Advances in Intelligent Systems and Computing 538

Masato Akagi
Thanh-Thuy Nguyen
Duc-Thai Vu
Trung-Nghia Phung
Van-Nam Huynh *Editors*

Advances in Information and Communication Technology

Proceedings of the International Conference, ICTA 2016

Springer

Advances in Intelligent Systems and Computing

Volume 538

Series editor

Janusz Kacprzyk, Polish Academy of Sciences, Warsaw, Poland
e-mail: kacprzyk@ibspan.waw.pl

About this Series

The series "Advances in Intelligent Systems and Computing" contains publications on theory, applications, and design methods of Intelligent Systems and Intelligent Computing. Virtually all disciplines such as engineering, natural sciences, computer and information science, ICT, economics, business, e-commerce, environment, healthcare, life science are covered. The list of topics spans all the areas of modern intelligent systems and computing.

The publications within "Advances in Intelligent Systems and Computing" are primarily textbooks and proceedings of important conferences, symposia and congresses. They cover significant recent developments in the field, both of a foundational and applicable character. An important characteristic feature of the series is the short publication time and world-wide distribution. This permits a rapid and broad dissemination of research results.

More information about this series at http://www.springer.com/series/11156

Masato Akagi · Thanh-Thuy Nguyen
Duc-Thai Vu · Trung-Nghia Phung
Van-Nam Huynh
Editors

Advances in Information and Communication Technology

Proceedings of the International Conference, ICTA 2016

Springer

Editors
Masato Akagi
School of Information Science,
Area of Human Life Design
Japan Advanced Institute of Science and Technology
Nomi-shi, Ishikawa
Japan

Thanh-Thuy Nguyen
Department of Computer Science
VNU University of Engineering and Technology
Hanoi
Vietnam

Duc-Thai Vu
Faculty of Information Technology
Thai Nguyen University of Information and Communication Technology
Thai Nguyen
Vietnam

Trung-Nghia Phung
Thai Nguyen University of Information and Communication Technology
Thai Nguyen
Vietnam

Van-Nam Huynh
School of Knowledge Science,
Area of Knowledge Management
Japan Advanced Institute of Science and Technology
Nomi-shi, Ishikawa
Japan

ISSN 2194-5357 ISSN 2194-5365 (electronic)
Advances in Intelligent Systems and Computing
ISBN 978-3-319-49072-4 ISBN 978-3-319-49073-1 (eBook)
DOI 10.1007/978-3-319-49073-1

Library of Congress Control Number: 2016955568

Printed on acid-free paper

This Springer imprint is published by Springer Nature
The registered company is Springer International Publishing AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

This volume contains the papers presented at the International Conference on Advances in Information and Communication Technology (ICTA2016), which was held in Thai Nguyen city, Vietnam, during 12–13 December 2016. The conference was jointly organized by Thai Nguyen University of Information and Communication Technology (ICTU), Institute of Information Technology—Vietnam Academy of Science (IoIT), Fengchia University, Taiwan (FCU), Japan Advanced Institute of Science and Technology (JAIST) and National Chung Cheng University, Taiwan (CCU). The principal aim of ICTA2016 Conference is to bring together researchers, academics, practitioners and students in order to not only share research results and practical applications but also to foster collaboration in research and education in information and communication technology. The ICTA2016's Program Committee received a total of 150 submissions. Each submission was peer reviewed by at least two members of the Program Committee. Finally, 66 papers were chosen for presentation at Conference Sections and publication in the proceedings. Besides the main track, the conference has three invited speeches. We would like to express our appreciation to all the members of the Program Committee for their support and cooperation in this publication. We would like to thank Prof. Janusz Kacprzyk (Series Editor) and Dr. Thomas Ditzinger (Executive Editor, Interdisciplinary and Applied Science, Engineering, Springer) for their support and cooperation in this publication. We are also thankful to Anand Chozhan and his colleagues at Springer for providing a meticulous service for the timely production of this volume. Last but not the least, we wish to thank all the authors and participants for their contributions and fruitful discussions that made this conference a success.

