RESEARCH ARTICLE OPEN ACCESS

Research Questions (RQs) Classification for Systematic Literature Reviews in Software Engineering

Zuhal Hamad Ahmed [1], Naomie Salim [2]
PhD Student [1]
Sudan University of Science and Technology,
Sudan
Assoc. Prof [2]
University Technology Malaysia
Malaysia

ABSTRACT

Background: A systematic literature review (SLR) is a methodology used to aggregate all relevant existing evidence to answer a research question of interest. Although crucial, the process of conducting SLR can be time consuming, and must often be conducted manually.

Objective: The aim of this paper is to support the process of answering SLR research questions by first detecting the question class or expected answer type using approach used in question answering field.

Method: We built a data set of research questions (RQs) collected from SLR papers in software engineering field and labeled it with our proposed taxonomy. The proposed question taxonomy or answer type consists of 6 classes derived from the data set. From the data set questions we extracted three types of features, lexical features like n-gram, syntactic features like part of speech and semantic features like Hypernym of head word. We used Support Vector Machine (SVM) and Naïve Bayes classifiers to classify questions into its corresponding answer type using the mentioned features.

Results: The SVM showed accuracy 97% when using lexical features, and 95 % when using syntactic features, but when combining lexical, syntactic and semantic features the accuracy increased to 98% which is higher than accuracy showed by naïve bayes (79%), with the same features.

Conclusion: The results that obtained by SVM with a combination of the three types of features are very good and can be used in developing a system for answer extraction process when performing an SLR.

Keywords:- systematic literature review, question classification, answer extraction, systematic review automation.

I. INTRODUCTION

The idea of the systematic literature review (SLR) in software engineering was adopted from Evidence Based Medicine and introduced in an Evidence Based Software Engineering (EBSE) domain[1], since it have been introduced, Systematic Literature Review has gained focus of software engineering research community. It is a structured, well-organized, and step-by-step comprehensive method of conducting a review of the body of literature relevant to a particular research question. SLRs are often useful in identifying literature and research gaps relevant to a topic of interest [2]. The three main phases of an SLR include planning the review, conducting the review and reporting the review [2].

The results of the planning phase should be a clearly defined review protocol containing the purpose and the procedures of the review. The purpose of the review is identified by a set of answerable research questions (RQs). Conducting the review phase consists of five obligatory stages: ***Identification of research*** is a first step in conducting a review that will result in a list of entire population of publications relevant to the research questions and obtained by performing a search strategy. ***Selection of primary studies*** is performed on all identified studies by applying an inclusion and exclusion criteria in order to assess their actual relevance.

Study quality assessment is the most important part of the conducting the review phase. The idea of this process is to analyze and assess the quality of each primarily selected study in order to be finally included in data extraction and reporting process. ***Data extraction*** as a next stage aims to accurately and without bias record the appropriate information

from selected studies. ***Data synthesis*** is the final step in the review conduction phase. During this activity extracted data are collected and summarized.
The aim of the final phase of the systematic literature review process is to write the results of the review in a form suitable to dissemination channel and target audience or parties

Systematic reviews require considerably more effort than traditional reviews, and currently, most of its activities are done manually. Automating the SLR process will reduce most if not all of the human effort and time consumed to conduct it. The aim in this paper is to support answer extraction process which start after receiving the resulting relevant studies from the automatic selection and quality checking processes by firstly detecting the question class or the expected answer type.

This paper is organized as follows. Section 2 describes types of question. Section 3 Approaches for Question type Identification. Section 4: Experimental Design. Section 5: results and discussion. Section6: concludes the paper.

II. TYPES OF QUESTIONS

Most of the work from the literature of SLR automation processes is about searching for the primary studies process [3] [4] [5] and selecting primary studies [6] [7] [8] [9] [10]. Other SLR processes still need more work.
An automated question answering system has three stages: question processing, passage retrieval and answer processing. The task of question processing is to analyze the question and create a proper IR query as well as detecting the *entity type* of the answer, a category name which specifies the type of answer. The first task is called *query reformation* and the second is called *question classification.* our focus here will be on the question classification stage to analyze SLR research questions(RQs).

