Journal of Network and Innovative Computing
ISSN 2160-2174 Volume 2 (2014) pp. 233-239
© MIR Labs, www.mirlabs.net/jnic/index.html

A study of applying Vietnamese voice interaction for a context-aware aviation search engine

Trần Lâm Quân[1], Vũ Tất Thắng[2]

[1] Vietnam Aviation Institute, Vietnam Airlines,
121 Nguyễn Sơn, Hà Nội, Việt Nam
quantl.vai@vietnamairlines.com

[2] Institute of Information Technology, Vietnam Academy of Science and Technology,
18, Hoàng Quốc Việt, Hà Nội, Việt Nam
thang.vu@isolar.com

***Abstract*: Voice searching is the technology underlying in many spoken dialog applications that enable users to access information using spoken queries. With some popular languages like English etc, voice search function is able to help user to search complex voice queries thanks to long time researches for those languages. For Vietnamese language, voice searching has been researched and its results bring some systems that enable user to make simple queries, like number letters. Generally, with a common voice searching look-up system in each language, there are two issues that a smart look-up system has to solve, they are building speech-processing approaches that help user to search complex queries, including some sentences, and researching data-mining approaches to evaluate queries. In this paper, we put efforts to inherit and improve former researches to build a smart look-up systems that help user to search aviation information by voice searching. There are two researches have applied in two steps: firstly, researching statistical models, adaptive algorithms based on Hidden Markov Model (HMM) to build voice interactive functions in Vietnamese, including Speech Recognition and Text To Speech. Secondly, applying data mining techniques to propose a context-aware search engine of Vietnam aviation information. The paper also provides some experimentations of testing the accuracy of this system. Their results show the ability for a Vietnamese voice search system to be applied in practice.**

Keywords: Language model; Acoustic model; Voice Search, Context-aware; Query Suggestion; Data mining**.**

I. Introduction

Nowadays, search engine has become a useful tool for users to search information in the Internet. Over times, traditional techniques based on keyword mapping gradually reveal many drawbacks: the answers of each queries are too large but not lack of valuable information. This fact come from disadvantages of keyword searching, which is not be able to understand the semantic of queries, therefore it can’t identify searching purposes of users, etc... [1], [2]. To overcome these disadvantages, there are many researches in techniques to improve the quality of searching over the last few years, such as semantic web, identifying searching purposes of user by context, classifying results as categories, applying statistical techniques, mining knowledge to interact with the user via queries in the form of natural language, etc... In this research, we present our direction to integrate speech recognition and speech synthesis techniques into a search engine, to create a search engine with interactive voice.

In the field of speech recognition, this is a helpful supporter to users, especially when we search for small devices such as mobile phones or tablets. In recent years, there are many applications applying speech recognition for English language which have been developing and improving in accuracy. However, with Vietnamese language, almost all existent applications which have speech recognition using methods of comparing recording sounds of common Vietnamese words and the sounds recorded by user. This method can only be helpful for recognition of single words and not be able to handle the case of voice searching by complex queries, which include several sentences. To answer complex spoken queries, we have to use advantages of statistic models to statistic sound features and give correct results. Hidden Markov Model (HMM) is a common statistic model which has been widely applied for speech recognition of English. With purposes to allow users to search complex queries, in this research we studied and improved a speech recognition method based on HMM for Vietnamese, along with the support of HMM-based toolkit (HTK).

Along with speech recognition, a good voice search system also has to support users to listen to information of searching results by automatically created synthesis voice. In fact, this feature can help user to manipulate completely with application by speaking and listening. The problem of speech synthesis systems is how to solve basic requirements of creating a pleasant voice. Specifically, the first requirement is that the voice needs to be created with correct pronunciation, tonal and has silences between the points or commas, or in phrases that Vietnamese people often make silences after them. Moreover, the second request is that synthetic voice needs to be not also understandable but also be inspiring voice. Along with advantages in speech recognition, statistic models of HMM can meet quality requirements and general performance. In this study, we have selected the utility of HMM tool inventories through the ability to perform speech synthesis that is understandable and inspired.

