VideoQA: Question Answering on News Video

Hui Yang, Shi-Yong Neo, Lekha Chaisorn, Tat-Seng Chua
School of Computing, National University of Singapore
Singapore 117543
{yangh, neoshiyo, lekhacha, chuats}@comp.nus.edu.sg

ABSTRACT

Many users are interested in searching for *information*, while the current video retrieval engines are designed to return only *video sequences*. In this research, we perform question answering (QA) to support personalized news video retrieval. Users interact with our system, VideoQA, using short natural language questions with implicit constraints on contents, duration, and genre of expected videos. VideoQA returns short precise news video fragments as answers. The main contributions of this work are: (a) the extension of question answering technology to support QA in news video; and (b) the use of external knowledge and visual content analysis to help correct speech recognition errors and to perform precise question answering. The system has been demonstrated to be effective.

Keywords

Video question answering, video retrieval, video summarization, transcript error correction

1. INTRODUCTION

Video is the most effective medium for capturing the events in the real world around us. It is also the most dramatic medium as it combines both photo-realistic images and sounds. One of the key technologies required for the efficient management of video data is to support personalized video services, especially on the more structured video sources such as news and sports. Unfortunately, the benefits of such materials are often impeded by the fundamental difficulties with information retrieval: that finding specific information on a video source can be a process which is not only time-consuming and tedious, but also frequently unreliable. This is because video is digitized and stored as a continuous stream. The stream may be up to one hour long for news video. When a user asks the system for information relevant to a query, it is insufficient to simply point the user to the entire hour of video. One would expect the system to return a reasonably short segment, preferably only as long as necessary to provide the requested information.

There are many simple factoid questions like: "*What is the score of the football match last night?*" or "*What are the symptoms of atypical pneumonia?*" posed over news video collection where the users expect to acquire short video segments containing the precise answers. To unearth the concise and informative answers from a given video requires good understanding of the video semantic content. The semantic contents of video come from multiple modal features like visual, audio, and most importantly, speech and text. Most of these features may contain errors. It is thus necessary to fuse these multiple sources of often imprecise information to discover the story units in video, identify the appropriate portions of stories that answer the queries, and to generate video summary. In fact, to correctly answer "*What is the score of the football match last night?*" requires the analysis of *key terms* ("score, football, match"), *video genre* (sports), and implicit constraint like *duration* (<= 30 seconds). The realization of such system requires the fusion of technologies in question answering (QA), speech recognition, and MM content analysis.

The development of a video-based QA system requires the solution to three fundamental problems in video and text processing. The first is to segment the video sequence into story units with correct genre classification. Several works have been done on this, including (Chaisorn et al 2002) and (Hsu & Chang 2003). These approaches perform multi-modal analysis using a combination of visual, audio and textual features based on HMM or entropy techniques and reported accuracies of about 90% in story segmentation and genre classification. The analysis should also generate summary to provide concise answers.

The second problem is that the users' questions are normally short and assume previous context. For example, for the aforementioned "*football match"* question, one need to also know also that the main match of interests to user involves, say, "*Manchester United and Real Madrid*". There are several ways to extract the context to a query. We can induce the context from the query logs, or the recent related news articles available on the news web sites. The analysis of domain and external knowledge will provide the context to help in understanding the precise meaning of the short queries, alleviate speech recognition errors in text transcripts video, retrieval relevant sentences in the transcripts, and ensure that they are of the right genre.

In order to extract relevant sentences in the video's speech track that answer the query precisely, we need the ability to analyze the text transcript at sentence level. Thus a third problem is to overcome the recognition errors in the speech accompanying the video. Text from speech (or transcript) is a major source of semantic information for news video. The conventional video retrieval based on transcript suffers a lot from the numerous speech recognition errors. This is especially severe for substitution errors that cause many names of person, location and organization to be wrongly recognized. As names are essential to induce the semantics of news, there is a need to identify and correct such errors using the names that we already know from related news articles.

Multimedia '03, November 2-8, 2003, Berkeley, CA, USA.

