Identifying Relevant Questions Regarding Laboratory Tests from Community Question Answering Websites

xxx xx[1], xxxx xxxx[2], xxxx xxxx[1], and xxxx xxxx[3]

[1] XXXX, xxx xxx, xx, xxx
xxx@xxx.xxx
[2] XXXX, xxx xxx, xx, xxx
xxx@xxx.xxx
[3] XXXX, xxx xxx, xx, xxx
xxx@xxx.xxx

Abstract. The adoption of patient portals has made clinical data, such as laboratory results, increasingly accessible to patients. However, many portals provide limited support for patients to make sense of their clinical data, causing patients to seek information and actionable advice from supplemental sources, including health forums and Community Question Answering (CQA) sites. In spite of the numerous research efforts on retrieving similar medical questions, few studies have examined this important topic in the context of lab tests. Since the ranges and types of lab tests can lead to different healthcare concerns, identifying and retrieving relevant questions containing lab tests according to inquiries from information seekers is critical. To this end, we build and evaluate an end-to-end system that can automatically identify similar medical questions about lab tests. Specifically, we investigate the traditional vector representation (BoW) of questions, neural sentence embedding of the questions, word embedding based question representations and their integration with the extracted lab test and other engineered features. The system is tested using questions collected from Yahoo! Answers' health section. The results show that the integrated multi-level vector representation with the extracted lab test features performs the best. It not only identifies relevant questions that contain lab tests effectively, but also can be applied to the medical questions that have no lab test mentioned.

Keywords: Medical question retrieval · Laboratory test · Community Question Answering

1 Introduction

Patients are increasingly using patient portals—an online service that gives patients secure and convenient access to personal health information—to view their clinical data, such as laboratory test results [21]. However, patients often have difficulties comprehending and acting upon the clinical data presented in portals [1,10]. In response, increasingly, patients turn to online resources (e.g., search

engines, health forums, and social media) to make sense of their data and obtain actionable advice [28, 30]. In fact, over 70% of adult Internet users in the U.S. reported searching online for health information [30].

Among various online resources from which patients may choose, community question answering (CQA) websites have been widely used due to their high levels of interactivity and quality — these sites allow users to post full questions in natural language and include key contextual information to seek health information and advice. As one of the most fundamental features, retrieving similar questions on CQA sites has been the focus of many studies [2, 5, 13, 19, 23, 25]. For example, in a recent study [16], Li et al. adopted Long Short-Term Memory (LSTM) to learn the vector representations of large-scale questions and answers collected from real-world websites. Similar medical questions were retrieved by calculating the vector similarity score of an input question and the candidate questions in the corpus. However, to date, prior work has not looked at how to support retrieving relevant medical questions in the context of lab tests. Retrieving relevant questions containing lab results is critical because the ranges and types of lab tests can lead to different healthcare concerns. Without considering these features, the system may return irrelevant posts with limited useful information, leaving the users entangled with what to do next.

To address this research gap, we build and evaluate a system consisting of multi-level text representations, extracted lab test information, and engineered features to identify the relevant medical questions containing lab tests. The multi-level text representations include traditional Bag-of-Words (BoW) text representation – TF-IDF, the deep contextualized word embeddings – ELMo, and the sentence-level embedding – Universal Sentence Encoder (USE). Two lab test features – the type of lab test and the clinically meaningful range of lab results – are extracted from the questions. Other engineered features including the length and type of the question, and the number of stop words are also investigated to determine whether they can improve system performance. The system is evaluated using questions posted on Yahoo! Answers diabetes subsection of the health section between 2009 and 2014. Specifically, we evaluate different approaches using questions that contain lab tests related to diabetes: creatinine, HbA1c, and glucose test. It is worth noting that the system can be generalized to questions that contain other lab tests and blood pressure measurements. In order to demonstrate that our system does not sacrifice the performance on the medical questions without lab tests mentioned, a group of questions without any lab tests are used to evaluate the system as well. The normalized discounted cumulative gain (nDCG) is used to compare the ranked results of each method. Our results show that the transformer-based universal sentence encoder works well in capturing the semantic meaning of questions, and adding the extracted lab test features improves the overall performance. We conclude this paper by discussing study limitations and providing an outlook for future work.