To build a smart search engine, we present an approach of data mining based on context-aware suggestion technique for queries. This approach was given by Huanhuan Cao and colleagues [3], [4]. The methodology of this approach starts with the first step: evaluating a set of previous queries before the query that has already put in (the current query) as a context of searching, in order to help the system to capture the intention of users when searching, which brings more accurate recommendations than tradition keyword mapping. Next, in the second step, perform mining method to determine future queries that may appear right after the current query in the search session as the form of a list of suggestions. In other words, this is a specific advantage of this approach compared to former method that only focuses on suggesting similar queries of the current query [3], [4]. A layer of queries that are behind the current query - is formally - reflecting the problems that the community of users often asked after the current query. This layer usually consists of more valuable information of queries and they reflect more correctly the intention of searching. At the same time, this approach helps the look-up system to be able to capture the context of searching, understanding intention of users, therefore it can collapse the answering set to avoid irrelevant results. With aspirations to applying this advantage, in this paper we put research and implement context-aware technique, identifying the advantages and disadvantages of this method, proposing ideas of improvements, as well as making analysis of its effectiveness by an experiment.

This research consists of 5 sections. After introduction in section 1, section 2 presents the principle and application of Hidden Markov models in speech recognition and speech synthesis of Vietnamese language. Section 3 introduces context-aware techniques used in the search engine, analyzing the pros and cons of the proposed technique. Experimental of applying voice searching in aviation search engine is described in section 4. Section 5 is the conclusion and some direction for future researches.

II. Applying HMM based on approach for Vietnamese language voice searching

A. Hidden Markov Models

Markov Chain: In probability theory, Markov process is a random process which has a specific characteristic that each state c_k at the timeline k is in a finite set {1, ... ,M}. In a condition that the process takes place only from timeline 0 to timeline N, the first state and the last state are clear, a chain of Markov state will be represented by a finite vector C = $\{c_0,...,c_N\}$.

Assuming $P(C_k \mid C_0, C_1, \ldots, C_{k-1})$ performs a probability (likelihood) of the state c_k appearing after the process passes through k-1 former state: $c_0, \ldots c_{k-1}$. Assuming that the process depends only on the right previous state and is independent on any other previous states. This process is called a Markov process of order 1 (first-order Markov process). This means the probability of state c_k occuring at the time k, when we know clearly its previous k-1 states depending only on the k-1 status. The process satisfying this condition is called the Markov one-level process. Markov state rank 1 can be represented by a formula:

$$P(C_k\,|C_0, C_1, \ldots, C_{k-1}) = P(C_k|C_{k-1}) \qquad (1)$$

Generally, we can formularize the formula of n-level Markov process.

$$P(C_k|C_0, C_1, \ldots, C_{k-1}) = P(C_k|C_{k-n}, \ldots, C_{k-1}) \qquad (2)$$

With a feature that each state at time k depends only on its right previous state (state at the time k-1), the HMM process is widely used in recognition issues, based on its advantages that do not need to store all states in each process.

Hidden Markov Model: the Hidden Markov model (HMM) is a statistical model in which the system is modeled as a Markov process with hidden parameters, is responsible for determining hidden parameters from the observations. The parameters extracted from this statistical model can be used to perform the successive analyzes, such as pattern recognition applications. The HMM model experiments are described in more detail in [13], [18], [19].

B. HMM based on speech recognition using HMM based Toolkit (HTK)

Nowadays, common architectures for Automatic Speech Recognition - ASR based on forecast, a set of sequences of words from an inserted audio signal. The most popular algorithms are implemented to forecast, which is based on statistical methods [16]. A sample vector y_t of sound features always has length from 10 to 30 milliseconds [15]. There are many ways to choose appropriate features to make vector and the impact of these features on performance of recognition is described in [17]. The process of training vectors of sound features is considered the observations about the phonetic model and is used to calculate $p(\mathbf{y_1^T}|\mathbf{W})$ - the probability of a sequence in the form of vector $\mathbf{y_1^T}$**,** after there was a sequence of word W is pronounced. Given a sequence named, $\mathbf{y_1^T}$, a sequence of W generated by the ASR system is presented by the formula:

$$\widehat{W} = \mathrm{argmax}_W \mathrm{p}(\mathrm{y}_1^T|\mathrm{W})\mathrm{p}(\mathrm{W}) \qquad (3)$$

$\widehat{W}$ is the quantity corresponding to the probability of maximum achievable (Maximum A-posteriori Probability - MAP), $\mathrm{p}(\mathrm{y}_1^T|\mathrm{W})$, which is calculated by the acoustic model (AM), when p(W) is inferred from the Language Model (LM).