December 2016 ICTA 2016

Masato Akagi
Thanh-Thuy Nguyen
Duc-Thai Vu
Program Chairs

Organization

Honorary Chairs

Bach Hung Khang	IoIT, Vietnam
Pham Viet Binh	ICTU, Vietnam

General Chairs

Nguyen Van Tao	ICTU, Vietnam
Nguyen Huu Cong	TNU, Vietnam

Program Chairs

Masato Akagi	JAIST, Japan
Nguyen Thanh-Thuy	VNU-UET, Vietnam
Vu Duc-Thai	LICTU, Vietnam

Publication Chairs

Huynh Van-Nam	JAIST, Japan
Phung Trung-Nghia	ICTU, Vietnam
Tran Dang Hung	Hanoi National University of Education, Vietnam
Nguyen Van Toi	ICTU, Vietnam

Organizing Chairs

Dang Thi Oanh	ICTU, Vietnam
Dam Thanh Phuong	ICTU, Vietnam
Le Hung Linh	ICTU, Vietnam
Ngo Manh Tuong	ICTU, Vietnam

Nguyen Hai Minh	ICTU, Vietnam
Nguyen Thanh Ha	TNU, Vietnam
Nguyen Van Nui	TNU, Vietnam
Nguyen Truong Thang	IoIT, Vietnam

Program Committee

Bui Thu Lam, Vietnam
Bui Ngoc Thang, Vietnam
Bui Quang Anh, Vietnam
Ching-Hwa Cheng, Taiwan
Chow Yen Desmond Sim, Taiwan
Chih-Hung Chou, Taiwan
Chyi-Ren Dow, Taiwan
Dang Quang A, Vietnam
Dang The Ngoc, Vietnam
Dang Thi Oanh, Vietnam
Dang Thanh Phu, Vietnam
Dang Van Duc, Vietnam
Dao Huy Du, Vietnam
Dao Phuong Nam, Vietnam
Dao Trung Kien, Vietnam
Do Nang Toan, Vietnam
Do Thanh Nghi, Vietnam
Do Thi Bac, Vietnam
Do Thi Ngoc Diep, Vietnam
Do Thi Tu Anh, Vietnam
Doan Nhat Quang, Vietnam
Doan Van Ban, Vietnam
Don-Gey Liu, Taiwan
Duong Chinh Cuong, Vietnam
Fang-Rong Hsu, Taiwan
Julia Tzu-Ya Weng, Taiwan
Gwo-Jiun Horng, Taiwan
Ha Duyen Trung, Vietnam
Ha Tien Ngoan, Vietnam
Hamido Fujita, Japan
Heng-Yi Su, Taiwan
Ho Dang Phuc, Vietnam
Ho Van Huong, Vietnam
Hoang Do Thanh Tung, Vietnam
Hoang Manh Thang, Vietnam
Huynh Van-Nam, Japan
Huynh Xuan Hiep, Vietnam
Kun-Chih Chen, Taiwan
Lai Hien-Phuong, Vietnam
Lai Khac Lai, Vietnam
Le Ba Dung, Vietnam
Le Duc Hau, Vietnam
Le Dung Muu, Vietnam
Le Hai Chau, Vietnam
Le Hoai Bac, Vietnam
Le Hoang Son, Vietnam
Le Hung Linh, Vietnam
Le Huy Tung, Vietnam
Le Quang Minh, Vietnam
Le Ngoc Tu, Japan
Le Nhat Thang, Vietnam
Le Thanh Ha, Vietnam
Le Thi Lan, Vietnam
Le Thi My Hanh, Vietnam
Le Trong Vinh, Vietnam
Le Trung Thanh, Vietnam
Le Viet Phuong, France
Luong Chi Mai, Vietnam
Luong The Dung, Vietnam
Masato Akagi, Japan
Nghiem Thi Phuong, Vietnam
Ngo Duc Thien, Vietnam
Ngo Hong Son, Vietnam
Ngo Quoc Tao, Vietnam
Ngo Thi Duyen, Vietnam
Ngo Xuan Bach, Vietnam
Nguyen Ai Viet, Vietnam
Nguyen Cat Ho, Vietnam
Nguyen Chien Trinh, Vietnam
Nguyen Cong Dieu, Vietnam
Nguyen Cuong, Vietnam
Nguyen Dai Hai, Vietnam