In this research, we make the following contributions: (i) This is the first research to investigate a medical question retrieval system considering the medical lab test and their ranges from the clinical point of view; (ii) We investigate

different question representations in the combination of different engineered features; (iii) We generate a human-annotated data set which can be used for future research on medical Q&A and question retrieval.

2 Related Work

A traditional way to identify similar questions is to retrieve questions based on the Bag-of-Words (BoW) approaches, which mainly rely on the shared keywords between new question and candidate questions. One limitation of the BoW approaches is that it considers the information of individual terms, while ignoring the domain-specific information. For medical-related questions, important information such as patients' health status, demographics, symptoms, and other medical-related information is not emphasized in BoW approaches. On the other hand, the BoW approaches do not consider different semantic representations of the same concept. In the medical question answering (Q&A) data set that we use, we observe that people use very different ways to express the same medical concept. For example, information seekers use different terms (e.g. 'blood sugar', 'glucose') to describe the glucose lab test in their questions. This exerts difficulties in calculating question similarities based on the BoW approaches only.

In light of the limitations of the BoW approaches, starting from 2013, word embedding approaches and neural information retrieval methods have been introduced and started gaining popularity, with seminal research studies attempted to apply them in the question-retrieval domain. For example, Zhou et al. [29] developed a word-embedding approach that treats a question as a Bag-of-Embedded-Words (BoEW) by employing the skip-gram model [17]. A generative model in the FK framework [6] was used to generate fisher vectors (FVs) by aggregating the BoEWs for constructing question representations. Question retrieval was achieved by calculating the similarity between the new representation of a query question and the candidate questions. Lei et al. [15] adopted a gated (non-consecutive) convolutional semi-supervised question retrieval algorithm, and used an encoder-decoder network to generate representations of the title of a question, body and merged title-body of the question. This encoder-decoder was compared with other benchmark encoders (LSTMs, GRUs, and CNNs). The Stack Exchange AskUbuntu dataset was used to evaluate these approaches. The results showed that Lei's model yielded substantial gains over a standard IR baseline and the ones being compared with. However, the approaches in both [29] and [15] are not applicable to the task of medical questions, since the encoder-decoder does not consider the details of the content, such as lab tests.

Some other related research has been done in the biomedical Q&A domain. Brokos et al. [3] represented questions as vectors using embeddings and then treated the query question as a centroid of the embeddings. The relevant questions were retrieved based on the distance between the candidate questions to the centroid. This method was tested on biomedical questions from the BioASQ competition [26]. The retrieved candidates were ranked using the relaxation of Word Movers Distance (WMD) proposed by Kusner et al. [14]. Although this method

outperformed PubMed — the most widely-used biomedical search engines, it did not take into account the context of medical lab test in the biomedical questions.

In summary, none of the previous question retrieval or CQA research investigates how to effectively identify relevant questions with regards to the medical lab tests. To the best of our knowledge, this research is the first to evaluate different approaches in comparison with the baseline BoW approach.

3 Overview of the System

Figure 1 shows a high-level overview of the proposed system that is designed to find similar medical questions about lab tests in CQA websites. Given a query question, such as "Not diabetic...3 months ago had a 6.1 Hemoglobin A1C...NOW ITS 5.8..Is that good?", we first apply a series of data pre-processing steps, such as converting all words into lower cases and removing the redundant punctuation. Then, we extract medical lab tests information from the question. Meanwhile, we construct other engineered features based on the content of the question. Next, for the query question and each question in the candidate set, we calculate the similarity score based on question vector representations produced by different approaches and the engineered features. Finally, all the candidate questions are ranked according to their corresponding similarity scores, and the top ones whose similarity scores are larger than a threshold are considered as the most similar medical questions. The details of each component are described in the following sections.