2.1 Research Questions (RQs) Classification

Question classification is a technique used to extract useful information from the question by assigning a question to an appropriate category from a set of predefined semantic categories (taxonomies) [11].
Question classification aims to predict the entity type or category of the answer expected.

Taxonomy

The set of question categories (classes) are referred to as question taxonomy. Different question taxonomies have been proposed in different works. There are two types of taxonomies: flat taxonomy and hierarchical taxonomy. Authors in [12] built the first hierarchical taxonomy for open domain question classification with 6 coarse grained classes and 50 fine grained classes and most of the recent work [13][14][15][16] used this taxonomy. For domain specific categories [17], taxonomy with 6 classes are proposed: define, describe, difference, enumerate advantages and reason in the technical domain and it is a flat taxonomy.

TABLE1: HIERARCHICAL TAXONOMY PROPOSED BY [12]

Coarse Class	Fine Classes
ABBREV.	abbreviation,expression
ENTITY	animal,body,color,creative,currency,disease, event,food,instrument,lang,letter,other, plant,product,religion,sport,substance, symbol,technique,term,vehicle,word
DESCRIPTION	definition,description,manner,reason
HUMAN	group,individual,title,description
LOCATION	city,country,mountain,other,state
NUMERIC	code,count,date,distance,money,order,other, period,percentage,speed,temp,volume-size,weight

TABLE2: FLAT TAXONOMIES FROM THE LITERATURE

Ref.	Taxonomy
[18]	Fact ,List ,reason ,solution ,definition ,navigation
[19]	Advantage/disadvantage ,cause and effect ,comparison ,definition ,example ,explanation ,identification ,list ,opinion ,rationale ,significance

Types of answers

Our work is differ from QA systems because we are dealing with short and long questions and our dataset is real data collected from SLR papers done manually by researchers.

After studying a number of SLR paper belonging to a various subjects of software engineering we tried to predict what is the expected answer type for a given question. Depending on that we classify the questions into the following classes:

1- **Quantification**: this class contains questions that ask about quantity information, percentage or rate of subject activity

How much empirical research has been done in CBSE since 1995?

2- **Person**: this class contains questions that ask about person (user/ researcher) or his role, participation or job in specified subject or topic.

1. Which software practitioners does the available SA research target?
2. What types of users are targeted in each approach?

3- **Confirmation:** this class contains confirmation questions which require answers in the form of yes or no with explanation of the answer.

1. Did any framework or model was proposed for the SE Curriculum?
2. Are there any benefits of software reusability?

4- **Description**: this class consists of questions which need descriptive answers, explanations or discussions regarding a topic or a definition or a brief explanation of an exact entity.

a. How can the current researches on software architecture optimization been classified?
b. What is a cloud-ready application and how it differs from conventional applications?

5- **Comparison**: this class contains questions that making comparison between entities or finding relationship between entities.

What is the relationship between the user involvement in system development and the system success?

6- **List**: This class contains questions that expect several key points about the subject matter or a list of entities, and it can be one of the following

a. ***Sources***: What are the sources of uncertainty perceived?
b. ***Practices***: What are the adopted practices of agile RE according to published empirical studies?
c. ***Challenges***: What are the challenges of traditional RE that may get alleviated by agile RE?
d. ***Factors***: What factors influence on requirements elicitation and how?
e. ***Methods***: What empirical research methods and data analysis techniques have been employed?
f. ***Activities***: What activities of the requirements elicitation process have been covered by the different proposals?
g. ***Limitations***: What are the limitations of the current research?
h. ***Evidences***: What are the existing empirical evidences for various approaches followed by the RTP techniques?
i. ***Approaches***: Which search-based approaches were used in collaboration with mutation testing?
j. ***Techniques***: What SPM techniques have been used or evaluated?
k. ***Characteristics***: What are the main characteristics of current Agile MDD approaches?

l. ***Issues***: What are the current issues facing the researchers in data carving?
m. ***Strategies***: what type of 'customer first' strategies are used in software oriented companies?
n. ***Areas***: What are the areas of application in which search based approaches were applied for mutation testing?
o. ***Topics***: what are the topics addressed by these challenges?
p. ***Artifacts***: What types of software artifacts are used as input sources for knowledge extraction?
q. ***Effects***: what are the effects of erp implementation towards it activities?
r. ***Causes***: what are the different causes of software requirement change?
s. ***Others***: this class contains questions that do not mach any of the above list types.
   - What implications can be derived for the industrial and research communities from the findings?
   - Which domains are covered by SA studies?

