

# Learning2extract for Medical Domain Retrieval

Yue Wang, Kuang Lu, and Hui Fang

Department of Electrical & Computer Engineering,  
University of Delaware, USA  
{wangyue, lukuang, hfang}@udel.edu

**Abstract.** Search is important in medical domain. For example, physicians need to search for literature to support their decisions when they diagnose the patients, especially for the complicated cases. Even though they could manually input the queries, it is not an easy task because queries are expected to include enough information about the patients. Therefore, the queries tend to be verbose. However, those verbose queries may not work well since the search engine would favor documents covering every term in the query, but not the ones which are truly important. Existing work on verbose query processing in Web search has studied the similar problem, but the methods are not applicable to the medical domain because of the complexity of the medical queries and the lack of domain-specific features. In this work, we propose a set of new features to capture the importance of the terms which are helpful for medical retrieval, i.e., **Key Terms**, from verbose queries. Experiment results on the TREC Clinical Decision Support collections show that the improvement of using the selected Key Terms over the baseline methods is statistically significant.

## 1 Introduction

Medical records contain valuable resources, such as the diagnoses and treatments, for the patients. In recent years, the growing usage of Electronic Medical Records (EMR) makes it possible for the physicians to access this valuable resource. One notable search scenario is, before the physicians make the clinical decisions, they need to browse previous medical records and literature that are similar to the situation of the current patient in order to ensure the accuracy of the diagnose, test, or treatment they would provide to the patient, especially for the complicated cases. Although the physician can manually enter the queries, these queries often need to be deliberated to ensure the search quality, since there are lots of detailed information about the patient, and it is not straightforward to identify which information should or should not to be included in the search query. Table 1 shows an example of how queries are formulated based on the EMR. Since the current search engines assume the queries are composed by key words, the documents that cover more query terms would be favored. However, from the example in Table 1, it is clear that not every term in the EMR is equally important. Returning the documents containing fewer important terms would not be helpful for the physicians. Thus, useful terms selection from the

EMR becomes an essential but challenging task, even for the physicians with extensive medical knowledge.

Query Type	Query Content
<b>The EMR of the patient</b>	78 M w/ pmh of CABG in early [**Month (only) 3**] at [**Hospital6 4406**] (transferred to nursing home for rehab on [**12-8**] after several falls out of bed.) He was then readmitted to [**Hospital6 1749**] on [**3120-12-11**] after developing acute pulmonary edema/CHF/unresponsiveness?. There was a question whether he had a small MI; he reportedly had a small NQWMI. He improved with diuresis and was not intubated. . Yesterday, he was noted to have a melanotic stool earlier this evening and then approximately 9 loose BM w/ some melena and some frank blood just prior to transfer, unclear quantity.
<b>A shorter version of the EMR</b>	78 M transferred to nursing home for rehab after CABG. Reportedly readmitted with a small NQWMI. Yesterday, he was noted to have a melanotic stool and then today he had approximately 9 loose BM w/ some melena and some frank blood just prior to transfer, unclear quantity.
<b>Simplified query</b>	A 78 year old male presents with frequent stools and melena.

**Table 1.** An example showing how queries are formulated based on the EMR

This problem is similar to verbose query processing in the Web search. Although the information retrieval with verbose query has been studied, existing methods are not applicable in the medical domain for two reasons. On one hand, existing work considered the queries with 5 or more terms as verbose queries [1], but the queries in medical domain are much longer and much more complicated than the Web queries. This can be clearly observed from the example in Table 1 as the simplified query still contains 11 terms. On the other hand, the features selected for the web queries may not work for the medical domain. For instance, the features which require the query logs of the general search engine, such as query log frequency[2] and similarity with previous queries[3], etc, can not be directly used in the medical domain because of lack of history of the verbose queries.

In this paper, to overcome the comprehensive requirement of the medical related knowledge for simplifying the verbose queries, we propose an automatic way to extract useful information from the verbose queries in the medical domain. Specially, we designed a set of features which could be helpful for identifying the Key Terms in the verbose queries. We then applied state-of-art machine learning techniques with the proposed features to select the terms used for retrieval. The experimental results over the TREC CDS collections showed that the proposed features could improve the performance.