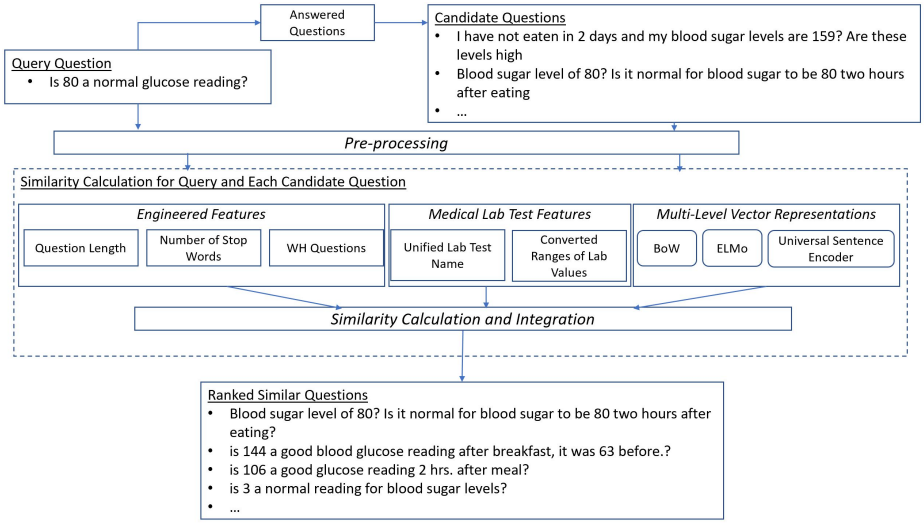

Fig. 1. System Overview.

4 Methods

4.1 Medical Lab Test Extraction and Feature Constructing

An existing numeric expression extraction algorithm, Valx [11], is used to extract lab test information from an input text. Valx processes text and extracts lab test results in the following steps:

- Preprocessing: it normalizes special symbols, deletes blank spaces, and corrects typos according to commonly used digit-grouping delimiters in numeric comparison statements.
- Structuring numeric expressions: it extracts numeric values, units (e.g., mmol/l), and comparison operators (e.g., equal to).
- Identifying lab test names: it identifies variables using hybrid knowledge including contextual knowledge, domain knowledge, the Unified Medical Language System (UMLS) Metathesaurus, to detect known and unknown variables, such as age and lab tests.
- Removing illegitimate units: it conducts context-based association filtering to remove the illegitimate unit list of an identified lab test.
- Normalizing units: it normalizes the measurement units to standardized ones, such as 'mg/dl', 'g/l', 'mmol/l', and so on.
- Removing erroneous extraction: it conducts a heuristic rule-based evaluation to examine and remove incorrect extraction.

Given a text, the extracted lab test(s) consists of four components: name of the lab test, a comparison operator (e.g., "equal to", "lower than", "greater than"), value of the lab test, and a measurement unit (e.g., "mmol/l"). If the lab test has no standard measurement unit, the measurement unit is left empty. Table 1 presents examples of extracted lab test information using Valx. Each extracted lab has a standardized name, a value, and a measurement unit which is highlighted in the table. Based on the extracted name and value of the lab test,

Table 1. Examples of Extracted Medical Lab Tests

Text	Extracted Lab Tests
" Fasting blood sugar is 130. am i diabetic?"	Glucose equal to 7.22 mmol/l
"is an A1c level higher than 8.0 bad?"	HBA1C greater than 8.0 %
"my creatinine equal lower than 0.5. Is it ok?"	creatinine lower than 0.5 mg/dL

we construct two engineered features. One is the name of the lab test, the other is the range of the lab test based on the value. We convert the value of the lab test into a range based on the official websites of Mayo Clinic and the National Institutes of Health [7–9, 18], which provide authoritative medical knowledge. We consider three range levels for each lab test in this research. For example,

for glucose, given a lab test value, we convert it into the normal range, pre-diabetic range, or diabetic range based on the published information on Mayo Clinic website. The HBA1C values and creatinine values are converted into low, normal and high ranges. Table 2 shows a few examples of converting extracted lab test values to ranges.

Table 2. Converting Lab Test Values to Ranges

Extracted Lab Tests	Range
Glucose equal to 7.22 mmol/l	diabetic
HBA1C greater than 8.0 %	high
creatinine lower than 0.5 mg/dL	low

4.2 Multi-level Vector Representations of the Questions

To measure the similarity between the query question and candidate questions, we investigate different representations of those questions. More specifically, TF-IDF is used to construct vectors as the baseline Bag-of-Words approach. Two different embeddings – ELMo and Universal Sentence Encoder (USE) – are also investigated for question representation. We assume that these embedding representations can better capture the semantic relationships between the words in the questions. The ELMo embedding is trained using the bi-directional Long-Short Term Memory model, whereas USE is trained using a transformer architecture. Before constructing different vector representations, stemming is performed to the words in the questions.