Nguyen Dinh Thuan, Vietnam
Nguyen Duc Dung, Vietnam
Nguyen Duc Nhan, Vietnam
Nguyen Duc Thuan, Vietnam
Nguyen Duy Minh, Vietnam
Nguyen Gia Nhu, Vietnam
Nguyen Hai Duong, Vietnam
Nguyen Hai Minh, Vietnam
Nguyen Hoai Nam, Vietnam
Nguyen Hoang Anh, Vietnam
Nguyen Hoang Nam, Vietnam
Nguyen Hoang Phuong, Vietnam
Nguyen Hong Quang, Vietnam
Nguyen Huu Cong, Vietnam
Nguyen Huu Quynh, Vietnam
Nguyen Kim Anh, Vietnam
Nguyen Linh Trung, Vietnam
Nguyen Long Giang, Vietnam
Nguyen Minh Tuan, Vietnam
Nguyen Nhi Gia Vinh, Vietnam
Nguyen Ngoc Linh, Vietnam
Nguyen Phuong Huy, Vietnam
Nguyen Quoc Cuong, Vietnam
Nguyen Quoc Khuong, Vietnam
Nguyen Quy Hy, Vietnam
Nguyen Son, Vietnam
Nguyen Tai Hung, Vietnam
Nguyen Thai Ha, Vietnam
Nguyen Thanh Hai, Vietnam
Nguyen Thanh-Thuy, Vietnam
Nguyen The Quang, Vietnam
Nguyen Thi Oanh, Vietnam
Nguyen Thi Phuong Giang, Vietnam
Nguyen Thi Thu Ha, Vietnam
Nguyen Thi Thuy, Vietnam
Nguyen Toan Thang, Vietnam
Nguyen Tran Hiep, Vietnam
Nguyen Truong Thang, Vietnam
Nguyen Van Chi, Vietnam
Nguyen Van Huan, Vietnam
Nguyen Van Huy, Vietnam
Nguyen Van Nui, Vietnam
Nguyen Van Tam, Vietnam
Nguyen Van Tao, Vietnam
Nguyen Van Thuy, Vietnam
Nguyen Van Toi, Vietnam
Nguyen Van Vy, Vietnam
Nguyen Viet Anh, Vietnam
Nguyen Viet Dung, Vietnam
Nguyen Viet Hung, Vietnam
Nguyen Xuan Huy, Vietnam
Nong Thi Hoa, Vietnam
Phan Thuong Cang, Vietnam
Pham Duc Long, Vietnam
Pham Minh Trien, Vietnam
Pham Ngoc Hung, Vietnam
Pham Quang Nhat Minh, Japan
Pham Thanh Giang, Vietnam
Pham Thanh Nam, Taiwan
Pham Viet Binh, Vietnam
Pham Xuan Nghia, Vietnam
Phan Xuan Hieu, Vietnam
Phung Trung-Nghia, Vietnam
Rakshtith Mysore Rajashekar, UK
Ren-Hung Hwang, Taiwan
Ta Tuyen, Vietnam
Tay-Jyi Lin, Taiwan
Teh-Lu Liao, Taiwan
Thai Quang Vinh, Vietnam
Tien-Yin Jimmy Chou, Taiwan
To Van Khanh, Vietnam
Tran Dang Hung, Vietnam
Tran Dinh Que, Vietnam
Tran Duc Tan, Vietnam
Tran Hoai Linh, Vietnam
Tran Hoang Tung, Vietnam
Tran Quy Nam, Vietnam
Tran Thanh Hai, Vietnam
Tran Thi Ngan, Vietnam
Tran Thu Ha, Vietnam
Tran Trung Duy, Vietnam
Tran Van Lang, Vietnam
Tran Xuan Nam, Vietnam
Truong Ha Hai, Vietnam
Truong Kien, USA
Truong Quoc Dinh, Vietnam
Vu Chan Hung, Vietnam
Vu Chien Thang, Vietnam
Vu Duc-Thai, Vietnam
Vu Duc Thi, Vietnam

Vu Duy Hai, Vietnam
Vu Duy Linh, Vietnam
Vu Duy Loi, Vietnam
Vu Hoai Linh, Vietnam
Vu Hoang Giang, Vietnam
Vu Manh Xuan, Vietnam
Vu Nhu Lan, Vietnam
Vu Tat Thang, Vietnam
Vu Tuyet Trinh, Vietnam
Vu Viet Vu, Vietnam
Vu Vinh Quang, Vietnam
Yao-Ting Huang, Taiwan
Yao-Tung Tsou, Taiwan
Yi-Chung Chen, Taiwan

Contents

BKCA, an E-Consultancy System for Studying

Thi Thu Trang Nguyen(✉), Viet Thang Dinh, and Tuan Dung Cao

School of Information and Communication Technology, Hanoi University of Science and Technology, 1 Dai Co Viet, Hai Ba Trung, Hanoi, Vietnam
{trangntt,dungct}@soict.hust.edu.vn