III. APPROACHES FOR QUESTION TYPE IDENTIFICATION

As in [20] the question classification approach requires the presence of four aspects: taxonomy corpus or dataset, a classification algorithm and a feature set.

3.1 Features Used for question type identification

In the classification of questions problem, the features can be in three categories:

3.1.1 Lexical features

It refers to the features that can be extracted directly from the question based on the context of the question or the words which appear in a question. In the classification process the question is represented in *vector space model* which means a question is a vector which is described by the words inside it.

- **Bag of words or N-gram**
  To extract n-gram features, any n consecutive words in a question are considered as a feature.
- **Wh-word**
  Authors in [15] consider question wh-words as a separate feature. They adapted 8 types of wh- words, namely what, which, when, where, who, how, why and rest.
- **Limited bigram**
  In n-gram when n=2 it is called bigram, the feature limited bigram means the wh-word and the immediate word next to it.
- **Word shape**
  It refers to apparent properties of single word,[15] proposes five types of shapes namely: all digit, lower case, upper case, mixed, and others

3.1.2 Syntactical features

Syntactical features are the features that can be extracted from the grammatical composition of the question.

- **Tagged unigrams**
  This feature introduced by [21], it means the unigrams augmented with POS tags, using this feature can help the classifier to distinguish a word with different tags as two different features. Also POS tags information can be used for extracting semantic features; POS tags can be used to disambiguate the meaning of a word to extract semantic features [21].
- **Head words:**
  Which is one single word specifying the object that the question seeks [15]. Identifying the head word correctly can improve the accuracy since it is the most informative word in the question [21], usually it is extracted based on the syntax structure of the question [22]

Head word extraction

The idea of headword extraction from syntax tree first was introduced by [23]. He

proposed some rules, known as Collins rules, to identify the headword of sentence. To find the headword of a sentence, the parse tree is traversed top-down and in each level the subtree which contains the headword is identified with rules. The algorithm continues on the resulting subtree until it reaches a terminal node. The resulting node is the sentence head word.

In question classification the use of these rules is not suitable since they have preferences for verb phrases over noun phrases whereas in a question the headword should be a noun. The author in [24] modified the rules in [23] to properly extract a question's headword. In fact, in the modified rules he set a preference of noun phrases over verb phrases.

3.1.3 Semantic features

Semantic features are the features that can be extracted based on the semantic meaning of the question.

The semantic feature used here is the Hypernym of head word. For a given word, a hypernym is a word with a more general meaning. As hypernyms allow one to abstract over specific words, they may be useful features for question classification [24].We used WordNet to extract hypernyms, and as in [24] we extract hypernyms to 6 dept.

3.2 Answer identification

Answer identification can be either via hand-crafted rules or supervised machine learning techniques. In either case, the ranking is based on a relatively small set of features that can be easily and efficiently extracted from a potentially large number of answer passages, which have been extracted from a pool of documents using keywords extracted from user question. Among the more common features are:

- The number of **named entities** of the right type in the passage [31] [32]
- Question term numbers matched in the answer passage. [28][29][30][31][32]
- Question terms numbers matched in the same phrase or sentence as the candidate answer. [28][29]
- Number of question terms matched, separated from the candidate. [28][29]
- Number of terms occurring in the same order in the answer passage as in the question. [28][29][30][5]
- Average distance from the candidate answer to the question term matches. [28][29]
- The ***N*-gram overlap** between the passage and the question: Count the *N*-grams in the question and the *N*-grams in the answer passages. Prefer the passages with higher *N*-gram overlap with the question [31] [32]

IV. EXPERIMENTAL DESIGN

Dataset: 260research questions (RQs) for training, 100 RQs for testing collected from 131 Systematic literature review (SLR) papers in software engineering field, each question was manually labeled with the proper question class. We used two different classifiers in this work: Support Vector Machines (SVMs) and Naïve bayes.