Bag-of-Words The traditional vector space model along with TF-IDF weighting scheme is employed as our basic vector representation. The equation of the TF-IDF used in this research is given in Equation 1.

$$tfidf = tf(t, d) \times idf(t) \tag{1}$$

where $tf(t, d)$ is the term frequency of t in document d, and $idf(t)$ is the inverse document frequency of term t. It is obvious that the BoW representation does not consider the semantic relations of the words within the text.

ELMo for Sentence Representation ELMo stands for Embeddings from Language Models [20]. It is a set of deep contextualized vector representations derived from bi-directional language models (biLMs), which take an entire sentence as input and is pre-trained on a large corpus of texts. ELMo is capable of producing word embeddings by considering the context words surrounding each other. Given a sequence of N tokens, $(t_1, t_2, \ldots, t_N)$, a forward language model (LM) computes the probability of the sequence by calculating the probability of token t_k based on all previous tokens $(t_1, ..., t_{\text{k-1}})$ (shown as Equation 2). A

backward LM works the same way as the forward LM except it scans through the sequence in reverse order to predict the previous token based on the future context-dependent tokens, as shown in Equation 3.

$$p(t_1, t_2, , t_N) = \prod_{k=1}^{N} p(t_k | t_1, t_2, , t_{\text{k-1}}) \tag{2}$$

$$p(t_1, t_2, , t_N) = \prod_{k=1}^{N} p(t_k | t_{\text{k+1}}, t_{\text{k+2}}, , t_{\text{n}}) \tag{3}$$

The training objective is to jointly maximize the log probability of the forward and backward directions. The output of the ELMo is token embeddings which are the combinations of the intermediate layer representations in the biLM. An averaging of the token embeddings is employed to produce the sentence embedding. In this research, we use the pre-trained ELMo which was trained on the 1 Billion Word Benchmark, approximately 800M tokens of news crawl data from WMT 2011 [4]. The output sentence vector is 3,072 dimensions.

Universal Sentence Encoder The universal sentence encoder (USE) is a sentence-based embedding model for encoding sentences into vectors. There are two types of USE model. One is based on the transformer architecture [27] and affords greater model complexity and resource consumption to achieve high accuracy. The other makes use of a deep averaging network [12] and reduces accuracy for more efficient inference.

The transformer-based sentence encoding model employs the encoding sub-graph of the transformer architecture [27]. The sub-graph is adopted to identify all other words and their ordering to produce a contextual representation of a word in a sentence. Then the model computes the element-wise sum of the representations at each word position to convert word representations to a fixed-length sentence encoding vector. In this research, we use the pre-trained USE model. The input to the USE model is a lower-cased PTB tokenized string. The output is a sentence embedding of a 512-dimensional vector.

4.3 Other Engineered Features

In addition to the type of lab test and the range of its corresponding value, we also investigate other engineered features to demonstrate whether they can be used to improve the performance of the system. These features include the sentence length by the number of words in total, the number of stop words in the questions, and whether it is a WH question. The definition of these engineered features is given below:

Sentence Length: In this research, we consider questions that have a length from 5 to 20 words, both inclusive. The reason is that we find some medical questions that have a very long description which involves content that is not directly related to the main content of the question. Hence, we limit the maximum length of the sentence to be 20 words.

Stop Word Count: Stop words are commonly-used, such as short functional words (e.g., “the”, “a”, “an”, “in”). Based on the literature [24], the number of stop words in the sentence helps to identify similar questions to some extent. Thus, we also investigate this engineered feature.

WH Question: In this research, we investigate whether the questions are similar to each other when they are both WH questions. The WH questions include questions begin with either ”wh” or ”how”. Specifically, based on our experimental data set, the four most frequently-occurred question types are selected: what, how, when, and why.

4.4 Similarity Calculation

Each question is represented using vector representations along with engineered features. The similarity between the questions is measured based on each engineered feature and the vector representation.

