Domain Specific Grammar based Classification for Factoid Questions

Alaa Mohasseb[a], Mohamed Bader-El-Den[b] and Mihaela Cocea[c]
School of Computing, University of Portsmouth, Portsmouth, U.K.

Keywords: Information Retrieval, Question Classification, Factoid Questions, Grammatical Features, Machine Learning.

Abstract: The process of classifying questions in any question answering systems is the first step in retrieving accurate answers. Factoid questions are considered the most challenging type of question to classify. In this paper, a framework has been adapted for question categorization and classification. The framework consists of three main features which are, grammatical features, domain-specific features, and grammatical patterns. These features help in preserving and utilizing the structure of the questions. Machine learning algorithms were used for the classification process in which experimental results show that these features helped in achieving a good level of accuracy compared with the state-of-art approaches.

1 INTRODUCTION

The process of classifying questions in any question answering systems is the first step in retrieving accurate answers. Question classification performed in question answering systems affects the answers and most errors happen due to miss-classification of the questions (Moldovan et al., 2003). The task of generating answers to the users' questions is directly related to the type of questions asked. Several question taxonomies have been proposed, the most popular classification taxonomy of factoid ('wh-') questions is Li and Roth's categories (Li and Roth, 2006), which consists of a set of six coarse-grained categories and fifty fine-grained ones. Developing an accurate question classifier requires using the accurate question features such as linguistics features (Le-Hong et al., 2015). In addition, it is found more challenging to classify 'wh-' questions into proper semantic categories more than other types in question answering systems (Li et al., 2008).

In this paper, a grammar-based framework for questions categorization and classification is adapted (Mohasseb et al., 2018). The framework is applied to question classification according to Li and Roth's (Li and Roth, 2006) categories of intent, using domain specific grammatical features and patterns. These features have the advantage of preserving the grammatical structure of the question. The aim is to assess the influence of using the structure of the question and the domain-specific grammatical categories and features on the classification performance. To achieve this aim, the following objectives are defined:

1. Investigate the influence of using domain-specific grammatical features on the classification performance;
2. Compare the performance of different machine learning algorithms for the classification of factoid questions intent;
3. Investigate the classification performance in comparison with state-of-the art approaches.

The rest of the paper is organised as follows. Section 2 outlines the categorization of questions and the previous work in question classification. Section 3 describes the proposed approach and the grammatical features used. The experimental setup and results are presented in Section 4, while the results are discussed in Section 5. Finally, Section 6 concludes the paper and outlines directions for future work.

2 BACKGROUND

In this section we review previous work on question classification. Different proposed question categories are outlined in Section 2.1, while Section 2.2 reviews previous work on question classification methods.

[a] https://orcid.org/0000-0003-2671-2199
[b] https://orcid.org/0000-0002-1123-6823
[c] https://orcid.org/0000-0002-5617-1779

Mohasseb, A., Bader-El-Den, M. and Cocea, M.
Domain Specific Grammar based Classification for Factoid Questions.
DOI: 10.5220/0007958601770184
In *Proceedings of the 15th International Conference on Web Information Systems and Technologies (WEBIST 2019)*, pages 177-184
ISBN: 978-989-758-386-5

2.1 Questions Categories

Different questions categories were proposed. In (Kolomiyets and Moens, 2011) authors classified questions to eight types which are: list, definition, factoids, causal, relationship, hypothetical, procedural, and confirmation questions. Authors in (Bu et al., 2010) proposed six question categories which were tailored to general QA, namely: fact, list, reason, solution, definition and navigation. Furthermore, in (Bullington et al., 2007) authors classified questions to 11 categories, which are: Advantage/Disadvantage, Cause and Effect, Comparison, Definition, Example, Explanation, Identification, List, Opinion, Rationale, and Significance. In (Mohasseb et al., 2018) questions were classified and labeled to six different categories, which are; causal, choice, confirmation (Yes-No questions), factoid (Wh-questions), hypothetical and list. This classification is based on the question types asked by the users and the answers given and categorization was motivated by the question types in English.

The most famous factoid question type taxonomy is the one of Li and Roth (Li and Roth, 2006). Many researchers focused on Li and Roth classification of question (Li et al., 2005), (Metzler and Croft, 2005), (Van-Tu and Anh-Cuong, 2016), (Xu et al., 2016), (Le-Hong et al., 2015), (May and Steinberg, 2004),(Zhang and Lee, 2003), (Mishra et al., 2013), (Huang et al., 2008), (Nguyen et al., 2008), (Nguyen and Nguyen, 2007), (Li et al., 2008). Their two-layer taxonomy consists of a set of six coarse-grained categories which are abbreviation, entity, description, human, location and numeric value, and fifty fine-grained ones, e.g., Abbreviation, Description, Entity, Human, Location and Numeric as coarse classes, and Expression, Manner, Color and City as fine-grained classes.