Naïve Bayes classifier

A Naive Bayes classifier is a simple probabilistic classifier based on applying Bayes' theorem with strong independence assumptions [25]. Bayes theorem can be stated as follows

$$P(c_k / q_j) = \frac{P(c_k) * P(q_j / c_k)}{P(q_j)}$$

Where P(Ck|qj) is the posterior probability,
P(Ck) is the prior probability,
P(qj| Ck) is the likelihood and P(qj) is the evidence

A naive Bayes classifier follows conditional independence since it assumes that the presence (or absence) of a particular feature of a class is unrelated to the presence (or absence) of any other feature, given the class variable. Thus terms are given a weight value which is independent of its position and presence of other terms. Naive Bayes classifier is trained by set of labeled training examples. Thus it is said to undergo supervised learning. we used Naïve Bayes java implementation done by the author in [24].

Support vector machine classifier

Support vector machine is a supervised learning method for classifying data. It is especially successful

for high dimensional data. SVM is a linear discriminant model which tries to learn a hyperplane with maximum margin for separating the classes.

Suppose we are given a training set (x_i, y_i), i = 1, ...,n, in which $x_i = (x_{i1}, ..., x_{id})$ is a d-dimensional sample and $y_i \in \{1,-1\}$ is the corresponding label. The task of a support vector classifier is to find a linear discriminant function $g(x) = w^T x + w0$, such that $w^T\mathbf{x}_i + w0 \geq +1$ for $y_i = +1$ and $w^T\mathbf{x}_i + w0 \leq -1$ for $y_i = -1$. Therefore we seek for a solution such that the following condition holds:

$$y_i\,(w^T\mathbf{x}_i + w0) \geq 1 \quad i = 1, ..., n$$

The optimal linear function is obtained by minimizing the following quadratic programming problem [27]:

$$\min \ \frac{1}{2}w^T w - \sum_{i=1}^{n} \alpha_i (y_i(w^T \mathbf{x}_i + w_0) - 1)$$

Which leads to the following solution:

$$w = \sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i$$

Where $\{\alpha_i,\ i = 1, ..., n;\ \alpha_i \geq 0\}$ are Lagrange multipliers. To be able to linearly separate data, typically the feature space should be mapped to a higher dimensional space. The mapping is done with a so-called *kernel function*. There are four types of basic kernel functions: linear, polynomial, radial basis function and sigmoid.

By default SVM is a binary classifier. But it can be utilized for multiclass classification by using one-vs-all like methods. For the task we use openly available LIBSVM toolkit[26]. And the features used are mentioned in table 3.

V. RESULTS AND DISCUSSION

From table3 and figure1 the classifiers obtained a very good result when combining the three types of features and SVM results are better than naïve bayes results.

TABLE3: ACCURACY OF SVM AND NAÏVE BAYES CLASSIFIERS WHEN USING DIFFERENT TYPES OF *FEATURES*

Features		Accuracy	
		SVM	Naïve Bayes
lexical features	Unigram	94%	77%
	Unigram+wh-word	96%	84%
	Unigram+limited bigram	96&	83%
	Unigram+word shape	97%	81%
Syntactic features	Tagged unigrams	92%	77%
	Tagged unigrams+Head words	95%	78%
Lexical, Syntactic and semantic features	Unigram+wh-word+limited bigram+word shape+tagged unigrams+hypernyms	98%	79%