Engineered Features For engineered features that include medical lab test and lab test range, we first normalize each into the same range using the min-max normalization shown in Equation 4, where x is the value of the feature, max is the maximum feature value of query and candidate questions, min the minimum feature value of query and candidate questions.

$$y = \frac{(x - min)}{(max - min)} \tag{4}$$

For the lab test type and range, we first code them into numeric values. After the feature values are normalized, the similarity between the feature values sim_{fe} is calculated using Equation 5, where y_a is the min-max normalized feature value of a query question, and y_b is the normalized feature value of a candidate question. Table 3 shows examples of how the features and similarities are calculated.

$$sim_{ef} = 1 - |y_a - y_b| \tag{5}$$

Vector Representations We use cosine similarity to measure the similarity between the vector representations of the questions. Since BoW- and ELMo-based sentence representations as well as Universal Sentence Encoder generate different vector representations for the same question, we measure the cosine distance for each vector representation using Equation 6, where a and b are vector representations of the same type (e.g. BoW or ELMo) of two questions, i is the dimension of the vector.

$$sim_{vf}(a, b) = \frac{\sum_{i=1}^{n} a_i b_i}{\sqrt{\sum_{i=1}^{n} a_i^2} \sqrt{\sum_{i=1}^{n} b_i^2}} \tag{6}$$

Table 3. Question Similarity Calculation

Features	Query Question	Candidate Question	Similarity
	My a1c is 5.4 do i have pre-diabetes?	Is a 7.0 a1c average over a 10 year period very good control?	
Sentence Length	8	13	0.53
Stop Word Count	3	3	0.69
WH Question	0 (Not WH Question)	0 (Not WH Question)	1
Medical Lab Test	1 (HBA1C)	1 (HBA1C)	1
Medical Test Range	1 (Normal)	2 (Pre-diabetic)	0.67
BoW	-	-	0.12
ELMo	-	-	0.55
USE	-	-	0.66

4.5 Question Ranking

In this research, we evaluate integration of different vector representations and engineered features including lab test features through weighted linear combination (Equation 7), where sim_{vf} includes similarity measures of different vector representations, sim_{ef} includes similarity measures of different engineered features, $sim_{lab\ test}$ includes similarity measures based on lab tests and their ranges. The weights can be given differently to each similarity measure or the group of similarity measures. In this research, the weights are experimented through extensive evaluations. The candidate questions are ranked based on their final values from largest to smallest. The ranked results are then evaluated by human annotators recruited for this research.

$$sim^{total} = \sum w_{vf} sim_{vf} + \sum w_{ef} sim_{ef} + \sum w_{lab\ test} sim_{lab\ test} \quad (7)$$

4.6 Experimental Setup and Evaluation

Data set In this work, we use questions posted in the diabetes section of Yahoo! Answers from 2009 and 2014 [22]. The data set consists of 58,188 Q&A threads, each of which contains question subject, content, and answers to the question. By applying Valx algorithm described in Section 4.1, we find that 31,165 out of 58,188 questions (53.6%) (shown in Figure 4.6) mentions at least one lab test and its value, demonstrating the importance of supporting users to find and retrieve relevant questions containing similar lab results. We analyze the length of the question subject in combination with its content, and find that a large number of questions contain less than 25 words. As such, in this work, we focus on questions with a length between 5 and 20 words. In order to evaluate the design system on questions containing medical lab tests, we focus on three lab tests that are related to diabetes: HbA1c, Glucose, and Creatinine. In total, we selected 3,000 questions for this study, including 305 questions that contain at least one of these three lab tests and 2,695 other questions without lab tests.

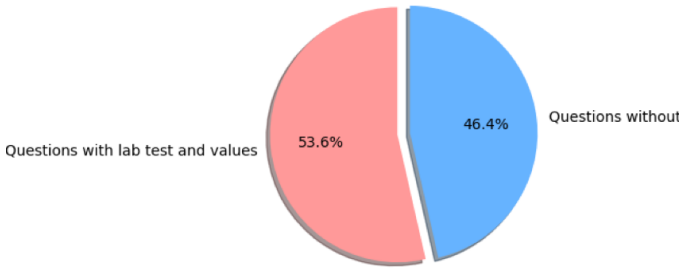

Fig. 2. Distribution of Medical Questions

Ranking Evaluation by Human Annotators Since there is no published labeled data set for this question retrieval task, we recruit three human annotators to rate the candidate questions for each query question. Each annotator is asked to annotate the candidate questions for 40 query questions. Among these 40 questions, 10 questions contain creatinine, glucose, and HbA1c results, respectively, and the last 10 questions contain no lab tests at all. The reason to include questions with no lab tests is to evaluate whether our system can be generalized to other medical question retrieval tasks. A guideline of relevance judgment is given to each annotator. Each annotator is asked to gives each candidate question a score in the range of 0 to 5, 5 is extremely relevant and 0 is no relevance at all. The candidate questions are independently scored by annotators who have only general knowledge about diabetes-related lab tests. The scores for each candidate question are then averaged to be used as the ground truth relevancy scores.