Abstract. In education organizations, a number of quotidian and duplicated consulting questions for training programs, university regulations... make difficulties, inconvenient and time-consuming for consultants. This work proposes a method to suggest answers of similar questions, which may help both students and consultants in orientation. This method includes two main tasks: (i) query generation which extracts key phrases from input questions, (ii) searching which find out most similar questions with their answers and relevant scores using extracted key phrases. For the first task, a check strategy was used to build key phrase candidates and the Naïve Bayes Classifier was used to select key phrases. The two different approaches were experimented for the second one: (i) Similarity Comparison Searching, and (ii) Sorl Search Engine. The precision of the key phrase extraction is about 69 %. BKCA, an e-consultancy system for SoICT-HUST, was built and experimented with about 46 % relevant question-answer pairs for the Similarity Comparison searching approach, and about 39 % for the Sorl Search Engine one.

Keywords: Keyphrase extraction · Recommendation system · Naïve Bayes · E-consultance

1 Introduction

Giving advices for students about studying, regulation... via emails is an important task in universities. This is a convenient and time-saving way for students who need consultancies. With a number of students and hence questions, consultants could not reply all students' questions promptly due to time limitation. They also need to refer to university regulations, old questions (via email) or other information for the answers. Different consultants may receive the same or similar questions at different times. This raises a need of a system, which not only stores and processes question-answer pairs but also could suggest a list of similar questions (and their answers) to the input question from previous question-answer pairs. BKCA, Bach-Khoa Consultancy Application, is such a system, which supports not only for consultants but also for students.

For the problem of giving a list of similar questions for an input question, there are two main tasks: (i) *Query Generation* that extracts key phrases from the input question and (ii) *Searching* that use generated key phrases as a query

M. Akagi et al. (eds.), *Advances in Information and Communication Technology*,
Advances in Intelligent Systems and Computing 538, DOI 10.1007/978-3-319-49073-1_20

to find out relevant questions from historical data (i.e. questions-answer pairs). Key phrases are informative ones that represent the main content of a document.

For the first task, there are a number of researches on key phrase extraction in English either using supervised or unsupervised approach. For example, *Feifan Liu* and his colleagues [4] used unsupervised way to make a query, but his work focused on how to find key words, but not the meaning of them. By using a supervised system, Witten and his colleagues [5] proposed a supervised learning machine for extracting key words with three features of frequency, inverse frequency and distance where key word appears. However, this work did not show the combination of key words, i.e. key phrases. Lee and Croft [7] came up with the idea of using two ways together (e.g. semi-supervised): (i) Find key words based on documents which are chosen by users, and (ii) Choosing candidates by dividing documents into chunks then use a statistical approach and machine learning to define key words. All these works were devoted for English, but not for Vietnamese.

An effort for Vietnamese proposed a method to generate key words by sending terms of a specific-domain ontology to the Google search engine [8]. However, to build ontology is a complex task with much time and expert knowledge. Vietnamese has a complex grammar structure and multi-meaning words. Therefore, it is a big challenge to make key phrase extraction and combine them. A research team of *Phạm Thị Thu Uyên*, which is used in automatic inquiry system, combined snowball method of Agichtein and Gravano [3] and searching machine method of Ravichandran Hovy [6] to extract model of meaning relation in Vietnamese documents. However, finding key words is not enough, sometimes the combination of key words make it change its own meaning. Therefore, the combination of extracted key words plays an important role and improve the information query.

For the second task, there are a number of researches and tools for the query tasks such as indexing, processing, scoring and ranking the result. This problem can be done by using a search engine such as Google Engine, Solr search, Elastic search... However, these systems do not have any support for the Vietnamese language.

In this paper, we propose an unsupervised model combined with machine learning and statistics for the key phrase extraction. There are three tasks in preprocessing Vietnamese documents: (i) Pre-process of the data because the data is raw data, (ii) Key phrase extraction: Candidate phrases are chosen by observation and experiment. (iii) Searching and Ranking to find the question-answer pairs, which are related to the input question. Two searching approaches were experimented: (i) Search Engine Solr, and (ii) Similarity Comparison.

The rest of this paper is organized as follows. Section 2 presents the proposed model for an e-consultancy system for studying – BKCA. In Sects. 3 and 4, the solution for the two main tasks, i.e. Key phrase extraction and Searching, is described. Section 5 is the experiment for the BKCA system. The final section gives conclusions and presents future works.