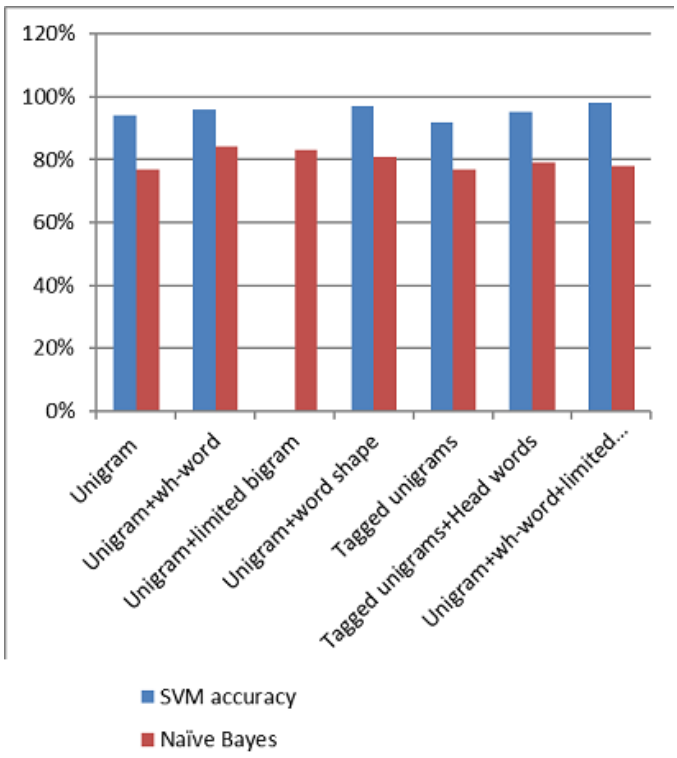

Fig1: accuracy of SVM and Naïve Bayes classifiers when using different types of features

To check the results stability we used a well know technique known as cross validation. We applied this technique on our dataset. The total amount of questions is 360(training and testing), we divided it into four sets of 90 questions each to experiment 4-folds cross validation on the data set. Table4 shows the mean of the 4-folds cross validation using SVM classifier based on a combination of the three types of features.

TABLE4: CROSS VALIDATION RESULT USING A COMBINATION OF *LEXICAL, SYNTACTICAL AND SEMANTIC FEATURES* :(SVM CLASSIFIER)

Fold	Accuracy
1	94.4
2	97.7
3	93.3
4	96.5
Mean	**95.47**

As noted above the best classification results (98%) were obtained by the support vector machine classifier which is better than the one obtained by [21] which is (93.2) using SVM classifier. We used the idea proposed by [21] but we got better result, because we are using a limited number of questions and all questions from the same domain.

VI. CONCLUSION

A systematic literature review (SLR) process commonly involves a large set of data to be analyzed and interpreted in order to answer research questions (RQs). In this paper we propose a set of expected answer types (Taxonomies) derived from the collected dataset. We used two types of classifiers to classify questions into its corresponding answer type. The classifier model is built and tested using lexical, syntactic and semantic features. We got the best result (98%) when using a combination of the three types of features.

The result of our work can be used to increases the accuracy of answer extraction when performing SLR process and hence accelerating and reducing the time needed in answer extraction.

REFERENCES

[1] B. A. Kitchenham, T. Dyba, and M. Jorgensen, “Evidence-based software engineering,” in *Proceedings. 26th International Conference on Software Engineering*, 2004, pp. 273–281.

[2] B. Kitchenham and S.Charters, “Guidelines for performing systematic literature reviews in software engineering,” Keele University and University of Durham, ESBE Technical Report 2.3, July 2007.

[3] H. Zhang et al., "An Empirical Assessment of A Systematic Search Process for Systematic Reviews", 2011.

[4] H. Zhang and M.Babar,"On Searching Relevant Studies in Software Engineering",2010.

[5] M.Ghafari et al.,"A Federated Search Approach To Facilitate Systematic Literature Review In Software Engineering",2012.

[6] Bekhuis, T., Demner-Fushman, D. "Towards automating the initial screening phase of a systematic review". *Studies in health technology and informatics*, *160*(Pt 1), 146–50,2010.

[7] Wallace, B. C., Trikalinos, T. a, et al.,"Semi-automated screening of biomedical citations for systematic reviews". *BMC bioinformatics*,2010.

[8] Ananiadou, S., & Rea, B. ,"Supporting systematic reviews using text mining",2009.

[9] Cohen, A. M., Adams, C. E., et al.,"the Essential Role of Systematic Reviews , and the Need for Automated Text Mining Tools",2010

[10] Cohen, A. M., Ambert, K., et al.," Cross-topic learning for work prioritization in systematic review creation and update". *Journal of the American Medical Informatics Association : JAMIA*, *16*(5), 690–704

[11] R. Mudgal, R. Madaan, and A. Dixit, "a Novel Architecture for Question Classification Based," vol. II, no. Ii, pp. 27–43.