2.2 Question Classification Methods

In this section, we review related work on question classification methods and machine learning algorithms using Li and Roth's question categories which is adopted in this study as it is the most widely used question intent taxonomy in the literature.

Authors in (Li et al., 2005) used compositive statistic and rule classifiers combined with different classifiers and multiple classifier combination methods. Moreover, many features such as dependency structure, wordnet synsets, bag-of-words, and bi-gram were used with a number of kernel functions. In (Van-Tu and Anh-Cuong, 2016) a method was proposed using feature selection algorithm to determine appropriate features corresponding to different question types. Authors in (Metzler and Croft, 2005) proposed a statistical classifier based on SVM which learns question word by using prior knowledge about correlations between question words and types. Furthermore, a SVM-based approach for question classification was proposed in (Xu et al., 2016). The proposed approach incorporates dependency relations and high-frequency words. In (Mishra et al., 2013) question classification method was proposed using three different classifiers, k-Nearest neighbor, Naive Bayes, and SVM. In addition, features such as using bag-of-words and bag-of-ngrams were used and a set of lexical, syntactic, and semantic features. Moreover, authors in (Zhang and Lee, 2003) used five machine learning algorithms which are, k-Nearest neighbor, Naive Bayes, Decision Tree, Sparse Network of Winnows, and SVM. In addition, two features were used; bag-of-words and bag-of-ngrams. In (Huang et al., 2008) authors adapted Lesk's word sense disambiguation algorithm and the depth of hypernym feature is optimized with further augment of other standard features such as unigrams. Moreover, authors in (Le-Hong et al., 2015) proposed a compact feature set that uses typed dependencies as semantic features. A hierarchical classifier was designed in (May and Steinberg, 2004). In addition, different classifiers has been used such as SVM, MaxEnt, Naive Bayes and Decision Tree for primary and secondary classification. In (Nguyen et al., 2008) authors used unlabeled questions in combination with labeled questions for semi-supervised learning. In addition, Tri-training were selected to improve the precision of question classification task. In addition, a two-level hierarchical classifier for question classification was proposed in (Nguyen and Nguyen, 2007) such as supervised and semi-supervised learning. Finally, in (Li et al., 2008) authors classified what-type questions by head noun tagging. In addition, different features such as local syntactic feature, semantic feature and category dependency were integrated among adjacent nouns with Conditional Random Fields to reduce the semantic ambiguities of head noun.

3 PROPOSED APPROACH

3.1 Factoid Questions Grammatical Features

This analysis was first introduced in (Mohasseb et al., 2018). Wh-questions (factoid) has its own characteristics, features, and structure that help in the identifi-

cation and the classification process.

The main feature of a factoid question (Wh-Questions) is the presence of question words, this kind of question starts with a question word, such as *What, Where, Why, Who, Whose, When, Which*, e.g. *"What did the only repealed amendment to the U.S. Constitution deal with ?"*. In addition, this question could start with question words that do not start with "wh" such as *how, how many, how often, how far, how much, how long, how old*, e.g. *"How long does it take light to reach the Earth from the Sun ?"*. In addition, the structure of this type of question could begin with a preposition followed by a question, *"P + QW"* such as *"In what year did Thatcher become prime minister?" OR "At what age did Rossini stop writing opera?"*. Also in many cases the question word could be found in the middle of the question, e.g. *"The corpus callosum is in what part of the body ?"*.

Most factoid questions are related to facts, current events, ideas and suggestions and could formulate an advice question, e.g. *"How do you make a paintball ?"*. In addition, some factoid questions could contain two types of question words, for example *"What does extended definition mean and how would one write a paper on it ?"*. Furthermore, factoid questions could have any kind of information given as an answer or response.

3.2 Question Classification Features

In a previous study (Mohasseb et al., 2018), a Grammar-based framework for Questions Categorization and Classification (GQCC) was proposed. In this study, the framework is applied to question classification, according to Li and Roth's (Li and Roth, 2006) categories of intent. Three main features which are, (1) Grammatical Features, (2) Domain specific Features and (3) Grammatical Pattern Features have been used for the analysis and classification of factoid questions. These features help in transforming the questions into a new representation which has the advantage of preserving the grammatical structure of the question. The used features are discussed in more details in the following sections.