Performance Evaluation To evaluate and compare the performance of different methods, we employ the normalize Discounted Cumulative gain ($nDCG$) which is a typical evaluation method in the information retrieval domain. The $nDCG$ at position p is calculated as $nDCG = \frac{DCG_p}{IDCG_p}$ which is based on the Discounted Cumulative Gain (DCG) calculated using Equation 8 and ideal Discounted Cumulative Gain ($IDCG$) at position p. The $IDCG$ calculation is similar to DCG, the only difference is that $IDCG$ is calculated based on the ideal ranking by using the ground truth relevancy scores. p is the position of the ranked results, rel_i is the ground truth relevancy score given by the human annotators. We also calculate the effect size statistically compare the performance of the different methods against the baseline BoW approach. The calculation of effect size is given as Equation 9, where μ_a and μ_{BoW} are the mean $nDCG$ values of 10 questions at a position of the method a and BoW, σ is the standard deviation of baseline BoW.

$$DCG_p = \sum_{i=1}^{p} \frac{rel_i}{\log_2(i+1)} \tag{8}$$

$$\Delta = \frac{\mu_a - \mu_{BoW}}{\sigma} \tag{9}$$

5 Experimental Results

In this research, we evaluated different integration of vector representations in combination with lab test features and engineered features. We compared those against the baseline BoW representation. Tables 4 and 5 show the results of query questions containing one of the three lab tests and no lab test. For each scenario, we took the mean $nDCG$ at position 5 (@5) and 10 (@10) of 10 query questions. ISM stands for the integration of three vector space representations with a weight distribution of 0.15, 0.2 and 0.3 for BoW, ELMo and USE respectively. When engineered features (EF) were considered, the weights for sentence length, WH question and stop word count were 0.08, 0.05 and 0.05 respectively. The weights for lab and lab range were set to 0.1 and 0.07 respectively.

Table 4. Comparison Results for Questions Contain Medical Lab Test

Method	Creatinie Test				Glucose Test				HbA1c Test			
	@5	Δ	@10	Δ	@5	Δ	@10	Δ	@5	Δ	@10	Δ
BoW	0.626	0	0.821	0	0.782	0	0.910	0	0.627	0	0.831	0
USE only	0.783	1.319	0.899	1.321	**0.895**	0.945	0.948	0.648	0.764	0.829	0.897	0.718
ELMo only	0.735	1.854	0.880	1.712	0.774	-0.064	0.900	-0.154	0.772	0.876	0.904	0.804
BoW+LAB	0.865	0.910	0.935	0.994	0.867	0.708	**0.951**	**0.695**	**0.837**	**1.272**	**0.938**	**1.171**
USE+LAB	**0.892**	**2.223**	**0.942**	**2.043**	0.870	0.736	0.945	0.588	**0.886**	**1.567**	**0.947**	**1.266**
ELMo+LAB	0.847	1.854	0.922	1.712	0.816	0.285	0.924	0.244	0.824	1.195	0.932	1.106
ISM	0.742	0.975	0.883	1.050	**0.887**	**0.879**	**0.952**	**0.708**	0.795	1.020	0.914	0.913
ISM+EF	0.741	0.967	0.898	1.303	0.804	0.182	0.918	0.151	0.709	0.498	0.872	0.450
ISM+LAB	**0.874**	**2.083**	**0.941**	**2.031**	**0.887**	**0.877**	**0.950**	**0.682**	**0.887**	**1.575**	**0.957**	**1.382**
ISM+LAB+EF	**0.876**	**2.094**	**0.946**	**2.123**	0.842	0.499	0.921	0.192	0.771	0.873	0.901	0.768

Table 5. Comparison Results for Questions Containing No Medical Lab Tests

Method	@5	Δ	@10	Δ
BoW	**0.892**	0	**0.953**	0
USE only	0.827	-0.782	0.916	-0.756
ELMo only	0.810	-0.989	0.935	-0.354
ISM	**0.890**	**-0.015**	**0.953**	**0.002**
ISM+EF	0.879	-0.154	0.946	-0.150

