DOMAIN SPECIFIC SYNTAX BASED APPROACH FOR TEXT CLASSIFICATION IN MACHINE LEARNING CONTEXT

ALAA MOHASSEB[1], MOHAMED BADER-El-DEN[1], HAN LIU[1], MIHAELA COCEA[1]

[1]School of Computing, University of Portsmouth
Buckingham Building, Lion Terrace, Portsmouth PO1 3HE, United Kingdom
E-MAIL: alaa.mohasseb@port.ac.uk, mohamed.bader@port.ac.uk, han.liu@port.ac.uk, mihaela.cocea@port.ac.uk

Abstract:

Due to the vast amount of data, searching and obtaining relevant information on the web is a challenging task. Despite that a broad range of classification techniques have been proposed to improve the information retrieval methods, many difficulties are still present because of the continuous increase in the amount of web contents, as well as its diversity. In this paper, we propose a method that automatically identifies and classifies user queries by using a domain specific syntax approach – this approach is based on the syntactical pattern of each type of search query. A framework is developed to test the performance of the proposed method. Experimental results show that our approach leads to accurate identification of different query types.

Keywords:

Query Classification, Machine Learning, Text Mining, Text Classification, Natural Language Processing

1. Introduction

The search for information through web queries has become more structurally complex over time; thus, two queries with overlapping sets of terms may reflect two totally different intents which makes the identification of user's query intent more challenging.

Despite that semantic search has advanced the information retrieval techniques by looking at different perspectives such as the meaning of words, search engines are still not capable of inferring the meaning of the terms from the query in which they are contained, leading to ambiguity and retrieval of irrelevant information.

Identifying users' intent is one of the major tasks in the enhancement of the query classification process. One major task in identifying the intent of users' query is to establish the query type. Broders taxonomy[1] for establishing the query types is one of the most commonly used ones. It includes three main categories, which are informational queries, navigational queries and transactional queries. Works in [2], [3],[4],[5] and [6] are based on Broder's taxonomy of user query intent.

To distinguish between different types of queries, many previous studies classified queries using different machine learning algorithms, such as Naive Bayes and k-means clustering [6],[7], [8], [9], [5].

Queries submitted to search engines are usually short and ambiguous and most of the queries might have more than one meaning. Therefore, classification based on just the text in the query, or even including users' behavior such as clicks could be misleading [10].

In this paper, we propose a method that automatically identifies and classifies user queries using a domain specific syntax approach based on the syntactical pattern of each type of search query. In particular, we develop a framework to test the performance of the proposed method. Experimental results show that our solution leads to accurate identification of different query types.

The rest of the paper is organized as follows: Section 2 outlines previous work on text and web queries classification. Section 3 provides a detailed description of the proposed approach and evaluation framework. Section 4 reports experimental results. Section 5 concludes the paper and outlines future work directions.

2. Related Work

In this section, we review different machine learning methods that have been popularly used in similar studies for text and query classification. Also, three different types of queries, namely informational, transactional and navigational, are introduced in terms of their characteristics.

2.1. Text and Query Classification

Many different machine learning approaches have been used for classifying natural language sentences and words, and recurrent neural networks (RNN) are one of the approaches that have been used in many studies. The works in [11] and [12], used the recurrent neural network approach for classifying natural language sentences as grammatical or ungrammatical. In addition, a recurrent convolutional neural network approach was introduced in [13] for text classification without human-designed features. It was stated in [14] that most of the previous neural network based methods involved learning based on single-task supervised objectives, which often suffer from insufficient training data. To address this problem, three RNN based architectures were introduced to model a text sequence using multi-task learning of sharing information to model text with task-specific and shared layers, and the entire network was trained jointly on all these tasks [14].

There are also other works such as [15], [16] and [17], which used K-Nearest Neighbour as a method of classification, in addition to feature selection.

Naive Bayes has also been used for automatically classifying text in some studies such as [18], [19] and [20]. According to [21], however, while Naive Bayes is effective in various data mining tasks, it showed disappointing results for automatic text classification. Moreover, they stated that naive Bayes, for the natural language text, has a serious problem in the parameter estimation process, which causes poor results in the text classification domain. Accordingly, they proposed two empirical heuristics: per-document text normalization and feature weighting method. The proposed naive Bayes for text classification performed very well in the standard benchmark collections, comparing with state-of-the-art methods based on a highly complex learning strategy such as SVM.

In addition, some other approaches have been used for classification such as knowledge tree [22], and multilayer SVM-NN based text classification [23].