3.2.1 Grammatical Features

The main objective of the grammatical features (Mohasseb et al., 2018), is to transform the questions (by using the grammar) into a new representation as a series of grammatical terms, i.e. a grammatical pattern. The grammatical features consist of the seven major word classes in English, which are Verb (V), Noun (N), Determiner (D), Adjective (Adj), Adverb (Adv) , Preposition (P) and Conjunction $(Conj)$ in addition to the six main question words: How (QW_{How}), Who (QW_{Who}), When (QW_{When}), Where (QW_{Where}), What (QW_{What}) and Which (QW_{Which}). Some word classes like Noun can have sub-classes, such as Common Nouns (CN), Proper Nouns (PN), Pronouns $(Pron)$, and Numeral Nouns (NN) as well as Verbs, such as Action Verbs (AV), Linking Verbs (LV) and Auxiliary Verbs $(AuxV)$. In addition, the grammatical features consist of other features such as singular (e.g. Common Noun – Other- Singular (CN_{OS})) and plural terms (e.g. Common Noun- Other- Plural (CN_{OP}))

3.2.2 Domain Specific Grammatical Features

Domain-specific features (i.e. related to question-answering) where identified, which correspond to topics (Mohasseb et al., 2018) – these are listed in Table 1. Instead of further classifying the question to fine grained which is based on a large number of categories, we have used domain specific features to determine the type of question. For example, question type *ENTY* consists of fine grained categories such as religion, disease/medicine, event, product. These type could be identified using the following domain specific grammatical features: religion = religious terms PN_R, disease/medicine = health terms CN_{HLT} and PN_{HLT}, product = Products PN_P, event = events PN_E. Hence the domain specific grammatical features contain less categories than the fine grained categories proposed by Li and Roth but still could identify the different coarse categories.

Table 1: Domain Specific Grammatical Features.

Domain specific Features	Abbr.
Celebrities Name	PN_C
Entertainment	PN_{Ent}
Newspapers, Magazines, Documents, Books	PN_{BDN}
Events	PN_E
Companies Name	PN_{CO}
Geographical Areas	PN_G
Places and Buildings	PN_{PB}
Institutions, Associations, Clubs, Foundations and Organizations	PN_{IOG}
Brand Names	PN_{BN}
Software and Applications	PN_{SA}
Products	PN_P
History and News	PN_{HN}
Religious Terms	PN_R
Holidays, Days, Months	PN_{HMD}
Health Terms	PN_{HLT}
Science Terms	PN_S
Database and Servers	CN_{DBS}
Advice	CN_A
Entertainment	CN_{Ent}
History and News	CN_{HN}
Site, Website, URL	CN_{SWU}
Health Terms	CN_{HLT}

3.2.3 Grammatical Patterns

The main objective of the question grammatical pattern help is the identification of the question type, since each factoid question type has a certain structure (grammatical pattern). For example, the following question which represent (HUM) type of question *"Who killed Martin Luther King?"* has the following grammatical pattern $QW_{Who} + AV + PN_C$. While, the question which represent (LOC) type of question *"What is the capital of Italy?"* has the following grammatical pattern $QW_{What} + LV + D + CN_{OS} + P + PN_G$. The different pattern representations help in distinguishing between different factoid question type. A full description of how these features are used is provided in the following sections

3.3 Question Classification Framework

The question classification framework takes into account the grammatical structure of the questions and combines grammatical features with domain-related information and grammatical patterns. The framework consists of three main phases; (1) Question Parsing and Tagging, (2) Pattern Formulation and (3) Question Classification. The following question from Li and Roth datasets will be used *"Where are the Rocky Mountains?"* to illustrate how these phases work.

(1) Question Parsing and Tagging: this step is mainly responsible for extracting user's question terms. The system simply takes the question and parses to tag each term in the question to its terms' category. In this phase parsing the keywords and phrases is done by; first parsed compound words then single words. In addition, the term tagging is done by tagging each term to its grammar terminals; each term will be tagged to its highest level of abstraction (domain specific).

For the given example the question will be parsed and tagged as follow:

Question: "Where are the Rocky Mountains?"

Terms extracted: Where, are, the, Rocky Mountains

After parsing, each term in the question will be tagged to one of the terms category using tag-set the was proposed by (Mohasseb et al., 2014) and (Mohasseb et al., 2018). The final tagging will be:

Question Terms Tagging: Where= QW_{Where}, are= LV, the= D, Rocky Mountains= PN_G

(2) Pattern Formulation: in this phase after parsing and tagging each term in the question, the pattern is formulated. This is done by matching the question with the most appropriate question pattern to help facilitate the classification processing and the identification of the factoid question type in the next phase.