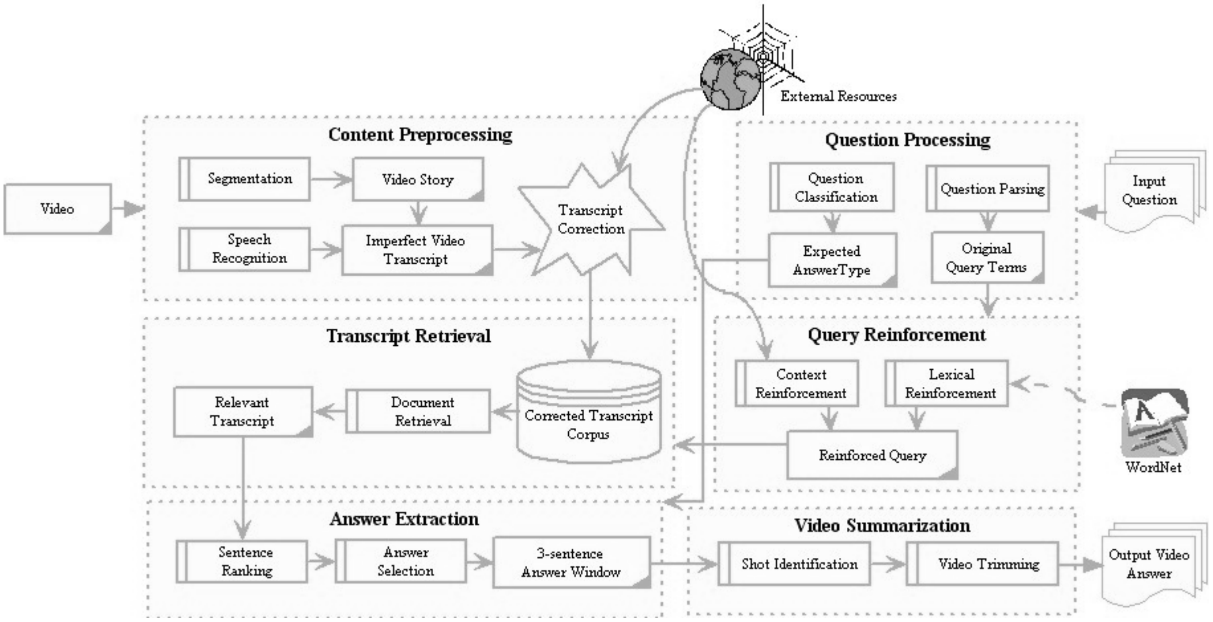


Figure 1: System architecture of VideoQA

In this paper, we discuss the design of a news video question answering system called VideoQA. Users interact with VideoQA using short natural language questions with implicit constraints on contents, duration, and genre of expected videos. The system returns the relevant news video fragments as the answers, supplemented by text version of latest news, summarized to the duration constraint as specified by the users. The paper discusses our research in tackling the above three problems, and the summarization of video into a single or multiple-sentence unit. The main contributions of this work are: (a) the extension of question answering technology to support QA in news video; (b) the use of external knowledge and visual content analysis to help correct speech recognition errors and to perform precise question answering.

The rest of the paper is organized as follows. Section 2 outlines our system architecture. Section 3 discusses our approach to correct news transcript errors by utilizing external web resource. Section 4 details the application of QA technology to retrieve precise answers in video database. Section 5 presents the experimental results. Section 6 outlines related work and Section 7 concludes the paper.

2. SYSTEM ARCHITECTURE OF VideoQA

Our system, named VideoQA, aims to provide precise video answers to simple factoid questions posed over the news video collection. It handles natural language questions by unearthing the answers embedded in the video collection and presenting the video segments in the form of video summary. It is naturally used in a personalized video setting in which a user may request for details of certain aspects of news or summary of latest news. It will be an essential component of future information systems.

During the preprocessing stage, VideoQA performs video story segmentation and classification, as well as video transcript generation and correction. During question answering, VideoQA employs modules for: question processing, query reinforcement, transcript retrieval, answer extraction and video summarization. Figure 1 gives the system architecture of VideoQA.