HMM's structure: A HMM in the field of recognition issues is defined as a pair of stochastic processes {X, Y}. The process X which is a one-level Markov chain, is not directly observable. The process Y is a sequence of random variables taking values in the space of phonetic parameters, called observation. For each y in Y is an instance of the observed variable and i, j is the state of X representing the state model, this model is shown by the three probabilities:

$A \equiv \{a_{i,j} \mid i,j \in X\}$ Transition distribution

$B \equiv \{b_{i,j} \mid i,j \in X\}$ Observation distribution

$\prod \equiv \{\pi_i \mid i \in X\}$ Initial state

Three quantities are defined as:

$a_{i,j} \equiv p(X_t=j \mid X_{t-1}=i)$;

$b_{i,j}(y) \equiv p(Y_t=y \mid X_{t-1}=i, X_t=j)$;

$\pi_i \equiv p(X_0=i)$;

HMM-based speech recognition using HTK: To apply the method of voice recognition using hidden Markov models, we have inherited and analyzed a HMM-based open source Toolkit (HTK) to implement the Vietnamese speech recognition. According to [20], a process using HTK done through 4 main steps. The first step is data preparation. In this step, we have to prepare audio data for training section and testing the data. With Vietnamese speech recognition, we have to collect phonetic dictionary with phonetic spellings converted from marked to unmarked to fit training in HTK. In this research, we inherited and completed the phonetic dictionary for Vietnamese from the dictionary researched in [14]. Step 2 is creating phonemes (Mono-phoneme HMM). This step will create a good data set of single-Gaussian HMM of phonemes. The first thing we have to do is building a set of phoneme HMMs which have similarities in mean and variance. Step 3 will create the sound of three phonemes (tri-phonemes), which are associated with each state. With HMM phoneme set created in step 2, step 3 will create the sound of three HMMs which depends on context. After preparation of phoneme of HMMs and triphones, these ingredients are used to perform step 4 - evaluating speech recognition. To run speech recognition and evaluation result analysis, we used the analytic tools HResults of HTK.

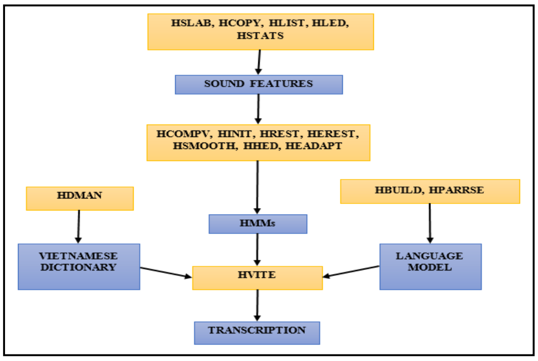


Figure 1. A set of tools to support speech recognition using HTK

Improve the speech recognition system for Vietnamese: In Speech Recognition, a HMM corresponds with phoneme and the HMM's number of state depends on phoneme duration. In English, the optimizing HMM's state is 3 for all phoneme, but for Vietnamese which the phoneme duration is less than English, so the HMM's number of state for Vietnamese phoneme should be less than 3, and for each phonemes. It should differently depends on phonemes duration. In this paper, we propose a method to optimizing HMM's number of state for Vietnamese. Our method was described as follow:

$$n(s) = c + f^*t(s) \qquad (4)$$

In this formula:

- n(s) is the HMM's number of state for phoneme s
- c is the initial number of state
- t(s) is the average time duration for phoneme s
- f is the parameter that need to be optimized.

By this formula, the HMM's number of state will depend on phoneme duration, the longer duration the bigger HMM's number of state, and all phonemes will have at least c state for initialing, to avoid the case that some phonemes have zero state.

C. *HMM-based text to speech using HTK*

In a voice searching system, the speech synthesis function will help search system to notify the results to users in the form of voice interaction. The advantages of HMM model are also the basis for building speech synthesis function of this search engine, along with HTK tool. The input of text to speech system is the set of trained phonemes and independent set of data elements stored the context of the phoneme pronunciation. Speech synthesis is generated by the HTK with 4 main steps. In the first step, HTK trains independent phonemes to produce a set of discrete tri-phonemes group for using in the next step of training data. Step 2 applies the embedded Baum-Welch algorithm for estimating the tri-phonemes. All states in the same position of the HMM tri-phonemes derived from single HMM phonemes are grouped by a layer classified algorithm, aiming to reduce the number of parameters and balance the complex of models, corresponding to the amount of data available. In step 3, tri-phonemes models are continued to estimate using embedded training process. In step 4, the training data is assigned to the model using the Viterbi algorithm, bringing the results of the density and degree obtained from states. Based on this density, we will create a synthetic voice understandable and tonal.