2 Proposed Model

Figure 1 illustrates our proposed model for suggesting similar questions to the input question. The input question is firstly pre-processed to clean data, normalize acronyms, phrase extraction, word segmentation and POS tagging. Key phrases are then extracted to build query for the next step. List of similar questions with their answers are finally found using the generated query. These three tasks are presented in below subsections.

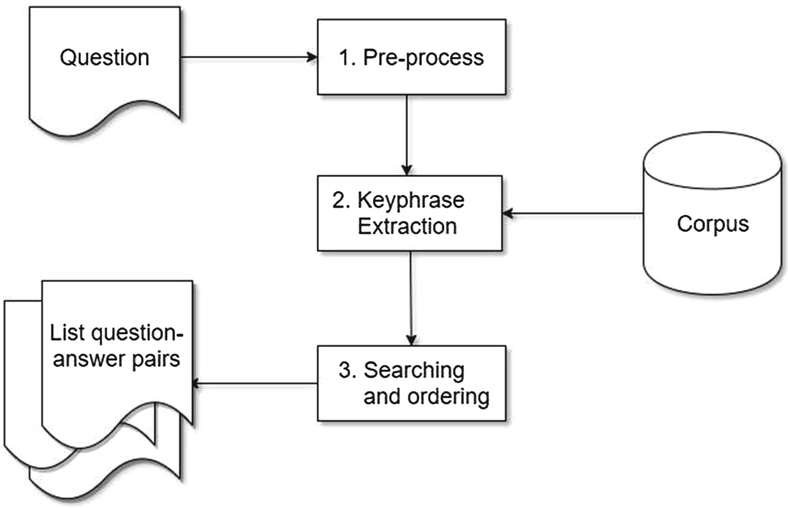

Fig. 1. Proposed model for suggesting similar questions.

2.1 Data Pre-processing

In this task, the raw data from the consultancy emails are collected and pre-processed. This includes the following sub-tasks:

1. *Question-answer extraction*: Dividing the email into question-answer pairs since it might have many question in an email, if in the training phase;
2. *Normalization*: Normalizing the data using an Acronym Dictionary, such as: "chương trình đào tạo" *(curriculum)* for "ctdt" or "CTĐT". Beside an acronym dictionary, we also used some rules for recognizing abbreviations such as all capitalization, all consonants (may be in lowercase), both characters and number...;
3. *Phrase extraction*: Dividing the input question into phases using punctuations;
4. *Word segmentation*: Segmenting words for each phrases;
5. *POS tagging*: Give POS (Part-of-Speech) for all word in phrases

Table 1 gives an example for all pre-processing steps for a question. We adopted the JVnTextPro [2] for the two last steps.

Table 1. An example of pre-processing task

Question		Xin cho em hỏi, theo ctđt, lớp có số lượng đăng ký bao nhiêu thì bị hủy lớp?? *(Could you please let me know, based on the syllabus, a class with how many registrations will be cancelled?)*
Sub-task		**Result**
(i)	Question-answer extraction	(In training phase)
(ii)	Normalization	ctđt: chương trình đào tạo
(iii)	Phrase extraction	Xin cho em hỏi theo chuong trình đào tạo lớp có số lượng đăng ký bao nhiêu thì bị hủy lớp?
(iv)	Word segmentation	Xin cho em hỏi theo chương_trình đào_tạo lớp có số_lượng đăng_ký bao_nhiêu thì bị hủy lớp?
(v)	POS tagging	Xin/V cho/V em/P hỏi/V Theo/V chương_trình/N đào_tạo/N lớp/N có/V số_lượng/N đăng_ký/N bao_nhiêu/X thì/V bị/V hủy/V lớp/N ạ/X?

2.2 Key Phrase Extraction

To automatically predict key phrases from the input question, both supervised and unsupervised approaches were used. Key words were firstly automatically extracted using Naïve Bayes classification and then an unsupervised approach for combining key words into key phrases. In the training corpus for the Naïve Bayes classification model, key phrases were semi-automatically selected by expert opinions. For example, three key phrases were found: "chương trình đào tạo" *(curriculum)*, "số lượng đăng ký" *(number of registration)*, "hủy lớp" *(cancel class)* for the question in Table 1.

2.3 Searching and Ranking

The purpose of this task is to find out the most similar questions to the input question from the historical question-answer pairs. As presented, extracted key phrases are used to search and rank the result. In this step, we experimented two approaches: (i) using Solr Search Engine with a plug-in for Vietnamese language, (ii) using the Similarity Comparison method.