Given the news video collection, the pre-processing stage “prepares” the video for later answer retrieval. We analyze the raw video using a two-level story segmentation scheme as proposed in Chaisorn et al (2002). The basic unit of analysis is the shots, and we employ multi-modal analysis involving visual, audio and textual features. Briefly, we model each shot using high-level object-based features (face, video text, and shot type), temporal features (background scene change, speaker change, motion, audio type, and shot duration), and low-level visual feature (color histogram). At the shot level, we employ the Decision Tree to classify the shots into one of 13 genre types of: *Intro/ Highlight, Anchor-person, 2-anchor-person, Meeting/ Gathering, Speech/Interview, Live-reporting, Still-image, Sports, Text-scene, Special, Finance, Weather, and Commercials*. We then perform HMM analysis to detect story boundaries using the shot genre information, as well as time-dependent features such as the speaker change, scene change and key phrases. The resulting video story may contain shots of different genre types. For example, a general news story typically contains shots of type *Anchor-person, Live-reporting* and *Speech/Interview*; while a sports story includes shots of type *Sports* and *Text-scene*. The scheme shown in Figure 2 was demonstrated to be effective for sparse data and we could achieve an F_1 measure of about 90%.

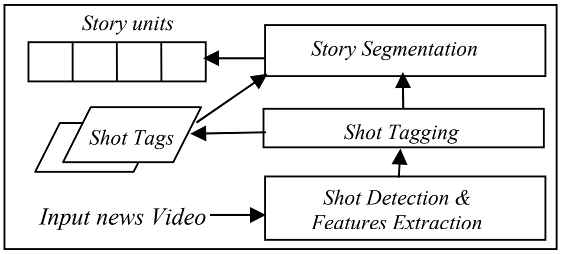


Figure 2: Overview of video story segmentation system

The pre-processing stage also generates the text transcripts of video by performing the speech recognition on the audio track. However, the transcripts contain numerous speech recognition errors, which cause many words, especially *names,* to be wrongly recognized as other similar sounding words. To correct such errors, we retrieve the news articles from the news web sites, and extract a list of possible names and their correlations. We then use a phonetic-based matching technique to match the likely names in the transcript to those found on the name list, and correct the speech recognition errors as appropriate.

In the question answering stage, we first perform question analysis to extract the key terms in the question; the type of questions and its likely answer targets; the type of video genre, and the implicit duration constraint. We next go to the news web sites to retrieve the latest related news and derive the words related to the queries. These terms, especially the named entities, provide the context to the queries. We expand the original query using these extracted terms to form a new query, ie. $q^{(0)} \rightarrow q^{(1)}$.

Next, we match the new query $q^{(1)}$ against the (corrected) transcripts at story level. We perform QA analysis to retrieve the relevant transcripts down to the sentence level. Given the set of retrieved sentence(s), we ensure that the genre of the associated video is, or the sub-class, of the expected type. For example, for speech query, we expect the video to be of type *Speech* with a detected face. The analysis helps to remove those retrieved passages with video genre of the unlikely types. For example, "*Saddam Hussein*" cannot be associated with sports video. Finally, we generate a single or multi-sentence video summary. The following sections described the details of our approaches

3. PROCESSING OF IMPERFECT NEWS TRANSCRIPT

One major source of the semantic information for news video is the text from speech (news transcript). In this work, we use the Sphinx-III speech recognition engine (Seymore et al 1997) to generate the synchronized news transcript segmented at sentence level (separated by speech pause). Because of the complexity of human vocal track, and the differences across different speakers, dialects, transmission distortions, and speaking environments (Lee 1998), many errors incurred during speech recognition. One main type of error is the substitution error, where one or more wrong similar sounding words are "substituted" in place of the correct word. This type of error has caused many name-entities such as the names of person, place, organization or object to be wrongly recognized. Examples of such errors, as listed in Figure 4, include: *pneumonia* → *new area*; and *Jose Maria Aznar* → *Jose Mari ask not*. For ease in discussion, we use the term *Answer Target (ATs*) to collectively denote the noun phrases and name entities, which include also dates and numbers etc.