[12] X. Li and D. Roth, "Learning question classifiers," *Proc. 19th Int. Conf. Comput. Linguist.* -, vol. 1, pp. 1–7, 2002.

[13] X. Li and D. Roth, "Learning question classifiers: the role of semantic information", *Natural Language Engineering*, *12*(03), 229,2006.

[14] N. Van-Tu and L. Anh-Cuong, "Improving Question Classification by Feature Extraction and Selection," *Indian J. Sci. Technol.*, vol. 9, no. 17, 2016.

[15] Z. Huang, M. Thint, and Z. Qin, "Question Classification using Head Words and their Hypernyms," *Proc. Conf. Empir. Methods Nat. Lang. Process. (EMNLP 2008)*, no. October, pp. 927–936, 2008.

[16] S. Quarteroni, A. Moschitti, S. Manandhar, and R. Basili, "Advanced structural representations for question classification and answer re-ranking," *Adv. Inf. Retr.*, vol. 4425, pp. 234–245, 2007.

[17] S. K. Mishra, P. Kumar, and S. K. Saha, "A Support Vector Machine Based System for Technical Question Classification," 2002.

[18] F. Bu, X. Zhu, Y. Hao, and X. Zhu, "Function-based question classification for general QA," *Acl*, no. October, pp. 1119–1128, 2010.

[19] J. Bullington, I. Endres, and M. A. Rahman, "Open-Ended Question Classification Using Support Vector Machines," *Science (80-.).*, 2007.

[20] H. Kahaduwa, D. Pathirana, and P. Liyana, "Question Answering System for the Travel Domain," pp. 449–454, 2017.

[21] B. Loni, G. Van Tulder, P. Wiggers, D. M. J. Tax, and M. Loog, "Question classification by weighted combination of lexical, syntactic and semantic features," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2011, vol. 6836 LNAI, pp. 243–250.

[22] J. Silva, L. Coheur, A. C. Mendes, and A. Wichert, "From symbolic to sub-symbolic information in question classification," *Artif. Intell. Rev.*, vol. 35, no. 2, pp. 137–154, 2011.

[23] M. Collins,"Head-Driven Statistical Models for natural Language Parsing", PhD thesis, University of Pennsylvania, 1999

[24] B. Loni, "Enhanced question classification with optimal combination of features", Master thesis, Department of Media and Knowledge Engineering, Faculty of Electronic Engineering, Mathematics and Computer Science Delft University of Technology, 2011.

[25] T. MITCHELL. Machine Learning. McGraw-Hill, New York,NY., 1997.

[26] Chang, C.-C., Lin, C.-J.: LIBSVM: a library for support vector machines (2001) Software, http://www.csie.ntu.edu.tw/˜cjlin/libsvm

[27] N. Vladimir and Vapnik. The nature of statistical learning theory. Springer-Verlag New York, Inc., New York, NY, USA, 1995.

[28] R. P. Kamdi and A. J. Agrawal, "Keywords based Closed Domain Question Answering System for Indian Penal Code Sections and Indian Amendment Laws," *Int. J. Intell. Syst. Appl.*, vol. 7, no. 12, pp. 57–67, 2015.

[29] F. Benamara, "Question Answering Systems: State of the Art and Future Directions," *LAICS-NLP Summer Sch. Lang. Artif. Intell. Comput. Sci. Nat. Lang. Process.*, no. October, 2006.

[30] Hong Sun et. al. ," Answer Extraction from Passage Graph for Question Answering", In Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence, pp. 2169-2175, August 2013.

[31] D. Jurafsky and J. H. Martin, "Question Answering and Summarization," *Speech Lang. Process.*, p. 1024, 2008.

[32] N. Dharmasiri, B. Gunathilake, U. Pathirana, S. Senevirathne, A. Nugaliyadde, S. Thellijagoda, and S. Lanka, "SIMPLIFYING LAW STATEMENTS USING," 2017.