseline), and evaluate both types of runs against our ElasticSearch (ES) index setup described in Section 6.2.

[1] https://github.com/manirupa/Phrase2VecGLM
[2] http://www.trec-cds.org/2016.html#documents

Algorithm 1 Document Ranking and Concept Selection by Phrase2VecGLM

Initialize hashtables $rankedListBestMatchedDocs$, $word2vecConcepts$, $TFIDFConcepts$; ▷
These hold ranked document matches and selected concept tags for documents $d \in C$;

procedure GENERATEDOCUMENTRANKINGSCONCEPTS($queryDocs, vectorEmbeddingsModel, < docStats >, < collStats >, lambda, alpha, beta, query\_length, K$)
  for $d \in queryDocs$ **do**
    rankedListBestMatchedDocs[d] = Phrase2VecGLM($dqTerms[d], query\_length, lambda, alpha, beta, < docStats >, < collStats >$)
    word2vecConcepts[d] = selectConceptsEmbeddingsModel($dqTerms[d], < docStats > rankedListBestMatchedDocs[d], vectorEmbeddingsModel, K$)
    TFIDFConcepts[d] = selectConceptsTFIDF($dqTerms[d], < docStats >, rankedListBestMatchedDocs[d], K$)
  end for
end procedure

For direct query expansion we take all granularity levels of query topics described in Section 3, i.e. Summary, Description and Notes text, and feed these into our GLMs obtaining the top-K ranked documents for each query and drawing our query expansion concept tags from this set according to the algorithm described in Section 5. For our augmented query baselines, we use UMLS terms within the above query texts generated from the UMLS Java Metamap API that is quite effective in finding optimal phrase boundaries (Bodenreider, 2004; Chen et al., 2016).

For the relevance feedback–based query expansion, we take the top 10-15 documents returned by our ES index setup for each of the Summary Text queries and use the concept tags assigned to each of these top returned documents by our unigram and phrasal GLMs as the concept tags for query expansion for the original query. We then re-run these expanded queries through the ES search engine to record the retrieval performance. The MeSH terms used for the augmented baseline for the feedback loop case, are directly available for a majority of the PMC articles from the TREC dataset itself. Section 4.1 outlines the details of how the dataset was processed to generate the vocabulary and various elements of the GLM.

6.1 Human–Judged Query Annotation

Additionally, to evaluate our feedback loop method against a human judgments–based baseline, we use Expert Term annotations for the query topics available from a 2016 submission to TREC CDS, where 3 physicians were invited to participate in a manual query expansion experiment. Each physician was assigned 10 out of the 30 query topics from the 2016 challenge. Based on the clinical note, each physician provided a list of 2 to 4 key-phrases. The key-phrases did not have to be part of the note, but could be derived from the physician's knowledge after reading the note (Chen et al., 2016). The search keywords for the query topics thus manually provided by these *domain experts*, were used to retrieve corresponding matching PMC article IDs from the PubMed domain. The expert then spot-checked the top–ranked articles to see if these were mostly relevant. If so, they finalized the keywords assigned. Otherwise, they kept fine-tuning the keywords, until they got a desired set of results, simulating exactly the adaptive decision support (relevance feedback loop) in IR. We also develop an interpolated model with a coefficient γ that interpolates between the unigram and phrasal models, which gets performance comparable to the phrasal model, but does not outperform the other models by itself, hence we do not report those results here. Because the challenge data provides relevance judgments only on a subset of documents (which Phrase2VeGLM is trained on), we report our results using the *inferred measures* (Voorhees, 2014), for "normalized discounted cumulative gain" (NDCG) and "Precision at 10" (P@10). Although the TREC CDS 2016 query set is categorized into three topic types for Diagnosis, Tests and Treatment, we do not divide our evaluation runs into three corresponding sets, evaluating our method's perfor-

mance on the entire TREC query data set instead.

6.2 Evaluation on ElasticSearch (BM25)

For the search engine–based evaluation of our proposed method, we replicated an ElasticSearch (ES) instance setup with similar settings used in a 2016 challenge submission (Chen et al., 2016). Among the different algorithms available, BM25 (with parameters k1=3 and b=0.75) was selected as the ranking algorithm in our setup due to slightly better performance observed than others, with a logical OR querying model implemented, and the *minimum percentage match* criterion in ES, for search queries, set at 15% of the keywords matched for a document. Since our GLM outlined in Section 4.2 uses the $abstract$ field of the article for query expansion, we boosted the $abstract$ field 4 times and the $title$ field 2 times in our ES search index setup.