As ATs are essential to understanding the semantics of video, they must be corrected to alleviate the effects of speech recognition errors. This is especially so for QA where greater matching accuracy at sentence level is needed. As substitution errors arise from the "substitution" of wrong but similar sounding words, one obvious approach is to convert the words to phonetic sounds and match the phonetic sequence of words at syllable level. A similar approach has been used to expand the spoken language queries for effective information retrieval by Singhal & Pereira (1999) and Shen et al (1998). As there may be a large number of similar sounding words, a straight-forward application of phonetic matching may result in low accuracy (Ng 2000). Thus we need to constrain the list of terms to be matched in order to ensure accuracy. Fortunately, in the closed domain like news, we are able to extract a list of possible ATs from recent news articles available on the news web sites. In addition, we use OCR output of video text in video news stories to help correct speech recognition errors. This section discusses our approach to correct most speech recognition errors in ATs to support QA.

3.1 Associating Answer Targets to Video Transcripts

We utilize the external resource by retrieving a list of recent news articles from the news web sites. Our aim here is to associate the list of ATs (*answer targets*) obtained from the news articles with each of the imperfect news transcript. The list then provides the basis for phonetic sound matching of words in the transcripts at the syllable level.

For each retrieved news article D_j, we extract the list of ATs, which we denote as $a_{jk} \in \underline{A}_j$. We employ the shallow parsing tools from UIUC (Roth 2003) to extract the noun phrases and the tool developed by Chua & Liu (2002) to extract the name entities. From the collection of recent news articles, we can harvest the list of all recent ATs as $\underline{A}_{all}$. Similarly we extract the list of correct ATs found in each news transcript T_i, as $t_{ik} \in \underline{\Gamma}_i$. As news transcript contains numerous errors and is likely to be ungrammatical, we use direct string matching to find $\underline{t}_i$ from $\underline{A}_{all}$.

The problem then becomes: given the set of known ATs from the recent set of news articles, $\underline{A}_{all} = (a_1, a_2, .., a_n)$, we want to estimate:

$$p(\underline{A}_i \mid T_i) \qquad (1)$$

the probability that given the transcript T_i, the set of $\underline{A}_i$ in $\underline{A}_{all}$ that will appear in T_i. We can estimate the list $\underline{A}_i$ using two approximations.

First, we estimate $\underline{A}_i$ from the set of news articles that is highly similar to transcript T_i. Here we use only ATs as the basis for matching the similarity as ATs convey more precise semantics of news stories. Thus for each news transcript T_i, we compute its cosine similarity (Salton & McGill 1983) with article D_j as:

$$Sim(T_i, D_j) = \frac{\sum_k (t_{ik} * a_{jk})}{\| \underline{\Gamma}_i \| * \| \underline{A}_j \|} \qquad (2)$$

We select the top m news articles (m = 2) with $Sim(T_i,D_j)>\sigma_1$. We extract the ATs from these top m articles as the likely ATs to appear in the transcript. We denote the list as $\underline{A}_{i1}$.

Second, we extract the list of ATs in $\underline{A}_{all}$ that have high co-occurrence probabilities within the same set of news articles as each t_{ik} found in transcript T_i. For each $t_{ik} \in \underline{\Gamma}_i$, we compute its correlation with $a_{jk} \in \underline{A}_j$ as:

$$Corr(t_{ik}, a_{jk}) = \frac{d_s(t_{ik} \wedge a_{jk})}{d_s(t_{ik} \vee a_{jk})} \qquad (3)$$

where $d_s(t_{ik} \wedge a_{jk})$ gives the number of news articles that contains both t_{ik} and a_{jk}; and $d_s(t_{ik} \vee a_{jk})$ gives the number that contains either t_{ik} or a_{jk}. We extract the top p ATs with $Corr(t_{ik},a_{jk})>\sigma_2$ and denote the list as $\underline{A}_{i2}$.

Finally, we merge the two lists to obtain the list of likely ATs to appear in transcript T_i, as: $\underline{A}_i = \underline{A}_{i1} \cup \underline{A}_{i2}$.