## 2 Related Work

We define our work as key terms identification in medical domain. There are several research areas that are related to our work.

**Clinical Query Generation.** Soldaini et al. studied the query reduction techniques for searching medical literature [4]. They followed the structure proposed in [5], which takes quality predictors as features to rank the sub-queries of the original query using SVM. In addition, they also studied how to utilize query expansion technique with the query reduction. However, they did not report the performance of their method on the verbose queries of TREC CDS collection: although they also used the public available CDS data collection, they created their own query set to test the performance.

**Keyphrase Extraction.** The concept of identifying useful information from verbose query was introduced by Turney[6]. After that, considerable amount of works have been done in this area[5, 1, 3]. Bendersky and Croft proposed to identify key concepts from given verbose queries using a set of features[2]. They considered the key concepts in a verbose query as a special form of sub-queries, and then proposed to use the machine learning methods to predict the usefulness of all the sub-queries. The experiments are conducted using the standard TREC collections (Robust04, W10G, and Gov2). Our work is similar to theirs, however, the differences are also clear. On one hand, our verbose query are much longer than the ones used in their experiments. On the other hand, we focus on the medical domain, so we would also like to explore how the domain specific features could be used in our experiments.

**Medical Domain Retrieval.** Bio-medical domain retrieval has received more and more attentions in the recent years. Existing work could be divided into two directions based on how the documents are represented, i.e., term based representation and concept based representation. The term based representation adopted the traditional bag-of-term assumption which consider each term independently. They then applied other techniques, such as query expansion with domain resources [7, 8], semantic similarity of the documents and the corresponding pseudo relevance feedback set[9], or a combination of different retrieval models[10] and types of documents [11] to improve the retrieval performance.

Concept-based representation assumes the documents are composed by concepts. It relies on specific NLP toolkits, such as MetaMap or cTAKES, to identify the concepts from the raw documents and then apply the existing retrieval methods[12–15]. Wang and Fang showed that the results from the NLP toolkit could generate less satisfied results because the concepts from the same aspect are related, and the one-to-many mapping from the MetaMap could inflate the weights of some query aspects[16]. In order to solve this problem, they proposed two weighting regulations to the existing retrieval models. Despite the different representation methods, the queries used in these work are the simplified version of the EMR, which is different from our problem since we focus on how to select the important terms from the verbose query in medical domain.

Type	Feature	Description
Domain	$Concept(t_i)$	whether $t_i$ is part of a medical concept
Features	$Unique(t_i)$	the ratio of the IDF value of $t_i$ in medical and web domain
	$Abbr(t_i)$	whether $t_i$ is an abbreviation
	$All\_Cap(t_i)$	whether $t_i$ only contains capital letters
Lexicon	$Capitalized(t_i)$	whether $t_i$ contains any capital letters
Features	$Stop(t_i)$	whether $t_i$ is a stopword
	$Numeric(t_i)$	whether $t_i$ is a number
	$Noun(t_i)$	whether $t_i$ is a noun, or part of a noun phrase
POS	$Verb(t_i)$	whether $t_i$ is a verb, or part of a verb phrase
Features	$Adj(t_i)$	whether $t_i$ is an adjective
	$tf_{des}(t_i)$	the term frequency in description of $t_i$
Statistical	$tf_c(t_i)$	the term frequency in collection of $t_i$
Features	$IDF(t_i)$	the invert document frequency $t_i$
	$wig(t_i)$	the weighted information gain of $t_i$ (Proposed in [2])
Locality	$Rank_{des}(t_i)$	the position of $t_i$ shown in the description
Features	$Rank_{sent}(t_i)$	the position of sentence that contains $t_i$ shown in the description

**Table 2.** Features used for identifying Key Terms

### 3 Methods

We define the terms that could be helpful for retrieving relevant documents in medical domain as **Key Terms**. The goal of our research is to identify those Key Terms from the verbose queries. We formulate this problem as a classification problem. Formally, the input of our system is the query  $\mathbb{Q}$  which contains  $n$  terms, i.e.,  $\mathbb{Q} = (t_1, t_2, \dots, t_n)$ . The classification problem is then to infer a Key Term label for each term, i.e., for each term  $t_i$ , to classify whether it is a Key Term. The Key Terms would be kept and then used for retrieval propose.