6.3 Results and Discussion

Table 1 outlines our results obtained with the various experimental runs described in Section 6. The hyper–parameters for our best performing models were empirically determined and set to be at $(\lambda, \alpha, \beta) = (0.2, 0.3, 0.2)$ for the word embedding–based GLM and $(\lambda, \alpha, \beta) = (0.2, 0.4, 0.2)$ for the phrasal embedding–based GLM, similar to those reported by Ganguly et al., (2015). All models were evaluated for statistical significance against the respective baselines using a two-sided Wilcoxon signed rank test, for $p << 0.01$, indicated by bold face value, if found to be significant.

As seen from the results, our unigram and phrasal GLM–based methods for query expansion appear quite promising for both direct query expansion and feedback loop based decision support. For both methods, our *trivial baseline* is the BM25 algorithm of ElasticSearch itself, that uses only the Summary text from the clinical note as the query, with no expanded set of terms.

We summarize our key findings as follows: We run two additional baselines for generation of QE terms: (i) a vanilla language model using standard Jelinek-Mercer smoothing, equivalent to Phrase2VecGLM with settings $(\lambda, \alpha, \beta) = (0.5, 0.0, 0.0)$ such that the embedding space is *not* used to derive term similarities, and (ii) the standard Phrase2vec embedding space model itself (De Vine et al., 2014) prior to deriving the GLM. Both these baselines actually perform worse than the trivial BM25 baseline for QE on the Summary text, in both direct and relevance feedback settings.

For direct query expansion, UMLS concepts found within the Summary, Description and Notes text of the query itself, were used as augmented baselines. Of these, the Notes UMLS–based expansion worked rather poorly (we attribute this to extra noise concepts in the lengthy Notes text). Though Description text–based UMLS terms did worse than our vanilla Summary text baseline, the Description UMLS terms run through the unigram GLM to get expanded terms did significantly better than Description UMLS terms indicating that our method helps improve term expansion. For direct query expansion, the biggest gain against the baseline was observed for the Summary text UMLS terms run through the unigram GLM to get expanded terms, with a P@10 value of 0.2817. The phrasal model did comparably to the unigram model, however did not beat it, for the direct setting of query expansion.

For the feedback loop based query expansion method, we had two separate human judgment–based baselines, one using the MeSH terms available from PMC for the top 15 documents returned in a first round of querying the ES index with Summary text, and the other based on the expert annotations of the 30 query topics as described in Section 6. The MeSH terms baseline got a P@10 of 0.2294, even less than our vanilla Summary Text baseline with no expanded terms, while our Expert Terms baseline beat this baseline significantly. One reason for the lower performance of the MeSH terms model, we believe, is lack of MeSH term coverage for all the documents chosen. Our unigram GLM–based expanded terms from the top–15 documents returned by Summary Text beat the Expert Terms baseline quite significantly with P@10 of **0.2792**. This was outperformed by the phrasal GLM–based expanded terms model with P@10 of **0.2872**.

Finally our combined model using the unigram + phrasal GLM terms from the top–15 off of the Summary text, beat our vanilla baseline, and was outperformed by our very best combined terms model which generated unigram + phrasal GLM–based terms for the top–15 documents for each query, off of the **Summary + Summary UMLS concepts**, getting a P@10 of **0.3091**. As an example to illustrate, a set of concept tags learned by our unigramGLM model may look like:

Query Expansion Method	Metric		
	Query Text	NDCG **	P@10 **
Direct setting:			
BM25+Standard LM (Jelinek-Mercer sm.) QE Terms (**baseline**)	Summary	0.0475	0.1172
BM25+Phrase2Vec (without GLM) QE Terms (**baseline**)	Summary	0.0932	**0.2267**
BM25+DescUMLS QE Terms (**augmented baseline**)	Summary	0.1070	**0.2299**
BM25+DescUMLS+unigramGLM QE Terms (**model**)	Summary	0.1010	**0.2414**
BM25+None (**baseline**)	Summary	0.1060	**0.2489**
BM25+SumUMLS QE Terms (**augmented baseline**)	Summary	0.1466	**0.2644**
BM25+SumUMLS+unigramGLM QE Terms (model)	Summary	0.1387	**0.2817**
Feedback Loop setting:			
BM25+Standard LM (Jelinek-Mercer sm.) QE Terms (**baseline**)	Summary	0.0265	0.0867
BM25+Phrase2Vec (without GLM) QE Terms (**baseline**)	Summary	0.0662	**0.1318**
BM25+MeSH QE Terms (**baseline**)	Summary	0.0970	**0.2294**
BM25+None (**baseline**)	Summary	0.1060	**0.2489**
BM25+**Human Expert** QE Terms (**augmented baseline**)	Summary	0.1029	**0.2511**
BM25+unigramGLM QE Terms (**model**)	Summary	0.1173	**0.2792 ***
BM25+Phrase2VecGLM QE Terms (model)	Summary	0.1159	**0.2872 ***
Feedback Loop Combined Models			
BM25+unigramGLM Terms+Phrase2VeclGLM Terms (**baseline**)	Summary	0.1057	0.2756
BM25+SumUMLS+unigramGLM Terms+Phrase2VecGLM QE Terms (model)	Summary	**0.1206**	**0.3091 ***

Table 1: Results for IR after Query Expansion (QE) by different methods using unigram and phrasal GLM–generated QE terms, in **direct** and **feedback loop** settings. Bold face values indicate statistical significance at $p << 0.01$ over the previous result or baseline. Single asterisks indicate our best performing models. Double asterisks indicate *inferred* measures (Voorhees, 2014). Numbers are from evaluation of ranking results based on document relevance judgments available for all 30 queries in the dataset.

*<**'query_doc'**:(4315343, ['dementia', 'cognitive', 'bp']), **'concept_tags'**: ['alzheimers', 'diabetes', 'behavioral'] >*, and for the phrasalGLM model we may have: *<**'query_doc'**:(3088738, ['albendazole', 'eosinophilic ascites', 'parasitic infection']), **'concept_tags'**: ['corticosteroid therapy', 'case hyperinfection', 'strongyloides stercoralis'] >*.

7 Conclusions and Future Work

In this work, we demonstrate that our proposed method of semantic tagging for query expansion, via word and phrasal GLM–based document ranking for pseudo-relevance feedback, can prove an effective means to serve complex, specific information needs such as clinical queries in medical information retrieval that require adaptive decision support, performing better in some cases than even human expert–provided query expansion terms. This is especially helpful to solve the problem of *lack of keyword coverage* for all documents in any collection, e.g. MeSH terms for PMC articles. In future we hope to leverage end-to-end recurrent neural architectures such as LSTMs, possibly with attention mechanisms (Rocktäschel et al., 2015; Bahdanau et al., 2014) to improve our current method of semantic tagging for complex querying in medical IR.

Acknowledgments

The authors would like to thank Alan Ritter, whose invaluable feedback helped to significantly improve portions of evaluation and presentation of this work, and our collaborators at Nationwide Children's Hospital whose valuable time, support and resources made this work possible. We also thank our anonymous reviewers for their feedback.

References

Joel Adams and Steven Bedrick. 2014. Automatic classification of pubmed abstracts with latent semantic indexing: Working notes. In *CLEF (Working Notes)*, pages 1275–1282. Citeseer.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Rahul Bhagat and Eduard Hovy. 2013. What is a paraphrase? *Computational Linguistics*, 39(3):463–472.

Rahul Bhagat and Deepak Ravichandran. 2008. Large scale acquisition of paraphrases for learning surface patterns. In *ACL*, volume 8, pages 674–682.

David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022.

Olivier Bodenreider. 2004. The unified medical language system (umls): integrating biomedical terminology. *Nucleic acids research*, 32(suppl_1):D267–D270.

Wei Chen, Soheil Moosavinasab, Anna Zemke, Ariana Prinzbach, Steve Rust, Yungui Huang, and Simon Lin. 2016. Evaluation of a machine learning method to rank pubmed central articles for clinical relevancy: Nch at trec 2016 cds. *TREC 2016 Clinical Decision Support Track*.