3.2 Correcting Transcript Errors

Given the restricted set of probable ATs, $\underline{A}_i$, we look for possible occurrences of some of these ATs in transcript T_i by using the less exact phonetic matching technique. For the Sphinx-III speech recognition engine (Seymore et al 1997) that we use, we noticed that most substitution errors involves the "substitution" of an AT by one or multiple simpler similar sounding words. Therefore, our system concentrates on matching the set of ATs in $\underline{A}_i$ to single or multiple words in the transcripts.

One example of substitution error is the AT "*pneumonia*" that was wrongly recognized as two simpler words or a phrase, "*new area*". Their respective phonetic strings as defined in the phonetic dictionary of Sphinx system are: *<N AH M OW N Y AH>* and *<N Y UW>*, *<EH R IY AH>*. It is clear that at the phonetic level, both set of strings are highly similar. By observations and through experimentations, we found that the similarity between the phonetic representations of two strings can be established on the basis of: (a) the similarities of their first and last syllables; and (b) the number of correct syllable matches in the occurrence sequence. In addition, the two phonetic strings should have approximately the same length.

Hence, given two phonetic strings x and y of approximately the same length, we derive two measures to compute their similarity as follows (see also Figure 3):

a) String Boundary Similarity, which measures the similarity in the starting (start())and ending (end()) phonetic sounds.

$$S_b(\text{x, y}) = \begin{cases} 1, & \text{if start(x)=start(y) and end(x)=end(y);} \\ 0.5, & \text{if start(x)=start(y) xor end(x)=end(y);} \\ 0, & \text{otherwise.} \end{cases} \quad (4)$$

b) Longest Common Sub-sequence (LCS) Similarity. This computes the phonetic matches between two strings in their occurrence order using the LCS algorithm (Cormen et al 1990), and normalizes the measure by the longer of string x or y.

$$S_l(x,y) = LCS(x,y)/\max\{|x|,|y|\} \quad (5)$$

For the above "*pneumonia*" example, the similarities values are: $S_b(x,y)=1$ and $S_l(x,y)=2/7$; and the overall similarity between "*pneumonia*" and the phrase "*new area*" is:

$$S_p(x,\, y) = \alpha_b\, S_b(x,y) + \alpha_l\, S_l(x,y) \quad (6)$$

where $\alpha_b+\alpha_l=1$ are weights where we set $\alpha_b=\alpha_l=0.5$.

In many cases, the AT may contain multiple words, such as the AT "*tony blair*". In this case, we compute the $S_p^k(x,y)$ for each word *k* in AT with a word or phrase of equivalent length in the transcript, and compute the overall similarity as:

$$S_p(x,y) = \frac{(\gamma)^{m-1}}{m} \sum_{k}^{m} S_p^k(x_k, y_k) \quad (7)$$

where m is the number of words in the AT to be matched, and $\gamma>1$ is a constant to give higher weight to multiple word matches.

The process of correcting the errors in transcript T_i is given below.

a) Convert all ATs in $\underline{A}_i$ to phonetic strings by looking up the Sphinx phonetic dictionary.

b) Similarly convert all words in transcript T_i to phonetic strings.

c) Exploit phonetic string similarity to perform phrase spotting. Compute the phonetic similarity between each AT in $\underline{A_i}$ and each possible word sequence of equivalent length in the transcript with phonetic string *y* using Equation (7).

d) Select the highest $S_p(x,\, y)$ above the threshold τ_s as the desired AT and insert that AT at position y in the transcript.

In the above procedure, an AT_k is considered to be present in transcript T_i iff: $AT_k \in \underline{A}_i$ && $\arg\max_k S_p(AT_k, y)$, for some string y of equivalent length in transcript T_i.