3 Key Phrase Extraction

The model of key phrase extraction includes two stages: Training stage and Extraction stage, which describes in Fig. 2. Key words are firstly identified using POS information. These words are then combined to build key phrase candidates by some unsupervised rules. Key phrases may be a key word or a combination of key words. In the training stage, following the above process, key phrases are chosen for all questions in the training corpus by experts.

Candidate identification and combination. Based on observations on expert opinions, verbs, nouns, adjectives may have important meaning in sentences and hence are considered as candidate key words. Due to their ambiguity, unknown words are also listed as candidate key words. Since Vietnamese has a complex grammar structure and multi-meaning words, key words are not enough. Therefore, adjacent key words are combined to build candidate key phrases, which may increase the accuracy and reliability for Vietnamese. In this case, the two combining rules are used: (i) if there is no adjacent key words to the observing one, this key word is chosen for the next step, and (ii) if the key word c_1 stands next to the key word c_2, a check function is then calculated for their combination c_1c_2 in the corpus, as illustrated in Eq. 1.

$$\text{check} = \frac{\text{count}(c_1c_2)}{\text{count}(\text{Non}c_1c_2) + \text{count}(c_1c_2)} \tag{1}$$

where

- $\text{count}(c_1c_2)$: number of questions which has the combination c_1c_2 (i.e. c_2 stands next to c_1)
- $\text{count}(\text{Non}c_1c_2)$: number of questions which have the two key words c_1 and c_2 but not the combination c_1c_2.

If the value of the check function is equal or greater than 50 % then c_1c_2 is considered as a candidate key phrase. If the value of check is less than 50 % then c_1 and c_2 are added to the key phrase list.

Key phrase selection. To predict and reduce unnecessary key phrases, a supervised method is used with a Naïve Bayes classification. After having some observations and experiments, we finally found five optimal features that can be used to decide the key phrases that can represent a question.

Table 2 shows these five optimal features. The first feature (F_1) is the frequency level of the candidate that appears in the input question. The second feature (F_2) is the percentage level of the question that contains the candidate. The third and the fourth one (F_3, F_4) let us know if the candidate appears in the subject or the in answer or not. The last feature (F_5) represents the distance level of the current candidate to the beginning of the questions, i.e. number of key phrases standing before the current candidate. After observing data, the level of F_1, F_2 and F_5 are proposed in the Fig. 2. They are relative level, which based on their minimum, average and maximum values.

The above five features are used to trained with the Naïve Bayes model due to its simplicity and efficiency. This model is then used to classify if the candidate is a key phrase or not for the input question. The query is finally built based on the list of key phrases.

Table 2. Optimal features for key phrase prediction

Feature	Description	Possible values
$\mathbf{F}_1$	The frequency of the candidate which appears in the input question	Very high, High, Medium, Low, Very low
$\mathbf{F}_2$	The percentage of the question that contains the candidate	Very high, High, Medium, Low, Very low
$\mathbf{F}_3$	Does candidate from the question appear in the subject?	Yes, No
$\mathbf{F}_4$	Does candidate appear in the answer?	Yes, No
$\mathbf{F}_5$	Distance: number of key phrases standing before current candidate	Very high, High, Medium, Low, Very low

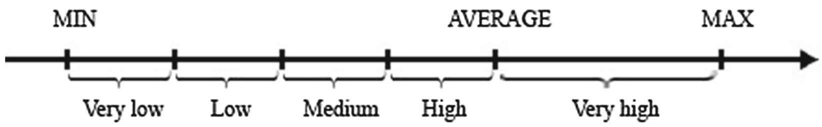

Fig. 2. Levels of feature values.

4 Searching and Ranking

To suggest a list of similar questions and their answers, we experimented 2 approaches: (i) Similarity comparison, and (ii) Sorl Search engine with a Vietnamese plug-in.

4.1 Similarity Comparison

Figure 3 describes the Searching method based on Similarity Comparison. The input is a query, which is a list of key phrases {kp_1, kp_2, ... kp_n} (n<10). The output is top 10 question-answer pairs, which are relevant to the input question. There are three steps in this approach: (i) key phrase comparison, (ii) similarity calculation, and (iii) ranking.