Many previous works used some of the text classification approaches and techniques such as Naive bayes, K-means and SVM for classifying web queries. Authors in [6] introduced an approach to automatically classifying queries using only the text included in the query. In this work more than 1692 queries were manually labeled, and two machine learning algorithms, naive Bayes and Support Vector Machine (SVM), were then used. Results showed that the two machine-learning algorithms were more suitable for informational and transactional queries. The precision, recall and F-measure rates for classifying navigational queries were very low with naive Bayes and null with SVM. These results indicate that using only the content of words in the queries is not sufficient to find all user intents.

Furthermore, a data-driven methodology was presented in [7] to disambiguate a query by suggesting relevant subcategories within a specific domain in order to find correlations between the users search history and the context of the current search keyword; neural networks and Naive Bayes were used to predict user intent.

In addition, works in [8] and [9] used supervised learning techniques to determine query intents. Moreover, authors in [9] applied unsupervised learning techniques and then combined these techniques with supervised learning techniques to identify user search goals.

Although the majority of the machine learning approaches use classification, or classification with other unsupervised techniques, some works such as [5] classified queries by means of unsupervised learning alone through the use of k-means clustering. The results from this work showed that more than 75% of web queries are informational ones in nature, while navigational and transactional queries are around 12% each.

Finally, another classification has been proposed in [3], where a software application was developed in order to automatically classify queries using a web search engine log of over 1.5 million queries. Results showed that more than 80% of web queries were Informational, Navigational and Transactional queries each represent about 10% of web queries. They used an algorithmic approach to match information from query logs to characteristics of different query types. In this experiment, 74% of the queries were correctly classified and the remaining 25% were vague or multi-faceted queries.

Unlike previous research, our work takes advantage of domain characteristics, thus enriching the query text with contextual information that helps distinguish between different query types.

2.2. Informational, Navigational and Transactional Queries

Web queries are classified according to their intent mainly into three categories based on Broder's taxonomy [1]; the three categories are: Informational, Navigational and Transactional. These categories could be defined as follows:

- *Informational Queries:* the purpose of this type of queries is to find information to either learn how to do something or just answer a question. This information is available on the web in a static form and no further interaction is needed. Topics for these types of queries are usually broad and general; for example, searching for information about

Paris. Others queries are very specific, such as searching for *Aplastic anemia*. In both cases, usually there is no particular web page containing all the information needed; users have to acquire the information from multiple web pages.

- *Navigational Queries:* queries in this category lead to deterministic results since the purpose of such queries is to reach a particular site, such as *British airways homepage*. In this type of queries, the searcher usually has a certain website in mind but either does not know the URL or may think that a particular website exists.

- *Transactional Queries:* the purpose of this type of queries is to find a site and further interaction may be required such as downloading software or online purchasing of a certain product. Also, the purpose may be to acquire something without the need to find information about it – for example to print it out or to look at it on the screen, such as in the case of *lyrics* or *recipes*.

2.3. Characteristics of Web Search Queries

We outline here characteristics of each type of query which are further used in our research to define contextual information.

Informational Search Characteristics: One of the major characteristics of Informational Searching is the use of natural language phrases. Queries for such search consist of informational terms such as *"list"* and *"playlist"*, and question words like *"who"*, *"what"* and *"when"*. Furthermore, it also consist of searches related to advice, help and guidelines such as *"FAQs"* or *"how to"*, in addition to searches related to ideas and suggestions terms and recent information and news such as *"weather"*. Moreover, some queries that involve multimedia like videos are considered informational such as *"how-to-do"* videos, as well as topics related to science, medicine, history, news and celebrities.

Navigational Search Characteristics: Navigational searching contains organization, business, company and universities names, in addition to domain suffixes such as *".com",".org"*, and domain prefixes such as *"www"*, *"http"* and *web* as the source. Furthermore, some Navigational queries contain URLs or parts of URLs.

Transactional Search Characteristics: Queries in Transactional searching are related to obtaining information or products, and include terms such as *"lyrics"*, *"recipes"*, *"patterns"*, and download terms like *"software"*. Also, queries containing audio, video and images are considered transactional. In addition, queries related to entertainment terms such as *"pictures"*, *"games"* and *"e-commerce"*, as well as interact terms such as *"buy"*, *"chat"*, *"book"*, and *"order"*, and file extensions like *"jpeg"* and *"zip"* are considered as typical for transactional queries.

3. Proposed framework

The proposed framework mainly relies on the search types of web queries and the characteristics of each type discussed in sections 2.2 and 2.3 above. Using these characteristics we propose the formulation of syntactical patterns for each query; thus, these give us domain-specific information. Each syntactical pattern is composed of a sequence of term categories. These categories of terms are described below.