III. Context-aware suggestion applied in an aviation search engine

A. *Overview*

The ideology of context-aware technique is based on two phases: online and offline. In the online phase, the search engine receive the online current query, while it also reviews preceding queries of the current query as a context, more accurately this process is interpreted to the concept sequence - this concept sequence expressed searching intention of users. Catching searching context, system finds matches with available context in the database (offline phase that has already processed and calculated from Query Logs), this context is stored in the suffix tree data structure. Matching process (maximum matching) will extract suggestions queries: This list of queries contains information about problems users often ask after the current query they already entered.

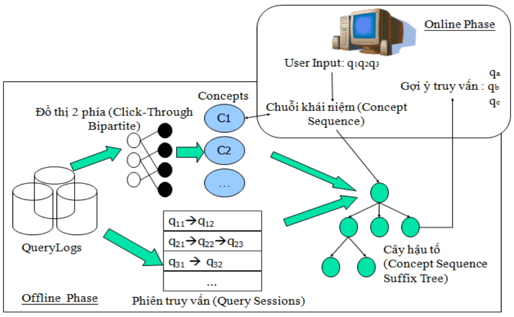

Online Phase
User Input: q1q2q3
Gợi ý truy vấn : qa qb qc
Concepts
Đồ thị 2 phía (Click-Through Bipartite)
C1
C2
…
Chuỗi khái niệm (Concept Sequence)
QueryLogs
q11→q12
q21→q22→q23
q31 → q32
…
Cây hậu tố (Concept Sequence Suffix Tree)
Offline Phase
Phiên truy vấn (Query Sessions)

Figure 2. Model of the context-aware suggestion queries approach [3],[4].

B. *Offline phase*

Offline phase aims to mine the data set (Query Logs - where stores historical information and searching behavior of users, denoted Q) to build the suffix tree. This phase performs sequential steps: the construction of bipartite graph (side of set of vertices queries Q - side of the set of vertices URL Clickeds U), performs clustering from the bipartite graph, construct a set of concept sequences, extracts this set to a set of commonly concept sequences (frequent sequence), to build the suffix tree for future querying. Indeed, performing clustering and building the suffix tree are the most two important steps of the **Clustering on the bipartite graph**: Starting from the basic idea: if two queries share the similar URLs click they are often be similar in content [5], [6], [7].

On the bipartite graph, each query q can be connected to one or more urlClicks entities, as well as every urlClicks entities can share through one or several queries. When conducting clustering on the bipartite graph, each query q_i ($q_i \in Q$) is considered as a vector, each dimension of the vector q_i corresponds with a url u_j on the set U ($u_j \in U$), U is the set of urlClicks. Dimension j of q_i is determined by the formula:

$$\vec{q_i}[j] = \text{norm}(w_{ij}) = \frac{w_{ij}}{\sqrt[2]{\sum_{u_{j'} \in U} w_{ij}^2}} \qquad (5)$$

In this formula:

- w_{ij}: the total number of u_j is clicked by q_i;
- $u_{j'}$: the other urls that are not u_j, is clicked by q_i. (the other dimensions of the vector q_i);

Distance between two queries q_i and q_j is measured by Euclidean distance between two vectors q_i, q_j:

$$\text{Distance}(q_i, q_j) = \sqrt[2]{\sum_{u_k \in U} (\vec{q_i}[k] - \vec{q_j}[k])^2} \qquad (6)$$

Each cluster C contains a set of queries, the center of this cluster is vectorized:

$$\vec{C} = norm\left(\frac{\sum_{q_i \in C} \vec{q_i}}{|C|}\right) \qquad (7)$$

- |C| is number of queries in C;

According to (6), we can calculate distance between query q and cluster C:

$$\text{Distance}(q, C) = \sqrt[2]{\sum_{u_k \in U} (\vec{q}[k] - \vec{C}[k])^2} \qquad (8)$$

The diameter of the cluster is determined by the formula:

$$D = \sqrt[2]{\frac{\sum_{i=1}^{|C|} \sum_{j=1}^{|C|} (\vec{q}[i] - \vec{q}[j])^2}{|C|(|C|-1)}} \qquad (9)$$

The idea of clustering algorithm: This algorithm scans all the queries in Query Logs once, in this time the cluster will be created during the scanning process. Each cluster was originally started by a query, then expanded gradually by similar queries. The expansion of cluster stops when the diameter of cluster is beyond the threshold D_{max}. After clustering, each cluster is considered as a concept. We consider that a set of clusters is a set of concepts.