Since we model this problem as a classification problem, the feature selection is the key component to success. In this work, we propose several new features for this domain specific problem, as well as adopt some features from existing work[2]. The list of all the features is included in the Table 2. Due to the limited space, we will only introduce the important ones in the following discussion.

The terms tend to be important if they are related to the medical domain, thus, we would like to keep a term if it is from a medical related concepts. For instance, the term “*disorder*” is common in the medical domain, so it may not be selected when extracting the Key Terms. However, this term is certainly important if it shows in the phrase “*post-traumatic stress disorder*”. We used the **Concept**( $t_i$ ) to capture this feature. Specifically, we used MetaMap<sup>1</sup> to identify the medical related concepts from the queries. We will set this feature to true if the term is part of the identified medical concepts.

In addition, the term is also important if it is unique in the medical domain. For example, the word “*vitamin*” is not a common word in the web domain, but it occurs many times in the medical domain. This phenomenon indicates

<sup>1</sup> <https://metamap.nlm.nih.gov/>

that the terms are more useful in the medical domain. In order to capture this feature, we computed the IDF value of the term in the CDS collection and the one in a regular web domain (i.e., a TREC Web collection). The ratio of these two IDF values is used as the feature. This feature is denoted as **Unique**( $t_i$ ).

Abbreviations are widely used in the medical domain, especially to stand for the names of a disease, such as “*PTSD*” and “*UTIs*”, or a diagnostic procedure (“*MRI*”, for example). Correctly locate those medical related abbreviations could improve the retrieval performance. Therefore, we proposed the **Abbr**( $t_i$ ) feature to capture this phenomenon. Due to the lack of a comprehensive abbreviation dictionary in medical domain, we used two online dictionaries, i.e., Oxford online dictionary<sup>2</sup> and Merriam-Webster<sup>3</sup>, to identify if a term is an abbreviation. This feature will be set to true if the term does not show in any of these two online dictionaries as a English term. Some abbreviations, e.g., “*COLD*”, which stands for “*chronic obstructive lung disease*”, may happen to be a English term, so it can not be captured by the previous feature. We propose to include those terms by using the **All\_Cap**( $t_i$ ) feature. The feature would be true if the every character in the term is capitalized. Similarly, capitalization could also be an indicator of the domain specific terms or proper nouns. We designed the feature **Capitalized**( $t_i$ ) to capture that. This value would be set to true if the term is capitalized.

Ideally, the key concepts can be captured by the nouns and verbs in the sentence. Therefore, we proposed to include the POS tagging as one set of the features. Specifically, there are three features belong to this category, i.e., **Noun**( $t_i$ ): whether a term is a noun (or part of the noun phrases), **Verb**( $t_i$ ): whether it is a verb (or part of the verb phrases), and **Adj**( $t_i$ ): whether it is an adjective.

One straightforward way to measure the effectiveness of the terms is to compute how the retrieval performance will change with and without the term. We adopt the weighted information gain (**wig**( $t_i$ )) from [2] to capture this phenomenon. The **wig**( $t_i$ ) is defined as the changes of information from the state of which only average document is retrieved to the state of which the term  $t_i$  is actually being used as the query term. It has been shown to be effective in the verbose query processing in web search. We adopt the same equation when computing the wig as in [2], i.e.:

$$wig(t_i) = \frac{\frac{1}{M} \sum_{d \in T_M(t_i)} \log p(t_i|d) - \log p(t_i|\mathcal{C})}{-\log p(t_i|\mathcal{C})} \quad (1)$$

where the  $T_M(t_i)$  is the top returned documents when using term  $t_i$ . We set  $M$  to 50 in our experiment.

By observing the shorter version of the EMR and the simplified query in Table 1, we can see that the important information, such as the previous medical history and chief complaint are introduced at the beginning of the EMR. Therefore, we proposed **Rank\_des**( $t_i$ ) and **Rank\_sent**( $t_i$ ) to capture the locality information of the terms.