For the given example, the following pattern will be formulated:

Question Pattern: $QW_{Where} + LV + D + PN_G$

(3) Question Classification: This phase is done by using the patterns generated in Phase (2), the aim of this phase is to build a model for automatic classification. The classification is done by following the standard process for machine learning, which involves the splitting of the dataset into a training and a testing dataset. The training dataset is used for building the model, and the test dataset is used to evaluate the performance of the model.

For the given example, the question will be classified to the following question type.

Question Type: LOC

4 EXPERIMENTAL STUDY AND RESULTS

In the experimental study we investigate the ability of machine learning classifiers to distinguish between different question types based on grammatical features and question patterns. Two machine learning algorithms, were used for question classification; Support Vector Machine (SVM) and J48. We used 1000, 2000 and 3000 questions that were selected from Li and Roth [1]. Their distribution is given in Table 2. Questions in the dataset are classified into two categories; coarse and fine, in this experiment coarse categories have been used.

Table 2: Data distribution.

Question Type	1000	2000	3000
ABBR	18	30	45
DESC	211	419	655
ENTY	244	486	710
HUM	220	442	655
LOC	156	312	457
NUM	151	311	478

To assess the performance of proposed features and the machine learning classifiers two experiments have been conducted (1) using our proposed features using the Weka[2] software and (2) using the most used features in previous works such as n-gram, Bag-of Words, Snowball Stemmer and stop word remover using Knime [3] software. The experiments were set up

[1] http://cogcomp.org/Data/QA/QC/

[2] http://www.cs.waikato.ac.nz/ml/weka/

[3] https://www.knime.com

using the typical 10-fold cross validation. The results are presented in the next sub-section.

4.1 Results

In this section we present and analyse the results of the machine learning algorithms for each of the three set of questions. Table 3 shows the accuracy for GQCC and n-gram based classifiers for the 1000, 2000 and 3000 questions. In following sections the results will be discussed in more details.

Table 3: Accuracy of GQCC and n-gram based classifiers for 1000, 2000 and 3000 questions.

Questions	**GQCC$_{SVM}$**	**n-gram$_{SVM}$**	**GQCC$_{J48}$**	**n-gram$_{J48}$**
1000	92.6%	87%	95.5%	86.7%
2000	95.1%	89.3%	96.6%	88.9%
3000	95.5%	92.4%	95.8%	91.1%

4.1.1 1000 Questions

Table 4 presents the classification performance details (Precision, Recall and F-Measure) of the classifiers that have been used SVM and J48 using the proposed grammatical features, while Table 5 presents the classification performance details (Precision, Recall and F-Measure) of SVM and J48 using features such as n-grams, punctuation eraser, stop-word remover and snowball stemmer. Results show that Decision Tree (GQCC$_{J48}$) identified correctly (i.e. Recall) 95.5% of the questions and GQCC$_{SVM}$ identified correctly 92.6% of the questions when grammatical features were used while Decision Tree (n-gram$_{J48}$) identified correctly (i.e. Recall) 86.7% of the questions and n-gram$_{SVM}$ identified correctly 87.3% of the questions when features such as n-grams and snowball Stemmer were used.

Comparing the performance of the classifiers when 1000 questions were evaluated, GQCC$_{J48}$ had a better performance than GQCC$_{SVM}$, in which GQCC$_{J48}$ has the highest precision, recall and f-measure for all the classes and GQCC$_{SVM}$ has a similar precision (100%) as GQCC$_{J48}$.

When comparing the performance of n-gram$_{J48}$ and n-gram$_{SVM}$, n-gram$_{SVM}$ has a better precision, recall and f-measure for classes such as ABBR and NUM. In addition, both classifiers have similar precision, recall and f-measure for class type HUM. For class type DESC and ENTY n-gram$_{SVM}$ has a recall of (100%) while n-gram$_{J48}$ has better precision and f-measure. Furthermore, comparing the classification performance of GQCC$_{SVM}$ and n-gram$_{SVM}$. GQCC$_{SVM}$ has better precision and f-measure for class type DESC and ENTY while n-gram$_{SVM}$ has better recall (100%). For class type HUM and LOC, GQCC$_{SVM}$ has better Recall and n-gram$_{SVM}$ has better precision and f-measure. In addition, n-gram$_{SVM}$ has a (100%) precision, recall and f-measure for class type ABBR and higher precision, recall and f-measure than GQCC$_{SVM}$ for class type NUM. Comparing GQCC$_{J48}$ and n-gram$_{J48}$, GQCC$_{J48}$ has a (100%) precision, recall and f-measure for class type ABBR, DESC and HUM. For class type ENTY, GQCC$_{J48}$ has higher recall and f-measure while n-gram$_{J48}$ has higher precision. While for class type LOC, GQCC$_{J48}$ has better precision and f-measure and n-gram$_{J48}$ has better Recall.