The categorization of terms in our solution is mainly based on the seven major word classes in English: Verb (V), Noun (N), Determiner (D), Adjective (Adj), Adverb (Adv), Preposition (P) and Conjunction (Conj). In addition to that, we added a category for question words that contains the six main question words (QW): how, who, when, where, what and which. We further extended this classification by adding two categories, which are Domain Suffixes (DS) and Prefixes (DP). It has to be stated that some word classes like Nouns consists of subclasses, such as Common Nouns (CN), Proper Nouns (PN), Pronouns (Pron) and Numeral Nouns (N), as well as Verbs, such as Action Verbs (AV), linking Verbs (LV) and Auxiliary Verbs (AuxV).

Furthermore, the syntactical patterns of each query search types have been identified by mapping each term in the query to one of the main word classes mentioned above, and then a further mapping is done to assign each term in the query to one of the domain specific term categories. For example, in the query *who is Nelson Mandela*, the terms will be mapped as follows: (a) *"who"* will be mapped to *"QW"*, (b) *"is"* is mapped to *"LV"* and (c) *"Nelson Mandela"* is mapped to *"PN"*. For this step, we used the term categories that were proposed by [24]. Due to space limitation we could not give a comprehensive listing of these categories.

Finally, after each term is mapped to one of the word classes, it will be mapped to the domain specific term category; for example, *"QW"* will be mapped to "Question Word Who" (QW_{Who}); *"LV"* will not be mapped to any further categories and *"PN"* will be mapped to "Proper Noun Celebrity" (PN_C). This step is executed by using a database that contains 10,440 terms [24].

3.1. Domain Specific Syntax Based Approach

To investigate the impact of using the domain specific syntax approach on the classification performance, the following framework, shown in Figure 1, has been developed. The proposed framework involves automatic identification and classification of user's queries based on the patterns described in the previous section. We illustrate the framework by using the following examples of queries: *"Songs by Adele"* and *"Download Songs by Adele"*.

1. Query Parsing and Mapping:

   This step is mainly responsible for extracting users query terms. The system simply takes the query and parses to help map each term in the query to its terms' category.

   Query 1: Songs by Adele

   Terms extracted: Songs, By, Adele

   Query 2: Download Songs by Adele

   Terms extracted: Download, Songs, By, Adele

   After parsing, each term in the query will be mapped to one of the terms category.

   The final mapping will be:

   Query 1 Terms Mapping:

   Songs= CN_{OP}, *by*= P, *Adele*= PN_C

   Query 2 Terms Mapping:

   Download= AV_D, *Songs*= CN_{OP}, *by*= P, *Adele*= PN_C

2. Pattern Formulation

   In this phase after mapping each terms in the query, the pattern is formulated.
   Query 1 Pattern: $CN_{OP} + P + PN_C$
   Query 2 Pattern: $AV_D + CN_{OP} + P + PN_C$

3. Query Classification

   In this step the system attempts to match the query pattern with the most appropriate Search Type Pattern [24] to determine the Query type. For the given examples.

   Query 1: Informational
   Query 2: Transactional

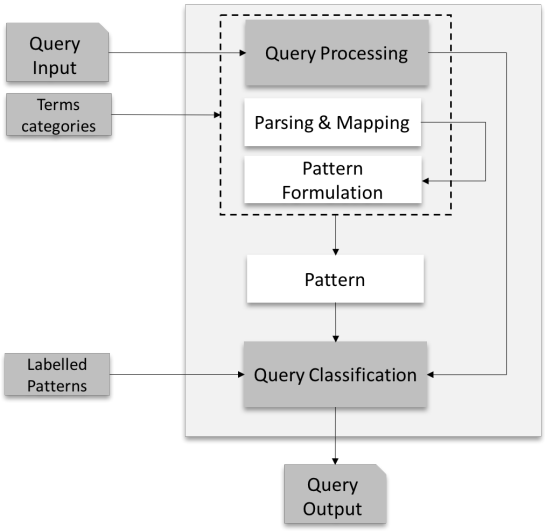

FIGURE 1. Query Classification Framework

4. Experimental Study and Results

In order to validate different machine learning algorithms for the classification of user intents, 10,000 queries were randomly selected from the TRec 2009 data-set[1]. The data set contains seven discrete attributes, each of which represents a word category, such as noun, verb and adjective. The data set involves three classes, namely transactional, navigational and informational, for queries classification.