Constructing suffix tree: As stated in section 3.1, to reflect from Query Logs to set frequent concept sequences, the context-aware approach will implement data mining in each query session. The first step - separating sessions, there are several principles of session separation: 2 input queries are divided into 2 sessions if the time interval between them is more than 30 minutes [8]. Each session after splitting will include a query chain. In the next step, with the set of concept obtained by clustering, the set of query sentences will be mapped into a set of concept sequences. The mapping process can be implemented easily by matching queries.

To mining knowledge from the set of concept sequences obtained, the context-aware approach determines the set of frequent sequences. In the scope of this research, we don’t describe details of algorithms to frequent sequences identifying like GPS[9], PrefixSpan[10]. However, the basic ideas of identifying frequent sequences in the context-aware approach is very clear. For example, with 2 concept sequences c2c3 and c1c2c3 as input, the result of frequent sequence will be c2c3. From a set of frequent sequence, the context-aware approach applies the tree construction algorithm. According to the features of the suffix tree and purposes of the ability suggestions query (use query layers after current query sentence to make suggestions), corresponding to a frequent sequence cs = c1 .. cl, the context-aware approach using cl is the candidate concept for cs' = c1 .. cl-1. Each node in the suffix tree associated with a ranked list of queries (by convention, queries that have more clicking have higher ranks). From suffix tree construction, the context-aware approach moves into query suggestions - the online phase.

C. Online phase

The idea of online phase is showed as following: With a query q entered, assuming that preceding queries (query sequence) of the current query q are q1q2q3, this query sequence will be translated into a concept sequence (by matching the query). Suppose that c9c2c5 is concept sequence obtained by interpretation process. After that, the system matches (maximum matching) c9c2c5 in the concept sequence suffix tree. The matching process is done as follows: first look for c5. If system finds c5 successful, start looking for c2c5 by examining the child nodes of c5. Continuing to spread this look-up from suffix tree to the candidate node, then the ranked list of queries on candidate node (ex, q5q9q20) becomes the suggested list.

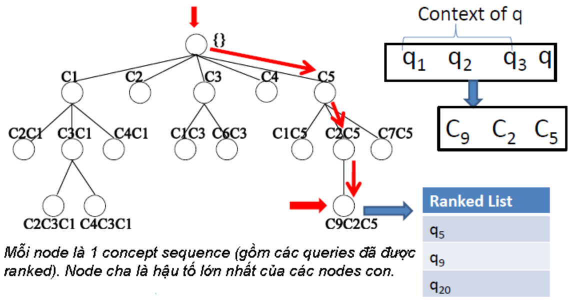


Figure 3. Phase online: the process of query suggestions[3],[4]

D. Advantages and Disadvantages of using context-aware technique

Advantages: Context-aware issue is a new approach. Performing query suggestions, almost all traditional approaches are often taken the classical queries which existed in Query Logs for proposals. This kind of queries can only proposed similar or related queries to the current query, rather than giving trends about which communities often asked after the current query. In addition, there is no public approach which places the previous queries in front of the current query into a search context - as a seamless expression for the intentions of the users. The context-aware technique, above all, is the idea suggested by the problems that users often asked after the current query,which is a unique, efficient, and a “smart focusing” on the field of query suggestions.

Disadvantages: Context-aware approach has some limitations. With a subjective perspective, we summarize several disadvantages of context- aware technique:

- On the bipartite graph, the vertex set of vectors Q is quite sparse (low dimensional), and the vertex set of vectors URLs click also encounters sparse data (URL click sparse), as the result of the sparse vectors, the clustering quality will be affected.

- In fact, during any search session, users might enter a query or many queries. Likewise, users might or might not click many result URLs, of course of which there are many unintended URL clicks (considered as a noise). The fact that the context-aware technique requires a continuous query string to form up the context not reflecting the reality, could be seen as a drawback. In our opinion, however, the dependence on the URL click without consideration of term similarities is regarded as the most obvious drawback of this technique.

IV. Technical proposal

In terms of query suggestions, although having the same philosophy with the team working in context-aware in [3], [4]: "It suggests that the majority of users are often asked after the current query", the approach, implementation, complex formulas, data structures, design, algorithms, source code,. etc. in our search engine are completely different. Mining Query Logs, clustering step in our application does not simply rely on click-through that focuses on three components of fixed and certained, including query; Top N results; set of URLs click. These are the three most important components of data mining tasks, with the premise:

- If the intersection of two keywords (terms) sets in the two queries reaches a certain rate, the two queries are considered similar.
- If the intersection of the top N results of two queries reaches a certain rate, the two queries are considered similar.
- If the intersection of sets of URLs click of two queries reaches a certain rate, the two queries are considered similar.