<sup>2</sup> <https://www.oxforddictionaries.com/>

<sup>3</sup> <https://www.merriam-webster.com/>

## 4 Experiments

### 4.1 Data Sets

In order to perform the machine learning techniques with the proposed features, a data set with importance of the query terms is desired. However, such data set is hard to obtain. We utilized the data sets from TREC Clinical Decision Support track to approximate that, since they contain different versions of the same query. Clinical Decision Support (CDS) track has been held from 2014 to 2016 in TREC. 30 queries are released for each year’s task. The example query shown in Table 1 is a query released in 2016. There are three types of queries: The EMR of the patient is the admission note from MIMIC-III, which describes the patient’s chief complaint, medical history and other useful information upon admission. This is named as *note* query. The shorter version of EMR and the simplified query are the narratives generated by the organizers based on the *note* query. They are named as *description* and *summary* queries, respectively. By observing the example, it is clear that the *note* query and *description* query are much more verbose than the *summary* query. The *description* query and *summary* query are provided for all three years, while the *note* query is only provided for CDS16. The average query length (number of terms) is reported in Table 3. We can see that even the shortest version of the query, i.e., *summary* query, is still much longer than the verbose query in web domain.

Year	Summary	Description	Note
CDS14	26.97	79.53	–
CDS15	21.87	83.97	–
CDS16	34.4	123.1	248.9

**Table 3.** Average query length (number of terms) for different query types.

Because the *note* query is not available for all years, we used the *description* and *summary* queries to train the classifier. When generating the training set, for each term occurred in the *description* query, we would consider it as an important term if and only if this term also occurs in the corresponding *summary* query. Term stemming is not used, and the stop words are not removed from the query. The same term shown in different queries is kept because although the term spelling is the same, the feature vectors it generated could be different ( $tf_{des}(t_i)$ , for instance).

Although the *summary* query is not a strict subset of the *description* query, we argue that the CDS track query set fit this problem setup well for three reasons: First, both the *description* query and the *summary* query are manually created by expert topic developers. Therefore, the quality of the topics is guaranteed. Second, there are 90 queries are given for three years, which generated more than 1000 mapping instances, which is a reasonable size for a training set.

Finally, TREC CDS track is a platform for comparing different retrieval models from participants. Therefore, we could compare our performance with the state-of-art runs in this domain.

## 4.2 Key Terms Identification Results

We tested different machine learning algorithms with the selected features as described in the Section 3 using the description queries and summary queries. All 90 queries across three data collections were used, and 5-fold cross validation is applied. The precision, recall and F1 of Key Term identification are reported in the Table 4.

	<b>Random Forest</b>	<b>Logistic Regression</b>	<b>Decision Tree</b>	<b>SVM</b>
<b>Precision</b>	0.753	<b>0.795</b>	0.642	0.735
<b>Recall</b>	0.631	0.676	<b>0.821</b>	0.668
<b>F1</b>	0.686	<b>0.731</b>	0.720	0.699

**Table 4.** Performance of Key Term selection.

It is clear that Logistic regression and Decision Tree perform better than the other two methods in terms of F1, which indicates that both of these two methods could be useful on identifying the Key Terms. However, since only the Key Terms is what we want to extract from the verbose query, the precision of the identification has a higher priority than recall. Therefore, we chose the logistic regression as our identification models in the following experiments.

## 4.3 Apply Identified Key Terms for Retrieval

The ultimate goal of this project is to improve the retrieval performance using the selected terms. Therefore, we conducted the experiments using the selected Key Terms. To be specific, we trained the classifier using the *description* query and *summary* query from two data collections, and tested it using the third year’s data collection. We did the experiment three times by switching the training data. The results, in terms of infNDCG, are reported in Table 5. We used the Indri with the default Dirichlet smoothing as the retrieval function. The parameter  $\mu$  is tuned to achieve the best performance.

We included several state-of-art methods as baselines. Noun Phrase is a baseline method described in [17], which only the noun phrases from the description are kept as the query. The Key Concept is a state-of-art baseline method proposed in [2]. Similar as our work, they proposed a set of features to identify the key concepts from the web query. Three features, i.e.,  $g_t f(c_i)$ ,  $qp(c_i)$  and  $qe(c_i)$ , are dropped because exterior resources are required to generate these features, while we don’t have such access to those resources. Fast QQP is the best query

reduction method proposed by Soldaini et al. in [4]. We trained the classifier based on the features described in their paper with the TREC CDS query set, since the query set used in their project is not public available. In addition, we also included the performance of using the *description* query and *summary* query directly, named Description and Summary in the table. The *Summary* baseline could serve as a upper bond for our method since our method is trained against it.