The *C4.5* and *Naive Bayes* algorithms were used for the automatic classification of the selected queries due to the fact that they among the most popular machine learning algorithms, and have also been popularly used in text classification tasks. Moreover, both C4.5 and Naive Bayes are very capable of dealing with discrete attributes towards learning a classifier. The classification accuracy is obtained by using the implementations of the above algorithms from the Weka software [25], and then the effectiveness of the queries classification is evaluated in terms of Precision, Recall, and F-measure.

The results are displayed in Table 1 for the J48 (the implementation of C4.5 in Weka) algorithm and Table 2 for the Naive Bayes algorithm.

The experimental results show that the decision tree classifier learned by C4.5 identified and classified correctly 99.5% of the queries (i.e. recall), while the classification recall for the classifier learned by using Naive Bayes is 94.5%.

The use of J48 (the implementation of C4.5 in Weka) results

[1] `http://trec.nist.gov/data/million.query09.html`

TABLE 1. J48 Decision Tree performance

Query Search Types	P	R	F
Informational Query	0.999	0.998	0.999
Navigational Query	0.981	1.000	0.990
Transactional Query	0.980	0.986	0.983
Overall	0.987	0.995	0.991

TABLE 2. Naive Bayes Classifier performance

Query Search Types	P	R	F
Informational Query	0.991	0.995	0.993
Navigational Query	0.905	1.000	0.950
Transactional Query	0.943	0.839	0.888
Overall	0.946	0.945	0.944

in incorrect classification of 0.5 % of the queries - these incorrect classifications occur for the informational and transactional queries. Navigational queries were 100% correctly classified.

The use of the Naive Bayes algorithm results in incorrect classification of 5.5% of the queries. The identification of informational queries led to the fewest errors, while the identification of transactional ones led to the higher number of errors. Similarly to the decision tree classifier, the navigational queries were 100% correctly classified.

In previous research, recall, precision and F-measure rates are generally around 80%, e.g. [3], [6], with some results over 90% for informational queries, such as in [6] when using SVM. However, the SVM classifier in [6] could not distinguish navigational queries, which were all misclassified as either informational or transactional (i.e. the value for precision, recall and F-measure was 0 for the navigational class).

These results indicate that the use of Search Type Syntactical Patterns helps with the improvement of the query classification accuracy as well as with the identification of different search types. In addition, our approach shows an improvement in particular in the identification of transactional and navigational queries, which have been shown in previous work to be often misclassified as informational.

5. Conclusion

In this paper, we have proposed a method that automatically identifies and classifies user queries by using a domain specific syntax approach, which is based on the syntactical pattern of each type of search queries. In particular, we developed a framework to test the performance of the proposed method and used machine learning algorithms (decision tree and Naive Bayes) to build models for the identification of user intent. The experimental results indicate that the proposed approach led to better identification of user intent in comparison with previous work, with precision, recall and F-measure values above 94% for Naive Bayes and above 98% for C4.5.

Granular computing [26, 27] has recently become a popular approach for information processing in depth. In future, we will extend the query classification framework introduced in Section 3 in the context of multi-granularity learning towards in-depth processing of text and search queries. In other words, the search queries will be processed in a structural way by providing different levels of granularity, such as sentences, phrases and words. We will also investigate how different learning algorithms can be combined effectively in the context of ensemble learning, towards improving the overall accuracy of queries classification.

Acknowledgements

This paper is related to the first author's PhD research, and is supported by the Computational Intelligence Research Group in the School of Computing at the University of Portsmouth.

References

[1] A. Broder, "A taxonomy of web search," in *ACM Sigir forum*, vol. 36, no. 2. ACM, 2002, pp. 3–10.

[2] Y. Liu, M. Zhang, L. Ru, and S. Ma, "Automatic query type identification based on click through information," in *Asia Information Retrieval Symposium*. Springer, 2006, pp. 593–600.

[3] B. J. Jansen, D. L. Booth, and A. Spink, "Determining the informational, navigational, and transactional intent of web queries," *Information Processing & Management*, vol. 44, no. 3, pp. 1251–1266, 2008.

[4] M. Mendoza and J. Zamora, "Identifying the intent of a user query using support vector machines," in *International Symposium on String Processing and Information Retrieval*. Springer, 2009, pp. 131–142.

[5] A. Kathuria, B. J. Jansen, C. Hafernik, and A. Spink, "Classifying the user intent of web queries using k-means clustering," *Internet Research*, vol. 20, no. 5, pp. 563–581, 2010.