Context-aware suggestion technique in [3], [4] refers only to the measurement of the similarity on URLs click, not taking into the role of similarity measure according to the set of URLs results and to the set of keywords (terms). Formally, the search intentions are the mapping of users thinking, and that thinking is reflected by language and writing, particularly the writing (terms). Therefore, the considering of the above premise, combined with the threshold derived from empirical measurements to ensure the accuracy of the similarity measure and the fast convergence. In this paper, we list the weight formula set, the similarity measurement of the query (simulation and inheritance from [21]), to propose a new complex formula, for the purpose of calculating the importance and similarity of the query (similarity queries) in order to solve the problem of query suggestion. In fact, the application only executes 02 loops for clustering similar queries. The paper presents the following formulas:

Term Frequency (TF) of term t_i in the query q_j:

$$\mathrm{TF}_{ij} = \frac{n_{ij}}{\sum_k n_{kj}} \qquad (10)$$

In this formula:

- n_{ij}: the number of occurrences of $t_i \in q_j$;
- n_{kj}: total number of terms of q_j;

Simulation of the IDF, the Inverted Query Frequency (IQF) of term t_i in the query q_j being rewritten:

$$\mathrm{IQF}_i = \log \frac{|Q|}{1+|\{q:t_i \in Q\}|} \qquad (11)$$

where: Q: total number of queries in Query Logs, q: $t_i \in$ Q: total number of queries containing t_i.

The weight of term t_i in the query q_j is calculated by TFIQF:

$$\mathrm{TFIQF}_i = w_i = \mathrm{TF}_{ij} \times \mathrm{IQF}_i \qquad (12)$$

Since, weighted q_j equal total weight of the terms of q_j.

The keyword similarity in 2 queries (p, q):

$$Sim_{keywords}(p,q) = \frac{\sum_{i=1}^{n} w(k_i(p)) + w(k_i(q))}{2\times \mathrm{MAX}(\mathrm{kn(p),kn(q)})} \qquad (13)$$

In the above formula:

- kn (p): the total weight of the terms in p, in q;
- w (k_i(p)): the weight of common ith term in p and q;

The similarity in top50 URL results of 2 queries (p, q):

$$Sim_{top50URL}(p,q) = \frac{\wedge(topUp, top\ Uq)}{2\times \mathrm{MAX}(\mathrm{kn(p),kn(q)})} \qquad (14)$$

In the above formula:

- ∧ ($\mathrm{TopU_p}$, $\mathrm{topU_q}$): the intersection of the results top50URL p and q;

The similarity of two queries p, q by Urls_clicked:

$$Sim_{URLsClicked}(p,q) = \frac{\wedge(U\_click\_p, U\_click\_q)}{2\times \mathrm{MAX}(\mathrm{kn(p),kn(q)})} \qquad (15)$$

In the above formula:

- ∧ (U_Click_p, U_Click_q): the common URLs_clicked in p and q;
- U_Click_p: the URL clicked in 1 query;

From (13), (14), (15), we have the equation for similarity combination:

$$Sim_{combination}(p,q) = \alpha.Sim_{keywords}(p,q) + \beta.Sim_{top50URL}(p,q) + \gamma.Sim_{URLsClicked}(p,q) \qquad (16)$$

In this formula, α + β + γ = 1, α, β, γ is the threshold parameter drawn during the experiment. In the search application, α = 0.4, β = 0.4, γ = 0.2.

V. Experimental

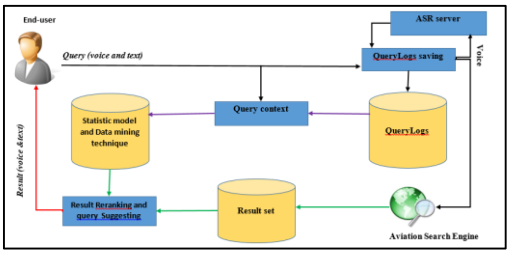


Figure 4. Aviation voice searching architecture.

Speech recognition testing: To experiment the method to improve the accuracy of speech recognition, we use the data for digits number from 0-9, which was spoken by 148 people, 14232 sentences. The data for testing is 1000 sentences. All data were recorded over telephone line. We got the result as follow:.