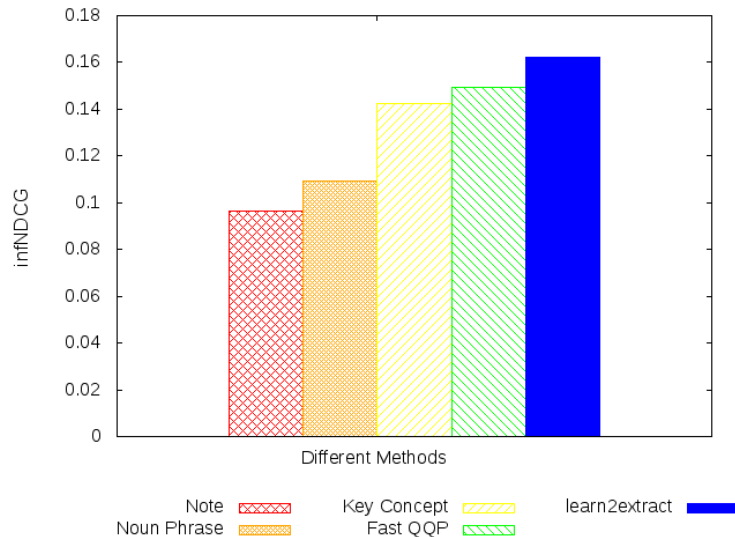
	<b>CDS14</b>	<b>CDS15</b>	<b>CDS16</b>
Summary	0.1712	0.2067	0.1844
Description	0.1397	0.1615	0.1537
Noun Phrase[17]	0.1195	0.1487	0.1322
Key Concept[2]	0.1426	0.1657	0.1594
Fast QQP[4]	0.1498	0.1753	0.1584
learning2extract	<b>0.1583<sup>†</sup></b>	<b>0.1779<sup>†</sup></b>	<b>0.1647</b>

**Table 5.** Retrieval performance using Key Terms selected from *description* query. The <sup>†</sup> indicates the improvement over *Description* is statistically significant at 0.05 level based on Wilcoxon signed-rank test.

By comparing the performance in Table 5, we could see that the improvement of our method is significant over the baseline that directly using the *description* query. Our method also outperforms the three baselines, which means that using the domain specific features would be helpful in the problem setup. Note that the Key Concept and Fast QQP methods are not trained as reported in the original paper, so it could be possible that their performances would be improved. The performance of the proposed method is actually close to the results of directly using the summary query. This suggests that our method could successfully identify the useful terms from the verbose query.

In addition to the *description* query, we also tested our method with the *note* query. The *note* query is only available in 2016 data collection. Since only limited training data is available, we used the *description* query from the three year to build the classifier. Figure 1 summarizes the performance of the runs. The performance is reported in terms of infNDCG. The results show that, not surprisingly, the best tuned performance of Key Term selection using the *note* queries is worse than the ones trained on using the description queries for all methods. There could be two reasons, for one thing, the rich information contained in the note query tend to generate more redundant terms which could hurt the performance. For the other thing, the lacking of training example from note query to summary query is also a reason for the performance decrease. However, our method could still improve the performance over note query and outperform the other baseline methods, which shows that the proposed feature is robust on identifying Key Terms from verbose queries.





**Fig. 1.** Retrieval performance using selected terms from *Note* query.

#### 4.4 Feature Importance

In order to better understand the usefulness of the features, we also tested the importance of the each type of features by removing it from the feature set and see how the retrieval performance would change. Specifically, we removed each type of features as described in Table 2, and trained the model with the remaining features. We did the experiments using the description query over the three collections. Table 6 summarize the feature importance analysis results. The negative value in the table means the retrieval performance drops after this type of features is removed from the feature space. Not surprisingly, the domain specific features played an significant role on identifying the Key Terms, while the term statistic features and lexicon features are also promising in important term classification. We further analyzed the performance by removing the features one at a time to see the performance changes. The results indicates that the  $abbr(t_i)$ ,  $IDF(t_i)$  and  $wig(t_i)$  are the most useful features except the domain features. The POS features did not work well. By further looking into the data, it shows that the noun phrases and verb phrases occur both as the Key Terms and non-Key Terms, so it is hard to learn the pattern from the POS features.