Lance De Vine, Guido Zuccon, Bevan Koopman, Laurianne Sitbon, and Peter Bruza. 2014. Medical semantic similarity with a neural language model. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*, pages 1819–1822. ACM.

Scott Deerwester, Susan T Dumais, George W Furnas, Thomas K Landauer, and Richard Harshman. 1990. Indexing by latent semantic analysis. *Journal of the American society for information science*, 41(6):391.

Anne Diekema, Ozgur Yilmazel, Jiangping Chen, Sarah Harwell, Lan He, and Elizabeth D Liddy. 2003. What do you mean? finding answers to complex questions. In *New Directions in Question Answering*, pages 87–93.

Debasis Ganguly, Dwaipayan Roy, Mandar Mitra, and Gareth JF Jones. 2015. A word embedding based generalized language model for information retrieval. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 795–798. ACM.

Clinton Gormley and Zachary Tong. 2015. *Elasticsearch: The Definitive Guide*. " O'Reilly Media, Inc.".

Thomas L Griffiths and Mark Steyvers. 2004. Finding scientific topics. *Proceedings of the National academy of Sciences*, 101(suppl 1):5228–5235.

Harry Halpin, Valentin Robu, and Hana Shepherd. 2007. The complex dynamics of collaborative tagging. In *Proceedings of the 16th International Conference on World Wide Web*, pages 211–220. ACM.

Iris Hendrickx, Su Nam Kim, Zornitsa Kozareva, Preslav Nakov, Diarmuid Ó Séaghdha, Sebastian Padó, Marco Pennacchiotti, Lorenza Romano, and Stan Szpakowicz. 2009. Semeval-2010 task 8: Multi-way classification of semantic relations between pairs of nominals. In *Proceedings of the Workshop on Semantic Evaluations: Recent Achievements and Future Directions*, pages 94–99. Association for Computational Linguistics.

Mahnoosh Kholghi, Laurianne Sitbon, Guido Zuccon, and Anthony Nguyen. 2015a. Active learning: a step towards automating medical concept extraction. *Journal of the American Medical Informatics Association*, 23(2):289–296.

Mahnoosh Kholghi, Laurianne Sitbon, Guido Zuccon, and Anthony Nguyen. 2015b. Active learning: a step towards automating medical concept extraction. *Journal of the American Medical Informatics Association*, 23(2):289–296.

Chenliang Li, Anwitaman Datta, and Aixin Sun. 2011. Semantic tag recommendation using concept model. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, pages 1159–1160. ACM.

Dekang Lin and Patrick Pantel. 2001a. Discovery of inference rules for question-answering. *Natural Language Engineering*, 7(4):343–360.

Dekang Lin and Patrick Pantel. 2001b. Induction of semantic classes from natural language text. In *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 317–322. ACM.

Dekang Lin and Patrick Pantel. 2002. Concept discovery from text. In *Proceedings of the 19th International Conference on Computational Linguistics-Volume 1*, pages 1–7. Association for Computational Linguistics.

Steven Loria. 2014. Textblob: simplified text processing. *Secondary TextBlob: Simplified Text Processing*.

Gang Luo, Chunqiang Tang, Hao Yang, and Xing Wei. 2008. Medsearch: a specialized search engine for medical information retrieval. In *Proceedings of the 17th ACM conference on Information and knowledge management*, pages 143–152. ACM.

Christopher D Manning, Prabhakar Raghavan, and Hinrich Schütze. 2009. Introduction to information retrieval. *An Introduction To Information Retrieval*, 151(177):5.

NLM UMLS Knowledge Sources Manual. 2008. National library of medicine. *Bethesda, Maryland*.

Julian McAuley and Alex Yang. 2016. Addressing complex and subjective product-related queries with customer reviews. In *Proceedings of the 25th International Conference on World Wide Web*, pages 625–635. International World Wide Web Conferences Steering Committee.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, pages 3111–3119.

Patrick Pantel, Rahul Bhagat, Bonaventura Coppola, Timothy Chklovski, and Eduard H Hovy. 2007. Isp: Learning inferential selectional preferences. In *HLT-NAACL*, pages 564–571.