[6] I. Hernández, P. Gupta, P. Rosso, and M. Rocha, "A simple model for classifying web queries by user intent," in *2nd Spanish Conference on Information Retrieval, CERI-2012*, 2012, pp. 235–240.

[7] C. Højgaard, J. Sejr, and Y.-G. Cheong, "Query categorization from web search logs using machine learning algorithms," *International Journal of Database Theory and Application*, vol. 9, no. 9, pp. 139–148, 2016.

[8] S. M. Beitzel, E. C. Jensen, O. Frieder, D. Grossman, D. D. Lewis, A. Chowdhury, and A. Kolcz, "Automatic web query classification using labeled and unlabeled training data," in *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2005, pp. 581–582.

[9] R. Baeza-Yates, L. Calderón-Benavides, and C. González-Caro, "The intention behind web queries," in *International Symposium on String Processing and Information Retrieval*. Springer, 2006, pp. 98–109.

[10] R. Song, Z. Dou, H.-W. Hon, and Y. Yu, "Learning query ambiguity models by using search logs," *Journal of Computer Science and Technology*, vol. 25, no. 4, pp. 728–738, 2010.

[11] S. Lawrence, C. L. Giles, and S. Fong, "Natural language grammatical inference with recurrent neural networks," *IEEE Transactions on Knowledge and Data Engineering*, vol. 12, no. 1, pp. 126–140, 2000.

[12] S. Roa and F. Nino, "Classification of natural language sentences using neural networks." in *FLAIRS Conference*, 2003, pp. 444–449.

[13] S. Lai, L. Xu, K. Liu, and J. Zhao, "Recurrent convolutional neural networks for text classification." in *AAAI*, 2015, pp. 2267–2273.

[14] P. Liu, X. Qiu, and X. Huang, "Recurrent neural network for text classification with multi-task learning," *arXiv preprint arXiv:1605.05101*, 2016.

[15] T. Basu and C. Murthy, "Effective text classification by a supervised feature selection approach," in *2012 IEEE 12th International Conference on Data Mining Workshops*. IEEE, 2012, pp. 918–925.

[16] K. Nithya, P. D. Kalaivaani, and R. Thangarajan, "An enhanced data mining model for text classification," in *2012 International Conference on Computing, Communication and Applications*. IEEE, 2012, pp. 1–4.

[17] G. Wei, X. Gao, and S. Wu, "Study of text classification methods for data sets with huge features," in *Industrial and Information Systems (IIS), 2010 2nd International Conference on*, vol. 1. IEEE, 2010, pp. 433–436.

[18] L. Lv and Y.-S. Liu, "Research of english text classification methods based on semantic meaning," in *2005 International Conference on Information and Communication Technology*. IEEE, 2005, pp. 689–700.

[19] H.-q. Han, D.-H. Zhu, and X.-f. Wang, "Semi-supervised text classification from unlabeled documents using class associated words," in *Computers & Industrial Engineering, 2009. CIE 2009. International Conference on*. IEEE, 2009, pp. 1255–1260.

[20] Z. Gong and T. Yu, "Chinese web text classification system model based on naive bayes," in *E-Product E-Service and E-Entertainment (ICEEE), 2010 International Conference on*. IEEE, 2010, pp. 1–4.

[21] S.-B. Kim, K.-S. Han, H.-C. Rim, and S. H. Myaeng, "Some effective techniques for naive bayes text classification," *IEEE transactions on knowledge and data engineering*, vol. 18, no. 11, pp. 1457–1466, 2006.

[22] L. Peng, Y. Gao, and Y. Yang, "Automatic text classification based on knowledge tree," in *2008 IEEE Conference on Cybernetics and Intelligent Systems*. IEEE, 2008, pp. 681–684.

[23] S. Suganya, C. Gomathi *et al.*, "Syntax and semantics based efficient text classification framework," *International Journal of Computer Applications*, vol. 65, no. 15, 2013.

[24] A. Mohasseb, M. El-Sayed, and K. Mahar, "Automated identification of web queries using search type patterns." in *WEBIST (2)*, 2014, pp. 295–304.

[25] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The weka data mining software: an update," *ACM SIGKDD explorations newsletter*, vol. 11, no. 1, pp. 10–18, 2009.

[26] W. Pedrycz and S.-M. Chen, *Granular Computing and Intelligent Systems: Design with Information Granules of Higher Order and Higher Type*. Heidelberg: Springer, 2011.

[27] H. Liu, A. Gegov, and M. Cocea, "Rule based systems: A granular computing perspective," *Granular Computing*, vol. 1, no. 4, pp. 259–274, 2016.