Example 1:
Original NP: "pneumonia" <N AH M OW N Y AH>
Recognized string: "new area" <N Y UW> <EH R IY AH>
$S_b(x, y)=1$; $S_l(x, y)=2/7$;
Overall similarity is: *S_p(pneumonia)*

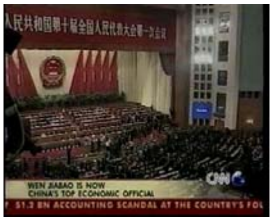
Example 2:
Original NP: "tony blair" <T OW N IY> <B L EH R>
Recognized string: "teddy bear" <T EH D IY> <B EH R>
For tony: $S_b(x, y)=1$; $S_l(x, y)=2/4$;
For blair: $S_b(x, y)=1$; $S_l(x, y)=3/4$;
Hence: *S_p(tony, blair) = $[(S_p(\text{tony}) + S_p(\text{blair})] * \gamma /2$*

Example 3:
Original NP: "Jose Maria Aznar"
<HH OW Z EY> <M ER IY AH> <?>
Recognized string: "Jose Mari ask not"
<HH OW Z EY> <M AA R IY> <AE S K> <N AA T>
For jose: $S_b(x, y)=1$; $S_l(x, y)=1$;
For maria: $S_b(x, y)=1/2$; $S_l(x, y)=2/4$;
For aznar: $S_b(x, y)=0$; $S_l(x, y)=0$;
Hence: S_p(jose, maria, aznar) = $[(S_p(\text{jose}) + S_p(\text{maria})] * (\gamma)^2/3$

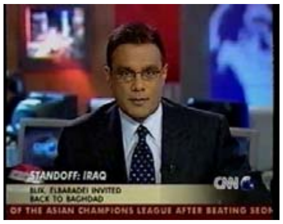
Figure 3: Examples of NPs and the wrongly recognized strings

3.3 Correcting Non-English Name Errors using Video Text

We noticed that many non-English names (especially those of Asian origins) are wrongly recognized. Because of the limitation of the version of Sphinx-III system that we are using, we cannot find the phonetic representations for many of these names. Thus we cannot correct the errors in these names using the phonetic matching technique as described above. Fortunately, many such names also appear in video text during the news story. For example, the name "*Wen Jiabou*" is not in the phonetic dictionary of Sphinx-III. But it appears as video text: "*WEN JIABOU IS NOW CHINA TOP ECONOMIC OFFICIAL*" (see Figure 4a). Similarly, "*Blix*" is not in Sphinx-III dictionary but appears in the video text of the news story as "*BLIX, ELBARADEI INVITED BACK TO BAGHDAD*" (see Figure 4b).


人民共和国第十届全国人民代表大会第一次会议
CNN
WEN JIABAO IS NOW
CHINA'S TOP ECONOMIC OFFICIAL
$1.3 BN ACCOUNTING SCANDAL AT THE COUNTRY'S FO

(a)



(b)

Figure 4: Video text appearing in video news stories

Given the video-text output (with about 25% character recognition errors), we adopt a greedy approach to approximately match the presence of ATs$\in\underline{A}_i$ in the video text, and in transcript T_i as:

a) We extract OCR output of video-text in video story i as $\underline{T}_{vtext}$. We look for possible ATs$\in\underline{A}_i$ in $\underline{T}_{vtext}$ by performing character level matching using the LCS algorithm. This is to cater to possible OCR errors in video text recognition. We denote the ATs found as $\underline{A}_{vtext}$.

b) If part of $\underline{A}_{vtext}$ has a phonetic equivalent in the phonetic dictionary (such as "*Wen*" in the name "*Wen Jiabou*"), we perform phonetic level matching using the procedure outlined in Section 3.2. Otherwise, we perform character level matching using the LCS algorithm on string of equivalent length in T_i.

c) If $\underline{A}_{vtext}$ is found with sufficiently high confidence, we append it at position y in T_i. Otherwise, we append $\underline{A}_{vtext}$ at the end of transcript T_i.

The above procedure is "greedy" as it tries to append $\underline{A}_{vtext}$ into the appropriate position in T_i as much as possible. This is to maximize the chances of retrieving the transcripts and sentences, as we have other measures during QA analysis to remove wrong transcripts.

4. VIDEO QUESTION ANSWERING

Given the segmented news video stories, each with the corrected news transcript, the next task is to perform QA during retrieval to select precise answers at the sentence level. We assume that a user wants to view a video summary in less than 30 seconds. Hence, by default, we return about 3 sentences for each query. For each query, we perform a series of analysis including query processing, query reinforcement, transcript retrieval and sentence extraction, and video answer processing. Our approach is adopted from the one described in Yang & Chua (2003). The following sub-sections describe the details of each step.