Firstly, for each predicted key phrase in the input question, identifications (ID) of questions having that key phrase are found. The number of times that the question has a key phrase is also recorded. For example, for a query having {đăng_ký, học_tập, cử_nhân_công_nghệ, đồ_án} *(register, study, bachelor of IT, thesis)*, question ids are listed as "1, 4, 5, 6, 8, 9, 10, 17, 39", in detail:

- "đăng_ký" *(register)* appears in question ID = 1, 17, 4, 5, 8
- "học_tập" *(study)* appears in question ID = 1, 4, 5, 9
- "cử_nhân_công_nghệ" *(bachelor of IT)* appears in question ID = 1, 17, 39, 5, 10
- "đồ_án" appears *(thesis)* in question ID = 39, 4, 5, 6.

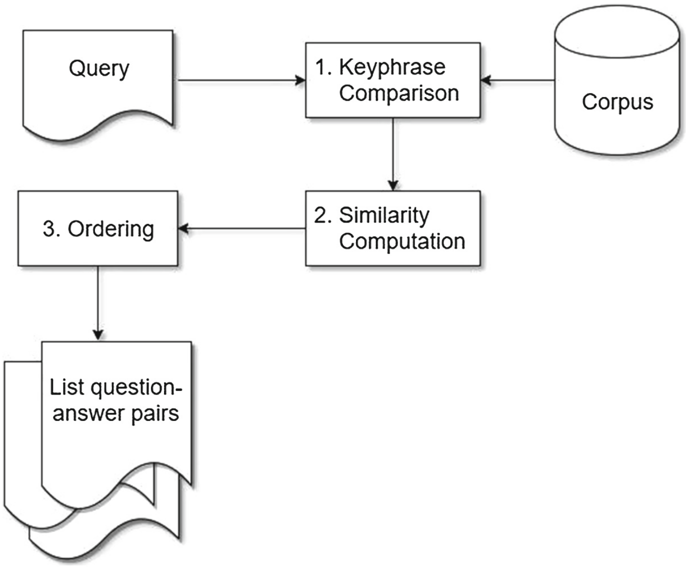

Fig. 3. Similarity comparison model.

The question list is ordered based on the number of times that the question has a key phrase, e.g. "ID = 5" appears 4 times; "ID = 1" appears 3 times; "ID = 4" appears 3 times; "ID = 39" appears 2 times; "ID = 17" appears 2 times. The question having the highest times is considered as the most similar to the input question. If there is more than one question having the same time, the weights of these questions are considered. The weight of a question is decided by the number of true key phrases it has, and then if it is a newer question. For example, the questions "1" and "4" both have 3 times, but the question "1" has more priority since it has a true key phrase "cử_nhân_công_nghệ" *(bachelor of IT)* while the question "4" not. The question "39" is ranked before the question "17" since it is newer, then has more updated information.

4.2 Solr Search Engine with a Vietnamese Plug-in

We experimented another approach to search similar questions based on a query, i.e. using a search engine – Apache Solr search. Apache Solr [1] is an open source search engine that allows users customized and adjusted according to the user's goals. Solr is built Apache Lucene based library support full-text search performance advanced. Currently, Solr search has supported for multiple languages such as English, French, and Japanese... without supporting for the Vietnamese language.

Therefore, to keep the meaning of Vietnamese words, we built a plugin that tokenizes Vietnamese words to the Apache Solr. Some other main tasks to search similar questions based on a query are (i) Removal of Vietnamese stop words, (ii) Synonym Filtering, (iii) Removal of word delimiters, and (iv) Lowercase processing. Word delimiters can be punctuations, or some special rules for tokenizing some special words, such as "BachKhoa" was token to "Bach" "Khoa".

All these words were then all converted to lowercase and finally indexed for searching using the Apache Solr search engine.

All parts of question-answer emails (subject, question, answer) were indexed with different weights, which were chosen based on the observation and experiment (Table 3). The Subject part has the highest priority, while Answer one has the lowest weight.

Table 3. Optimal weight of email parts

Email part	Subject	Question	Answer
Weight	20	10	5

5 Experiment

This section describes how we experimented our proposal for the e-consultancy system, BKCA. This system is being deployed for the School of Information and Communication Technology (SoICT), Hanoi University of Science and Technology (HUST).

5.1 Preparation of Corpus

We investigated on the raw data including more than 1,000 previous emails between students and consultants at SoICT-HUST from April 2015 to April 2016 (about 1 year). We found a number of duplicated questions or answers with out-of-date regulations in the raw data. About one third of emails had threaded conversations between students and consultants, which might give several question-answer pairs per email. Finally, about 620 question-answer pairs were semi-automatically. We divided in to 3 parts: 70 % for training, 20 % for optimization, and 10 % for test.