Table 4: Performance of the classifiers using grammatical features (1000 questions) - Best results are highlighted in bold.

	GQCC$_{SVM}$			**GQCC$_{J48}$**		
Class:	P	R	F	P	R	F
ABBR	**1.000**	0.833	0.909	**1.000**	**1.000**	**1.000**
DESC	0.995	0.995	0.995	**1.000**	**1.000**	**1.000**
ENTY	0.845	0.893	0.869	**0.873**	**0.955**	**0.912**
HUM	0.995	0.995	0.995	**1.000**	**1.000**	**1.000**
LOC	0.848	0.821	0.834	**0.936**	**0.840**	**0.885**
NUM	0.848	0.821	0.834	**0.986**	**0.940**	**0.963**

Table 5: Performance of the classifiers using n-gram features (1000 questions) - Best results are highlighted in bold.

	n-gram$_{SVM}$			**n-gram$_{J48}$**		
Class:	P	R	F	P	R	F
ABBR	**1.000**	**1.000**	**1.000**	**1.000**	0.800	0.889
DESC	0.887	**1.000**	0.940	**0.911**	0.984	**0.947**
ENTY	0.712	**1.000**	0.831	**0.965**	0.743	**0.840**
HUM	**1.000**	**0.788**	**0.881**	**1.000**	**0.788**	**0.881**
LOC	**1.000**	0.553	0.712	0.605	**0.978**	**0.748**
NUM	**1.000**	**0.933**	**0.966**	0.953	0.911	0.932

4.1.2 2000 Questions

Table 6 presents the classification performance details (Precision, Recall and F-Measure) of the classifiers that have been used SVM and J48 using the proposed grammatical features, while Table 7 presents the classification performance details (Precision, Recall and F-Measure) of SVM and J48 using features such as n-grams, punctuation eraser, stop-word remover and snowball stemmer. Results show that Decision Tree (GQCC$_{J48}$) identified correctly (i.e. Recall) 96.6% of the questions and GQCC$_{SVM}$ identified correctly 95.1% of the questions when grammatical features were used while Decision Tree (n-gram$_{J48}$) identified correctly (i.e. Recall) 88.8% of the questions and n-gram$_{SVM}$ identified correctly 89.3% of the questions when features such as n-grams and snowball Stemmer were used.

When 2000 questions were evaluated, GQCC$_{J48}$ outperformed GQCC$_{SVM}$ in terms of precision, re-

call and f-measure for classes such as ABBR, ENTY, LOC and NUM. While, for classes such as DESC and HUM both classifiers had (100%) recall. For n-gram based classifiers n-gram$_{SVM}$ had better precision, recall and f-measure for classes such as ABBR, NUM. While, n-gram$_{J48}$ had better performance for class type ENTY. In addition, for class type DESC n-gram$_{SVM}$ had better recall while n-gram$_{J48}$ had better precision and f-measure. On the other hand, for class type LOC, n-gram$_{SVM}$ had better precision while n-gram$_{J48}$ had better recall and f-measure. Comparing the performance of GQCC$_{SVM}$, GQCC$_{J48}$ and n-gram$_{SVM}$, n-gram$_{J48}$. GQCC$_{SVM}$ had a better precision, recall and f-measure for class type ABBR, DESC and HUM. while, n-gram$_{SVM}$ had better precision, recall and f-measure for class type NUM. Moreover, n-gram$_{SVM}$ has better precision and f-measure for class type ENTY. While, n-gram$_{SVM}$ has better recall. On the other hand, for class type LOC GQCC$_{SVM}$ has better recall and f-measure while n-gram$_{SVM}$ has higher precision. Comparing GQCC$_{J48}$ and n-gram$_{J48}$, GQCC$_{J48}$ has better performance in terms of precision, recall and f-measure for classes such as ABBR, DESC, HUM and NUM. While for class type ENTY GQCC$_{J48}$ has higher recall and f-measure and n-gram$_{J48}$ has better precision. On the other hand, for class type LOC GQCC$_{J48}$ has better precision and f-measure while n-gram$_{J48}$ has higher recall.

Table 6: Performance of the classifiers using grammatical features (2000 questions).