Because the differences between $nDCG$ values and Δ values are not very significant for some of the methods, we highlight the top 3 $nDCG$ with the Δ for each column in Table 4 and the top 2 $nDCG$ with the Δ for each column in Table 5. Based on the results of questions containing medical tests, we find that most of the methods outperform the baseline BoW. More specifically, ISM+LAB model performs consistently well for all the cases. Even though this model does not have the best performance for some creatinine and glucose tests, the nDCG@10 is very

close to the best. USE+LAB performs well except for glucose test. For both creatinine and HbA1c tests, the method considering medical lab test features perform better than the ones without. For example, BoW+LAB works better than BoW only. However, for glucose test, we find that USE only works the best for the top 5 ranked results, and ISM works the best for top 10 ranked results. Both USE only and ISM don't contain any medical features. After thorough investigation, we find that since the lab extraction algorithm Valx was originally designed to extract medical tests from the text in the eligibility criteria section of clinical trial summaries, it does not work optimally in detecting some terms used in lay person's language. For example, 'fasting sugar' can not be recognized by Valx as a glucose test. We also find that adding general engineered features does not improve the performance for most of the cases, except for creatinine tests.

Table 5 shows results for questions without any lab tests. We find that both baseline BoW and ISM perform well but BoW performs slightly better for top 5 ranked results, and ISM performs slightly better for top 10 ranked results. One possible explanation is that users often use common words in query and candidate questions to determine whether the questions are similar. For the questions without lab tests, adding engineered features does not improve the performance.

Overall, we find ISM model performs well on capturing the semantic similarity between questions. By adding lab test features to the ISM model, it can work well for medical questions containing lab tests.

6 Conclusions, Limitations and Future Work

In this study, we investigate the integration of multi-level text representations (BoW, ELMo, USE) with medical lab test features (the type of lab test and the clinically meaningful range of the lab values) extracted using Valx and other engineered features built from the questions. We find that by adding the lab test features to the integrated multi-level text representation (ISM+LAB), we can gain a consistent result for all three medical lab tests in this research. The ISM+LAB model can be extended to other lab tests including blood pressure. In this study, the performance of the compared model relies on the accuracy of Valx, a tool used to extract lab test information from posted questions. The errors caused by Valx might impact our results. To limit the impact, post-processing after lab extraction with Valx needs to be conducted. On the other hand, all of the annotators we recruited for this study only have basic knowledge of diabetes. They are not aware of the normal ranges for the examined lab tests. If two lab values are very close but fall into different categories (e.g., normal vs. abnormal), they tend to think the results are similar. Therefore, in the future, we need to provide sufficient training to the annotators or even add a feature to measure the value differences. We will also apply learning algorithms, such as polynomial regression or deep-learning regression, to optimize the weights for feature integration, and use other datasets collected from health forums, such as MedHelp.