#### 4.5 Example of Identified Key Terms

It is useful to further analyze the identified Key Terms by revealing the characteristics of those useful terms. This could allow users to learn how to formulate an effective query. Therefore, we report the actual terms being kept by different methods from both description query and note query as in Table 7 and Table 8.

	Domain	Lexicon	POS	Statistical	Locality
<b>CDS14</b>	-0.067	-0.025	-0.005	-0.058	-0.004
<b>CDS15</b>	-0.074	-0.037	0.013	-0.047	0.003
<b>CDS16</b>	-0.066	-0.028	-0.004	-0.045	-0.007

**Table 6.** Retrieval performance changes when one type of features is removed.

Methods	Identified Key terms
<b>A shorter version of the EMR</b>	78 M transferred to nursing home for rehab after CABG. Reportedly readmitted with a small NQWMI. Yesterday, he was noted to have a melanotic stool and then today he had approximately 9 loose BM w/ some melena and some frank blood just prior to transfer, unclear quantity.
<b>Simplified query</b>	A 78 year old male presents with frequent stools and melena.
<b>Noun Phrase</b>	nursing home a small NQWMI a melanotic stool 9 loose BM some melena and some frank blood
<b>Key Concept</b>	nursing home CABG a small NQWMI noted stool prior to transfer
<b>Fast QQP</b>	nursing CABG readmitted with a small NQWMI melanotic stool approximately loose melena
<b>learn2extract</b>	rehab CABG NQWMI melanotic stool BM melena frank blood

**Table 7.** Identified Key Terms from the *description* query

By observing the simplified query, we see that the chief complaint of the patient is frequent stools and melena. These two concepts should be covered in the extracted key terms in order to achieve a reasonable performance. We first examined the terms selected by each method from the description query (i.e. Table 7). For the Noun Phrase method, although both two concepts are covered in the shorter version, too many irrelevant terms have been kept since they are nouns. Thus, the identified query is drifted because of these noisy terms. The Key Concept and Fast QQP methods solved this problem to a certain extent by involving other features when selecting the query terms, but they still suffer from the noisy terms, such as “nursing home” and “approximately”. In addition, they missed some important terms in the shorter version (such as “melena” for Key Concept). This would also hurt the performance too. Our method, on the other hand, successfully identified these two aspects, and our method could also bring additional useful term (i.e., “frank blood”) to the query.

We then examined the key term selection from the note query as shown in Table 8. Since the note query contains more information than the description query, every methods generated a much longer key terms list comparing with the key terms selected based on description query. This also explains the performance decrease of using note query as shown in Table 1. After a close look at the identified key terms in Table 8, we find that although all methods are suffered

Query Type	Query Content
<b>The EMR of the patient</b>	78 M w/ pmh of CABG in early [**Month (only) 3**] at [**Hospital6 4406**] (transferred to nursing home for rehab on [**12-8**] after several falls out of bed.) He was then readmitted to [**Hospital6 1749**] on [**3120-12-11**] after developing acute pulmonary edema/CHF/unresponsiveness?. There was a question whether he had a small MI; he reportedly had a small NQWMI. He improved with diuresis and was not intubated. . Yesterday, he was noted to have a melanotic stool earlier this evening and then approximately 9 loose BM w/ some melena and some frank blood just prior to transfer, unclear quantity.
<b>Simplified query</b>	A 78 year old male presents with frequent stools and melena.
<b>Noun Phrase</b>	CABG home acute pulmonary edema unresponsiveness a small MI NQWMI diuresis loose BM melanotic stool frank blood unclear quantity
<b>Key Concept</b>	CABG nursing home acute pulmonary edema CHF unresponsiveness small NQWMI melanotic stool loose BM
<b>Fast QQP</b>	pmh CABG nursing home falls bed pulmonary edema CHF unresponsiveness diuresis was not intubated melanotic loose frank blood
<b>learn2extract</b>	pmh CABG nursing home rehab pulmonary edema CHF NQWMI melanotic stool loose BM melena frank blood

**Table 8.** Identified Key Terms from the *note* query

from the query drifting problem because of the additional terms, our method contains the least, yet most useful, terms comparing with the other methods.