	GQCC$_{SVM}$			**GQCC$_{J48}$**		
Class:	P	R	F	P	R	F
ABBR	**1.000**	0.967	0.983	**1.000**	**1.000**	**1.000**
DESC	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**
ENTY	0.915	0.903	0.909	0.954	0.936	0.945
HUM	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**
LOC	0.859	0.901	0.879	**0.881**	**0.929**	**0.905**
NUM	0.960	0.936	0.948	**0.977**	**0.952**	**0.964**

Table 7: Performance of the classifiers using n-gram features (2000 questions) - Best results are highlighted in bold.

	n-gram$_{SVM}$			**n-gram$_{J48}$**		
Class:	P	R	F	P	R	F
ABBR	**1.000**	**0.889**	**0.941**	**1.000**	**0.889**	**0.941**
DESC	0.920	**1.000**	0.958	0.933	**0.992**	**0.962**
ENTY	0.777	0.979	0.867	**0.959**	**0.795**	**0.869**
HUM	**0.973**	**0.827**	**0.894**	**0.973**	0.820	0.890
LOC	**0.896**	0.645	0.750	**0.650**	0.957	**0.774**
NUM	**0.978**	**0.957**	**0.967**	0.977	0.925	0.950

4.1.3 3000 Questions

Table 8 presents the classification performance details (Precision, Recall and F-Measure) of the classifiers that have been used SVM and J48 using the proposed grammatical features, while Table 9 presents the classification performance details (Precision, Recall and F-Measure) of SVM and J48 using features such as n-grams, punctuation eraser, stop-word remover and snowball stemmer. Results show that Decision Tree (GQCC$_{J48}$) identified correctly (i.e. Recall) 95.8% of the questions and GQCC$_{SVM}$ identified correctly 95.5% of the questions when grammatical features were used while Decision Tree (n-gram$_{J48}$) identified correctly (i.e. Recall) 91.1% of the questions and n-gram$_{SVM}$ identified correctly 92.4% of the questions when features such as n-grams and snowball Stemmer were used.

When 2000 questions were evaluated, both GQCC$_{J48}$ and GQCC$_{SVM}$ had nearly similar performance, both classifiers had (100%) recall for class type ABBR and similar recall for classes such as ENTY and HUM. However, GQCC$_{J48}$ has higher precision and f-measure for these classes. In addition, GQCC$_{SVM}$ has better performance for LOC class while for classes such as DESC and NUM GQCC$_{SVM}$ has higher precision and GQCC$_{J48}$ has higher recall and f-measure. Moreover, comparing the performance of n-gram$_{SVM}$ and n-gram$_{J48}$; n-gram$_{SVM}$ has higher performance for class type NUM while n-gram$_{J48}$ has higher performance (100%) precision, recall and f-measure for class type ABBR. For class such as HUM, LOC n-gram$_{SVM}$ has better precision and n-gram$_{J48}$ has better recall and f-measure. In addition, for class type DESC, n-gram$_{SVM}$ has better recall and f-measure while n-gram$_{J48}$ has better precision. while n-gram$_{SVM}$ has higher recall for class type ENTY and n-gram$_{J48}$ has higher precision and f-measure. On the other hand, for classes such as HUM and LOC, n-gram$_{SVM}$ has better precision and n-gram$_{J48}$ has better recall and f-measure. Furthermore, comparing the performance of GQCC$_{SVM}$, GQCC$_{J48}$ and n-gram$_{SVM}$, n-gram$_{J48}$. GQCC$_{SVM}$ has higher precision, recall and f-measure for class type ABBR and HUM. While, n-gram$_{SVM}$ has higher precision, recall and f-measure for class type NUM. In addition, for classes such as DESC, ENTY GQCC$_{SVM}$ has better precision and f-measure while n-gram$_{SVM}$ has better Recall. On the contrary, for class type LOC, GQCC$_{SVM}$ has better recall and f-measure while n-gram$_{SVM}$ has better precision. Furthermore, GQCC$_{J48}$ has better performance for classes such as DESC and HUM while, both classifiers (GQCC$_{J48}$ and n-gram$_{J48}$) have similar precision, recall and f-measure (100%) for class type ABBR. Moreover, for classes such as ENTY and NUM, GQCC$_{J48}$ has better recall and f-measure while n-gram$_{J48}$ has better precision and for class type LOC, n-gram$_{J48}$ has better Recall and GQCC$_{J48}$ has better precision and f-measure.

Table 8: Performance of the classifiers using grammatical features (3000 questions) - Best results are highlighted in bold.