5.2 Evaluation of Key Phrase Extraction

To evaluate the key phrase extraction step, Precision is calculated as the rate of key phrases that are classified correctly while Recall is the rate of correct predicted key phrases over the actual key phrases. Table 4 shows the result of the key phrase extraction step using Naïve Bayes with five optimal features. The Precision, Recall and F-score are nearly 69 %, which is good for this problem, compared to other languages.

5.3 Evaluation of Searching and Ranking

For each question in 10 % questions in the test corpus, the BKCA system suggested a list of similar questions to the input question. The expert then checked

Table 4. Evaluation of key phrase extraction using Naïve Bayes

Measure	Precision	Recall	F-score
Rate (%)	68.8 %	68.9 %	68.8 %

out all truly relevant questions in that suggesting list. Table 5 shows the experiment results of the two searching approaches. The Similarity Comparison approach gives a better result (about 6 %) and less time-consuming than the Solr search engine one. The Solr search engine also depends on the quality and performance of the Vietnamese tokenizer, while the Similarity Comparison depends on only the extracted key phrases.

Table 5. Comparison between the two searching approaches

Searching approach	Relevant rate
Solr Search engine	38.4 %
Similarity Comparison	46.0 %

6 Conclusion

This study puts an initial step for the problem of suggesting similar questions in Vietnamese with two main tasks: (i) Key phrase extraction to build a query for each question, and (ii) Searching and ranking similar questions based on the generated query. In key phrase extraction, a semi-supervised approach was used: (i) POS information and combination rules were used to build candidate key phrases from key words, and (ii) Naïve Bayes classification with five optimal features was used to choose key phrases from candidates. The five features are: (i) F_1: The frequency level of the candidate in the question (in the corpus) (ii) F_2: The percentage level of the question (in the corpus) having the candidate, (iii) F_3: If the candidate appears in the subject? (iv) F_4: If the candidate appears in the answer, and (v) F_5: the distance level from the candidate to the beginning of the question. Using classified method, Naïve Bayes classification with database of 620 question-answer pairs brought a result with 68.8 % corrected key word extraction. We experimented two approaches to search and rank similar questions: (i) Similarity Comparison, and (ii) Sorl search engine with a Vietnamese plug-in. The Similarity Comparison approach gives a better result (about 6 %) and less time-consuming than the Solr search engine one. About 46 % questions were marked as relevant questions-answers to the input question. In the future, we will study more on how to analyze the meaning of key words and the question/answer. The system is being deployed in the way to get feed back from users to improve data and model.

Acknowledgments. This research is funded by Hanoi University of Science and Technology through the "E-consultancy system for studying in SoICT-HUST" project – grant number T2016-PC-049.

References

1. Apache Solr Search. http://lucene.apache.org/solr/
2. Nguyen, C.T., Phan, X.H., Nguyen, T.-T.: JVnTextPro: a Java-based Vietnamese Text Processing Tool (2010). http://jvntextpro.sourceforge.net/
3. Agichtein, E., Gravano, L.: Snowball: extracting relations from large plain-text collections. In: Proceedings of the Fifth ACM Conference on Digital Libraries, DL 2000, Texas, USA (2000)
4. Liu, F., Pennell, D., Liu, F., Liu, Y.: Unsupervised approaches for automatic keyword extraction using meeting transcripts. In: Proceeding of the Conference of the North American Chapter of the ACL, Boulder, Colorado (2009)
5. Witten, I.H., Paynter, G.W., Frank, E., Gutwin, C., Nevill-Manning, C.G.: KEA: practical automatic key phrase extraction. In: Proceedings of the Fourth ACM Conference on Digital Libraries, California, USA (1999)
6. Ravichandran, D., Hovy, E.: Learning surface text patterns for a question answering system. In: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics (ACL 2002), Philadelphia, Pennsylvania (2002)
7. Lee, C.-J., Bruce Croft, W.: Generating queries from user-selected text. In: Proceedings of 4th Information Interaction in Context Symposium (IIIX 2012), Nijmegen, The Netherlands (2012)
8. Cao, T.-D., Dieng-Kuntz, R., Fiès, B.: An ontology-guided annotation system for technology monitoring. In: IADIS International WWW/Internet 2004 Conference, Madrid, Spain, 6–9 October 2004