References

1. Jordan M Alpert, Alex H Krist, Rebecca A Aycock, and Gary L Kreps. Applying multiple methods to comprehensively evaluate a patient portals effectiveness to convey information to patients. *Journal of medical Internet research*, 18(5):e112, 2016.
2. Kyoungman Bae and Youngjoong Ko. Efficient question classification and retrieval using category information and word embedding on cqa services. *Journal of Intelligent Information Systems*, pages 1–23, 2019.
3. Georgios-Ioannis Brokos, Prodromos Malakasiotis, and Ion Androutsopoulos. Using centroids of word embeddings and word mover's distance for biomedical document retrieval in question answering. *arXiv preprint arXiv:1608.03905*, 2016.
4. Ciprian Chelba, Tomas Mikolov, Mike Schuster, Qi Ge, Thorsten Brants, Phillipp Koehn, and Tony Robinson. One billion word benchmark for measuring progress in statistical language modeling. *arXiv preprint arXiv:1312.3005*, 2013.
5. Zheqian Chen, Chi Zhang, Zhou Zhao, Chengwei Yao, and Deng Cai. Question retrieval for community-based question answering via heterogeneous social influential network. *Neurocomputing*, 285:117–124, 2018.
6. Stéphane Clinchant and Florent Perronnin. Aggregating continuous word embeddings for information retrieval. In *Proceedings of the workshop on continuous vector space models and their compositionality*, pages 100–109, 2013.
7. Mayo Clinic. A1c test, n.d. `https://www.mayoclinic.org/tests-procedures/a1c-test/about/pac-20384643`.
8. Mayo Clinic. Creatinine test, n.d. `https://www.mayoclinic.org/tests-procedures/creatinine-test/about/pac-20384646`.
9. Mayo Clinic. Diabetes, n.d. `https://www.mayoclinic.org/diseases-conditions/diabetes/diagnosis-treatment/drc-20371451`.
10. Perry M Gee, Debora A Paterniti, Deborah Ward, and Lisa M Soederberg Miller. e-patients perceptions of using personal health records for self-management support of chronic illness. *CIN: Computers, Informatics, Nursing*, 33(6):229–237, 2015.
11. Tianyong Hao, Hongfang Liu, and Chunhua Weng. Valx: a system for extracting and structuring numeric lab test comparison statements from text. *Methods of information in medicine*, 55(03):266–275, 2016.
12. Mohit Iyyer, Varun Manjunatha, Jordan Boyd-Graber, and Hal Daumé III. Deep unordered composition rivals syntactic methods for text classification. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, volume 1, pages 1681–1691, 2015.
13. Dipankar Kundu and Deba Prasad Mandal. Formulation of a hybrid expertise retrieval system in community question answering services. *Applied Intelligence*, 49(2):463–477, 2019.
14. Matt Kusner, Yu Sun, Nicholas Kolkin, and Kilian Weinberger. From word embeddings to document distances. In *International conference on machine learning*, pages 957–966, 2015.
15. Tao Lei, Hrishikesh Joshi, Regina Barzilay, Tommi Jaakkola, Katerina Tymoshenko, Alessandro Moschitti, and Lluis Marquez. Semi-supervised question retrieval with gated convolutions. *arXiv preprint arXiv:1512.05726*, 2015.
16. Yaliang Li, Liuyi Yao, Nan Du, Jing Gao, Qi Li, Chuishi Meng, Chenwei Zhang, and Wei Fan. Finding similar medical questions from question answering websites. *arXiv preprint arXiv:1810.05983*, 2018.

17. Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.
18. National Institute of Diabetes, Digestive, and Kidney Diseases. The a1c test diabetes.
19. Nouha Othman, Rim Faiz, and Kamel Smaïli. Enhancing question retrieval in community question answering using word embeddings. 2019.
20. Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*, 2018.
21. Francesca Pillemer, Rebecca Anhang Price, Suzanne Paone, G Daniel Martich, Steve Albert, Leila Haidari, Glenn Updike, Robert Rudin, Darren Liu, and Ateev Mehrotra. Direct release of test results to patients increases patient engagement and utilization of care. *PLoS One*, 11(6):e0154743, 2016.
22. Yahoo! QA. Yahoo! health question and answering, n.d. `https://answers.yahoo.com/dir/index?sid=396545018`.
23. Andreas Rücklé, Krishnkant Swarnkar, and Iryna Gurevych. Improved cross-lingual question retrieval for community question answering. In *The World Wide Web Conference*, pages 3179–3186. ACM, 2019.
24. Anna Shtok, Gideon Dror, Yoelle Maarek, and Idan Szpektor. Learning from the past: answering new questions with past answers. In *Proceedings of the 21st international conference on World Wide Web*, pages 759–768. ACM, 2012.
25. Jan Trienes and Krisztian Balog. Identifying unclear questions in community question answering websites. In *European Conference on Information Retrieval*, pages 276–289. Springer, 2019.
26. George Tsatsaronis, Georgios Balikas, Prodromos Malakasiotis, Ioannis Partalas, Matthias Zschunke, Michael R Alvers, Dirk Weissenborn, Anastasia Krithara, Sergios Petridis, Dimitris Polychronopoulos, et al. An overview of the bioasq large-scale biomedical semantic indexing and question answering competition. *BMC bioinformatics*, 16(1):138, 2015.
27. Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.
28. Yan Zhang. Contextualizing consumer health information searching: an analysis of questions in a social q&a community. In *Proceedings of the 1st ACM International Health Informatics Symposium*, pages 210–219. ACM, 2010.
29. Guangyou Zhou, Tingting He, Jun Zhao, and Po Hu. Learning continuous word embedding with metadata for question retrieval in community question answering. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 250–259, 2015.
30. Kathryn Zickuhr. *Generations and their gadgets*. Pew Internet & American Life Project, 2010.