	GQCC$_{SVM}$			**GQCC$_{J48}$**		
Class:	P	R	F	P	R	F
ABBR	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**
DESC	**0.998**	0.998	0.998	**0.998**	**1.000**	**0.999**
ENTY	0.917	**0.920**	0.918	**0.937**	**0.920**	**0.928**
HUM	0.998	**0.998**	0.998	**1.000**	**0.998**	**0.999**
LOC	**0.861**	**0.908**	**0.884**	0.859	0.904	0.881
NUM	**0.987**	0.931	0.958	0.970	**0.948**	**0.959**

Table 9: Performance of the classifiers using n-gram features (3000 questions) - Best results are highlighted in bold.

	n-gram$_{SVM}$			**n-gram$_{J48}$**		
Class:	P	R	F	P	R	F
ABBR	**1.000**	0.909	0.952	**1.000**	**1.000**	**1.000**
DESC	0.961	**1.000**	**0.980**	**0.990**	0.954	0.972
ENTY	0.788	**0.981**	0.874	**0.977**	0.803	**0.881**
HUM	**0.995**	0.866	0.925	0.924	**0.934**	**0.929**
LOC	**0.922**	0.691	0.790	0.700	**0.971**	**0.813**
NUM	**0.991**	**0.942**	**0.966**	0.985	0.916	0.949

5 DISCUSSION

These results indicate that in term of precision, recall and f-measure GQCC$_{J48}$ and GQCC$_{SVM}$ had the better performance when 1000, 2000 and 3000 questions were evaluated. In addition, for class type NUM, which consists of questions such as how many, how much and how long, n-gram$_{SVM}$ performed marginally better than both GQCC classifiers when 1000, 2000 and 3000 questions were evaluated, which indicate that n-gram based classifiers is more suitable in the identification of this type. While CQCC performed better for all other classes (ABBR, DESC, LOC, ENTY and HUM) in which combining grammatical features and domain specific grammatical features improved the classification of these type and enable the machine learning algorithms to better differentiate between different class types, since questions related to these type of classes contain terms related to companies name, geographical areas, places and buildings..etc. *(e.g "What does IBM stand for", "What is the name of the largest water conservancy project in China ?", "Who was Jean Nicolet ?"*

Comparing our approach with the state-of-the-art methods as shown in Table 10, the majority of the previous works used SVM for the classification process; in our experiments it has been shown that other classifiers like J48 could have a better performance and classification accuracy.

The proposed hierarchical classifier in (Li and Roth, 2006) classified questions into fine grained classes, using Sparse Network of Winnows (SNoW); using only syntactical features, the proposed approach achieved accuracy of 92.5% for coarse grained classes. In (Zhang and Lee, 2003) bag-of-words features were used with different machine learning algorithms in which SVM performed better comparing with the other classifiers and has achieved an accuracy of 85.8% with coarse grained classes. Furthermore, In (Huang et al., 2008) head word features were used in addition to wordNet and unigrams; using liner SVM and maximum entropy models the proposed approach has achieved an accuracy accuracy of 89.2% and 89% respectively. In (Metzler and Croft, 2005) the statistical classifier is based on SVM and has achieved an accuracy of 90.2% using coarse grained classes. In (Li et al., 2008) head Noun tagging was used and was combined with syntactical and semantic features; for the classification process conditional random fields (CRFs) and SVM were used; the model achieved an accuracy of 85.6%. In addition, in (Nguyen et al., 2008) the proposed method which is based on question patterns and designed features has achieved an accuracy of 95.2% using SVM. In (Mishra et al., 2013) a combinations of semantic features with the lexico-syntactic features were used which achieved an accuracy of 96.2% for coarse classification. Work in (Van-Tu and Anh-Cuong, 2016) which was based on a new type of features and question patterns ob-

Table 10: Previous approaches performance.

Authors	Features	Algorithms	Acc.
(Li and Roth, 2006)	syntactical features	sparse network of winnows (SNoW)	92.5%
(Zhang and Lee, 2003)	bag-of-words features	SVM	85.8%
(Huang et al., 2008)	head word features, unigrams and word-Net	liner SVM/ maximum entropy	89.2%/ 89%
(Metzler and Croft, 2005)	syntactic and semantic features	SVM	90.2%
(Li et al., 2008)	head noun tagging, syntactical and semantic features	SVM	85.6%
(Nguyen et al., 2008)	question patterns and designed features	SVM	95.2%
(Mishra et al., 2013)	semantic features with the lexico-syntactic features	KNN, NB, SVM	96.2%
(Van-Tu and Anh-Cuong, 2016)	question patterns	SVM	95.2%
(May and Steinberg, 2004)	part-of-speech, parse signatures and WordNet	SVM, MaxEnt, NB, Decision Tree	77.8%
(Xu et al., 2016)	Parts-of-Speech, Bi-Gram and named entities	SVM	93.4%

tained an accuracy of 95.2% for coarse grain using SVM. Moreoever, The hierarchical classifier in (May and Steinberg, 2004) achieved accuracy of 77.8% using different classifiers such as SVM, MaxEnt, NB and Decision Tree. Finally, in (Xu et al., 2016) the proposed SVM-based approach achieved accuracy of 93.4% using a Bi-Gram mode and SVM kernel function.