This statistic shows that the optimizing parameter is c = 1 and f = 10^-9, the result supports the assumption that the HMM's number of state for Vietnamese phoneme should be less than 3. At that point, the accuracy of speech recognition achieves the highest value.

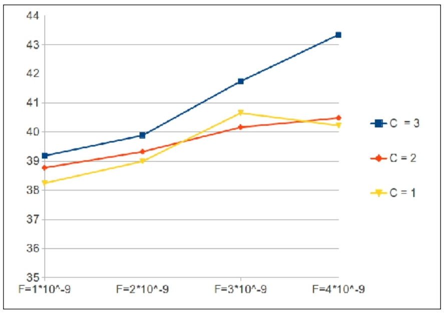
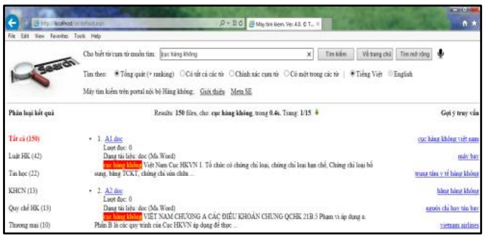


Figure 5. The chart of word error rate of speech recognition relate to number of initial state (C) and f parameter.

Context-aware search engine: Instead of building a general search engine for all fields, we aims to build deeply specialized search engines for corpus in Vietnamese, which allow us to mine the data and understand the searching behavior of users as well [11] [12]. A specialized search engine is different from a general search engine in 3 points: The input data are for specialized fields (aviation herein), in fact many of its articles are hard to be searched on the Internet; The suggest queries using specific techniques (formulas) in typical Query Logs; and The grouping of return results. These 3 main features make the search engine a specific one. In this section, we will conclude two mains evaluations of this system on the accuracy of speech recognition and the accuracy of the answers.


http://localhost/.../default.aspx
Máy tìm kiếm Ver.4.0 © T...
File Edit View Favorites Tools Help
Search
Tìm kiếm
Về trang chủ
Tìm mở rộng

Cho biết từ/cụm từ muốn tìm: cục hàng không

Tìm theo: ● Tổng quát (+ ranking) ○ Có tất cả các từ ○ Chính xác cụm từ ○ Có một trong các từ | ● Tiếng Việt ○ English

Máy tìm kiếm trên portal nội bộ Hàng không. Giới thiệu Meta SE

Phân loại kết quả

Tất cả (150)
Luật HK (42)
Tin học (22)
KHCN (13)
Quy chế HK (13)
Thương mại (10)

Results: 150 files, cho: **cục hàng không**, trong 0.4s. Trang: 1/15

- 1. A1.doc
  Lượt đọc: 0
  Dạng tài liệu: doc (Ms Word)
  cục hàng không Việt Nam Cục HKVN 1. Tổ chức có chứng chỉ loại, chứng chỉ loại hạn chế, Chứng chỉ loại bổ sung, bằng TCKT, chứng chỉ sửa chữa ...
- 2. A2.doc
  Lượt đọc: 0
  Dạng tài liệu: doc (Ms Word)
  cục hàng không VIỆT NAM CHƯƠNG A CÁC ĐIỀU KHOẢN CHUNG QCHK 21B.5 Phạm vi áp dụng a. Phần B là các quy trình của Cục HKVN áp dụng để thực ...

Gợi ý truy vấn

cục hàng không việt nam
máy bay
trung tâm y tế hàng không
hãng hàng không
người chỉ huy tàu bay
vietnam airlines

Figure 6. The search engine applies the knowledge mining techniques.

As illustrated, the left frame is implemented by classification technique based on topics, located on the middle of the screen is the result set returned by aviation search engine, the right frame is done by technique for context-aware for query suggestions.

VI. Conclusion

In this research, we introduced a new approach of interacting with a context-aware search engine for aviation data by Vietnamese voice. To do this task, we point out some features when applying Vietnamese speech recognition, text to speech and algorithm to build a smart search engine. The experiment shows that this approach is feasible to be applied in practice. In future research, we will make further research about these three main issues. The purposes of our future researches is to build a system that has accurate speech recognition, fluent text to speech and semantic searching. Our direction to solve these issues in the future is based on mining data of users and input it to make the system of self-training to be more accurate. For instance, based on the sound recorded from users, the system will learn to make it enable to record all kinds of voice correctly. Besides, although the search engine applies the context-aware method and algorithms on the Vietnamese natural language processing, etc., it is still a search engine working based on keywords. In future studies, we will apply semantic web and techniques for data mining to improve this search engine to be a semantic search engine, which has fully understand meaning of natural questions.