Radim Řehůřek and Petr Sojka. 2010. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta. ELRA. http://is.muni.cz/publication/884893/en.

Alan Ritter, Luke Zettlemoyer, Oren Etzioni, et al. 2013. Modeling missing data in distant supervision for information extraction. *Transactions of the Association for Computational Linguistics*, 1:367–378.

Andreia Rodrıguez Rivas, Eva Lorenzo Iglesias, and L Borrajo. 2014. Study of query expansion techniques and their application in the biomedical information retrieval. *The Scientific World Journal*, 2014.

Kirk Roberts, Matthew S Simpson, Ellen M Voorhees, and William R Hersh. 2016. Overview of the trec 2015 clinical decision support track. In *TREC*.

Tim Rocktäschel, Edward Grefenstette, Karl Moritz Hermann, Tomáš Kočiskỳ, and Phil Blunsom. 2015. Reasoning about entailment with neural attention. *arXiv preprint arXiv:1509.06664*.

Alessandro Sordoni, Yoshua Bengio, and Jian-Yun Nie. 2014. Learning concept embeddings for query expansion by quantum entropy minimization. In *AAAI*, volume 14, pages 1586–1592.

Karen Sparck Jones. 1972. A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation*, 28(1):11–21.

Yu Su, Shengqi Yang, Huan Sun, Mudhakar Srivatsa, Sue Kase, Michelle Vanni, and Xifeng Yan. 2015. Exploiting relevance feedback in knowledge graph search. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1135–1144. ACM.

Huan Sun, Hao Ma, Wen-tau Yih, Chen-Tse Tsai, Jingjing Liu, and Ming-Wei Chang. 2015. Open domain question answering via semantic enrichment. In *Proceedings of the 24th International Conference on World Wide Web*, pages 1045–1055. ACM.

Choon Hui Teo, Houssam Nassif, Daniel Hill, Sriram Srinivasan, Mitchell Goodman, Vijai Mohan, and SVN Vishwanathan. 2016. Adaptive, personalized diversity for visual discovery. In *Proceedings of the 10th ACM Conference on Recommender Systems*, pages 35–38. ACM.

Suppawong Tuarob, Line C Pouchard, and C Lee Giles. 2013. Automatic tag recommendation for metadata annotation using probabilistic topic modeling. In *Proceedings of the 13th ACM/IEEE-CS joint conference on Digital libraries*, pages 239–248. ACM.

Peter D Turney and Patrick Pantel. 2010. From frequency to meaning: Vector space models of semantics. *Journal of artificial intelligence research*, 37:141–188.

Ellen M Voorhees. 2014. The effect of sampling strategy on inferred measures. In *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval*, pages 1119–1122. ACM.

Jinxi Xu and W Bruce Croft. 2000. Improving the effectiveness of information retrieval with local context analysis. *ACM Transactions on Information Systems (TOIS)*, 18(1):79–112.

Wei Xu, Alan Ritter, Chris Callison-Burch, William B Dolan, and Yangfeng Ji. 2014. Extracting lexically divergent paraphrases from twitter. *Transactions of the Association for Computational Linguistics*, 2:435–448.

Yin Yang, Nilesh Bansal, Wisam Dakka, Panagiotis Ipeirotis, Nick Koudas, and Dimitris Papadias. 2009. Query by document. In *Proceedings of the Second ACM International Conference on Web Search and Data Mining*, pages 34–43. ACM.

Chengxiang Zhai and John Lafferty. 2004. A study of smoothing methods for language models applied to information retrieval. *ACM Transactions on Information Systems (TOIS)*, 22(2):179–214.

Ye Zhang, Md Mustafizur Rahman, Alex Braylan, Brandon Dang, Heng-Lu Chang, Henna Kim, Quinten McNamara, Aaron Angert, Edward Banner, Vivek Khetan, et al. 2016. Neural information retrieval: A literature review. *arXiv preprint arXiv:1611.06792*.

Guido Zuccon, Bevan Koopman, Peter Bruza, and Leif Azzopardi. 2015. Integrating and evaluating neural word embeddings in information retrieval. In *Proceedings of the 20th Australasian Document Computing Symposium*, page 12. ACM.