In conclusion, GQCC had a better results than previous ones due to the ability of our approach to identify different classes of the factoid question using domain-specific information which facilitate the identification of domain categories, unlike previous works which used additional fifty fine-grained categories.

6 CONCLUSION AND FUTURE WORK

A framework has been adapted for question categorization and classification. The framework consists of three main features which are, grammatical features, domain specific features and patterns. These features help in utilizing the structure of the questions. In addition, the performance of different machine learning algorithms (J48 and SVM) were investigated. The results show that our solution led to a good performance in classifying questions compared with the state-of-arts approaches. As future work, we aim to investigate the impact of combing different features like semantic, syntactic and lexical attributes and compare the results. In addition, We are also planning to test other machine learning algorithms to classify the questions.

REFERENCES

Bu, F., Zhu, X., Hao, Y., and Zhu, X. (2010). Function-based question classification for general qa. In *Proceedings of the 2010 conference on empirical methods in natural language processing*, pages 1119–1128. Association for Computational Linguistics.

Bullington, J., Endres, I., and Rahman, M. (2007). Open ended question classification using support vector machines. *MAICS 2007*.

Huang, Z., Thint, M., and Qin, Z. (2008). Question classification using head words and their hypernyms. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 927–936. Association for Computational Linguistics.

Kolomiyets, O. and Moens, M.-F. (2011). A survey on question answering technology from an information retrieval perspective. *Information Sciences*, 181(24):5412–5434.

Le-Hong, P., Phan, X.-H., and Nguyen, T.-D. (2015). Using dependency analysis to improve question classification. In *Knowledge and Systems Engineering*, pages 653–665. Springer.

Li, F., Zhang, X., Yuan, J., and Zhu, X. (2008). Classifying what-type questions by head noun tagging. In *Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1*, pages 481–488. Association for Computational Linguistics.

Li, X., Huang, X.-J., and WU, L.-d. (2005). Question classification using multiple classifiers. In *Proceedings of the 5th Workshop on Asian Language Resources and First Symposium on Asian Language Resources Network*.

Li, X. and Roth, D. (2006). Learning question classifiers: the role of semantic information. *Natural Language Engineering*, 12(03):229–249.

May, R. and Steinberg, A. (2004). Al, building a question classifier for a trec-style question answering system. *AL: The Stanford Natural Language Processing Group, Final Projects*.

Metzler, D. and Croft, W. B. (2005). Analysis of statistical question classification for fact-based questions. *Information Retrieval*, 8(3):481–504.

Mishra, M., Mishra, V. K., and Sharma, H. (2013). Question classification using semantic, syntactic and lexical features. *International Journal of Web & Semantic Technology*, 4(3):39.

Mohasseb, A., Bader-El-Den, M., and Cocea, M. (2018). Question categorization and classification using grammar based approach. *Information Processing & Management*.

Mohasseb, A., El-Sayed, M., and Mahar, K. (2014). Automated identification of web queries using search type patterns. In *WEBIST (2)*, pages 295–304.

Moldovan, D., Paşca, M., Harabagiu, S., and Surdeanu, M. (2003). Performance issues and error analysis in an open-domain question answering system. *ACM Transactions on Information Systems (TOIS)*, 21(2):133–154.

Nguyen, T. T. and Nguyen, L. M. (2007). Improving the accuracy of question classification with machine learning. In *Research, Innovation and Vision for the Future, 2007 IEEE International Conference on*, pages 234–241. IEEE.

Nguyen, T. T., Nguyen, L. M., and Shimazu, A. (2008). Using semi-supervised learning for question classification. In *Information and Media Technologies*, volume 3, pages 112–130. Information and Media Technologies Editorial Board.

Van-Tu, N. and Anh-Cuong, L. (2016). Improving question classification by feature extraction and selection. *Indian Journal of Science and Technology*, 9(17).

Xu, S., Cheng, G., and Kong, F. (2016). Research on question classification for automatic question answering. In *Asian Language Processing (IALP), 2016 International Conference on*, pages 218–221. IEEE.

Zhang, D. and Lee, W. S. (2003). Question classification using support vector machines. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 26–32. ACM.