References

[1] Wolfram, D., et al. Vox Populi. “The public searching of the web”, *Journal of the American Society of Information Science and Technology*, 2001.

[2] Jansen, B.J. and Spink, A. “How are we searching the world wide web? a comparison of nine search engine transaction logs”. In *Proceedings of Information processing & management*, 2006.

[3] Cao, H., Jiang, D., Pei, J., He, Q., Liao, Z., Chen, E., and Li, H. “Context-oriented query suggestion by mining click-through and session data”. In *Proceedings of KDD*, pp. 875-883, 2008.

[4] Z hen Liao, Daxin Jiang, Enhong Chen, Jian Pei, Huanhuan Cao, Hang Li. “Mining Concept Sequences from Large-Scale Search Logs for Context-oriented Query Suggestion”, *Journal of ACM Transactions on Intelligent Systems and Technology (TIST) Volume 3 Issue 1, October 2011*. Article No. 17.

[5] Beeferman, D. and Berger, A. “Agglomerative clustering of a search engine query log”. In *Proceedings of the 6th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD’00)*, pp. 407-416, 2000.

[6] Wen, J., Nie, J. and Zhang, H. “Clustering user queries of a search engine”. In *Proceedings of the 10th International Conference on World Wide Web (WWW’01)*, pp. 162-168, 2001.

[7] Baeza-Yates, R.A., Hurtado, C.A. and Mendoza, M. *Query recommendation using query logs in search engines*. Springer, 2004.

[8] White, R.W., Bilenko, M. and Cucerzan, S. “Studying the use of popular destinations to enhance web search interaction”. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR’07)*, pp. 159-166, 2007.

[9] Srikant, R. and Agrawal, R. “Mining sequential patterns: Generalizations and performance improvements”. In *Proceedings of 5th International Conference of Extending Database Technology (EDBT’96)*, pp. 3-17, 1996.

[10] Pei, J., Han, J., Mortazavi-Asl, B., Pinto, H., Chen, Q., Dayal, U. and Hsu, M.-C. “PrefixSpan: Mining Sequential Patterns Efficiently by Prefix-Projected Pattern Growth”. In *Proceedings of the 2001 International Conference on Data Engineering (ICDE’01)*, pp. 215-224, 2001.

[11] Trần Lâm Quân. “Máy tìm kiếm thông minh trên tài liệu Hàng không”, *Journal of Chuyên san ngành Hàng không Việt Nam*, 2009.

[12] Trần Lâm Quân. “Tìm kiếm thế hệ mới: Tìm kiếm thông minh lai”. *Journal of Chuyên san ngành Hàng không Việt Nam*, 2011.

[13] L.R. Rabiner. “A tutorial on hidden Markov models and selected applications in speech recognition”. In *Proceedings of the IEEE, 77(2)*, pp. 257-286, 1989.

[14] Đặng Ngọc Đức, John-Paul Hosom, Lương Chi Mai. “HMM/ANN System for Vietnamese Continous Digit Recognition”. In *Proceedings of IEA/AIE*, pp. 481-486, 2003.

[15] Melvyn. J. Hunt. “Signal Representation”, in *Survey of state of the Human Language Technology*, the Press Syndicate of the University of Cambridge.

[16] Renato De Mori, Fabio Brugnara. “HMM Methods in Speech Recognition”, in *Survey of state of the Human Language Technology*, the Press Syndicate of the University of Cambridge.

[17] R. Haeb-Umbach, D. Geller, and H. Ney. “Improvements in connected digit recognition using linear discriminant analysis and mixture densities”. In *Proceedings of ICASSP*, pp. 239-242, 1993.

[18] Mark Stamp. “A Revealing Introduction to Hidden Markov Models”. San Jose State University. (http://www.cs.sjsu.edu/faculty/stamp/RUA/HMM.pdf)

[19] Hidden Markov Model tutorial. University of Leeds. (http://www.comp.leeds.ac.uk/roger/HiddenMarkovModels/html_dev/main.html)

[20] HTK-documentation. (http://htk.eng.cam.ac.uk/docs/docs.shtml)

[21] Rachna Chaudhary, Nikita Taneja. “A novel approach for Query Recommendation via query logs”, *Journal of Scientific & Engineering Research*, Volume 3, Issue 8, 2012.