## 5 Conclusion

In this work, we proposed a new set of features to identify the Key Terms from verbose query for retrieval in medical domain. Experiment results over three data collections show that using the selected Key Terms could significantly improve the retrieval performance than directly using the original verbose query, and it also outperform two strong baselines.

There are many directions that we plan to work on as future work. First, we would like to explore more features, especially more domain features, to enrich the feature space. Second, instead of the classifier, we would like to design a weighting schema for each term based on their importance. Finally, it would be interesting to see how the proposed feature would work with the other machine learning algorithm, such as CNN, to solve this problem.

**Acknowledgments.** This research was supported by the U.S. National Science Foundation under IIS-1423002.

## References

1. Gupta, M., Bendersky, M.: Information retrieval with verbose queries. In: Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval. SIGIR '15, New York, NY, USA, ACM (2015) 1121–1124
2. Bendersky, M., Croft, W.B.: Discovering key concepts in verbose queries. In: Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. SIGIR '08 (2008) 491–498
3. Jones, R., Fain, D.C.: Query word deletion prediction. In: Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. SIGIR '03, New York, NY, USA, ACM (2003) 435–436
4. Soldaini, L., Cohan, A., Yates, A., Goharian, N., Frieder, O.: Retrieving Medical Literature for Clinical Decision Support. Springer International Publishing, Cham (2015) 538–549
5. Kumaran, G., Carvalho, V.R.: Reducing long queries using query quality predictors. In: Proceedings of the 32Nd International ACM SIGIR Conference on Research and Development in Information Retrieval. SIGIR '09 (2009) 564–571
6. Turney, P.D.: Learning algorithms for keyphrase extraction. *Information Retrieval* **2**(4) (2000) 303–336
7. Díaz-Galiano, M.C., Martín-Valdivia, M., Ureña López, L.A.: Query expansion with a medical ontology to improve a multimodal information retrieval system. *Comput. Biol. Med.* **39**(4) (April 2009) 396–403
8. Martinez, D., Otegi, A., Soroa, A., Agirre, E.: Improving search over electronic health records using umls-based query expansion through random walks. *Journal of Biomedical Informatics* **51** (2014) 100 – 106
9. Yang, C., He, B., Xu, J.: Integrating feedback-based semantic evidence to enhance retrieval effectiveness for clinical decision support. In: Asia-Pacific Web (APWeb) and Web-Age Information Management (WAIM) Joint Conference on Web and Big Data, Springer (2017) 153–168
10. Zhu, D., Carterette, B.: Combining multi-level evidence for medical record retrieval. In: Proceedings of the 2012 international workshop on Smart health and wellbeing (SHB'12). (2012) 49–56
11. Limsopatham, N., Macdonald, C., Ounis, I.: Aggregating evidence from hospital departments to improve medical records search. In: Proceedings of the 35th European Conference on Advances in Information Retrieval. ECIR'13 (2013) 279–291
12. Wang, Y., Fang, H.: Exploring the query expansion methods for concept based representation. In: TREC'14. (2014)
13. Limsopatham, N., Macdonald, C., Ounis, I.: Learning to combine representations for medical records search. In: Proceedings of SIGIR'13. (2013)
14. Qi, Y., Laquerre, P.F.: Retrieving Medical Records: NEC Labs America at TREC 2012 Medical Record Track. In: TREC 2012. (2012)
15. Koopman, B., Zuccon, G., Nguyen, A., Vickers, D., Butt, L., Bruza, P.D.: Exploiting SNOMED CT Concepts & Relationships for Clinical Information Retrieval: Australian e-Health Research Centre and Queensland University of Technology at the TREC 2012 Medical Track. In: TREC'12. (2012)
16. Wang, Y., Liu, X., Fang, H.: A study of concept-based weighting regularization for medical records search. In: ACL2014. (2014)
17. Wang, Y., Fang, H.: Extracting Useful Information from Clinical Notes. In: TREC 2016. (2016)