4.1 Query Processing

Users may issue short questions like "*What is the score of the football match last night?*" or "*What are the symptoms of atypical pneumonia?*" For each question, we need to infer the intents of the users, both in terms of precise information needs, and the type of expected answer targets and video genre type. In general, the question $Q^{(0)}$ can be modeled as:

$$Q^{(0)} = Content + Constraint \qquad (8)$$

Content :=query words $\underline{q}^{(0)}$*; noun phrases* $\underline{n}$*; named entities* $\underline{h}$

Constraint := answer-target; video-genre-type; time-duration

The *Content* parts models the user's precise information needs, while *Constraint* specifies the expected answer types.

The query analysis aims to classify the query into one of 8 main question classes or answer targets as shown in Table 1: *Human, Location, Organization, Time, Number, Object, Description and General*. The last answer target *General* is used to group questions that cannot be categorized into the other classes. The answer target found will be used to locate precise sentences that contain entity of this type. The analysis module also induces the possible genre type of video in one of 14 types as explained in Section 2, with the addition of a *General News* video type. Our system employs a rule-based question classifier to determine the answer target and video genre type and could achieve an accuracy of over 95%. The examples are given in Table 1.

Table 1: Question classification and possible video genres
* Show only the likely video genres for the specific question examples

Answer Target	Likely Video Genre	Example
Human	Anchor, meeting, speech, General-news	Who is the Secretary of State of the United States?
Location	Live report, Anchor, General-news	Where is Saddam Hussein hiding?
Organization	Live report, anchor	Which hospital is the center for SARS treatment in Singapore?
Time	Anchor, General-news	When did the Iraq war start?
Number	Finance	What is the expected GDP of Singapore this year?
	Sports, Text-scene	How many points did Yao Ming score?
	Weather, Text-scene	What is the highest temperature tomorrow?
Object	Anchor, Still-image, Text-scene	Which kinds of bombs are used in the current Iraq war?
Description	Anchor, Text-scene	What does SARS stand for?

The query analysis also extracts important content information that is crucial for later processing. Detailed analysis is performed here in order to get as much useful information as possible. The three kinds of word groups that we extract from the original query are:

a. ***Query Words:*** These include nouns, adjectives, numbers, and some non-trivial verbs that appear in the question string. For example: "*Which company is the first to find SARS patient in Singapore?*", the content word vector will be $\underline{\mathbf{q}}^{(0)}$**: (company, first, SARS, patient, Singapore).**

b. ***Base Noun Phrases:*** we use noun phrase recognizer to identity all base noun phrases appearing in the query. For the above example, the base noun phrase vector $\underline{\mathbf{n}}$**: ("SARS patient")**

c. ***Named Entities:*** They refer to noun phrases that represent Person, Organization, Location, Time, Number, and Object etc. For the above example, the named entity vector $\underline{\mathbf{h}}$**: ("SARS", "Singapore")**

Table 2 shows the analysis of the above two query examples.

Table 2: Question analysis

Question	*What is the score of the football match last night?*	*What are the symptoms of atypical pneumonia?*
$\underline{q}^{(0)}$	score, football, match, last, night	symptoms atypical pneumonia
$\underline{n}$	*football match, last night*	symptom, *atypical pneumonia*
$\underline{h}$	football	*atypical pneumonia*
Answer Target	Number	Description
Video Genre	Sports, Text-scene	General News

4.2 Query Reinforcement

Given a short, fact-based question of the form of $\underline{q}^{(0)} = [q_1^{(0)}\ q_2^{(0)} \ldots q_k^{(0)}]$, the problem for retrieving all the video segments relevant to $\underline{q}^{(0)}$ is that *the question does not provide sufficient* *hints for us to locate the targeted answer*. We resort to using general open resources to overcome this problem by exploring the external knowledge from the Web and WordNet.

As with the pre-processing stage in correcting speech recognition errors, we go to the news web sites to retrieve the latest news articles related to the query and used them to extract the context for the query. To achieve this, we use the query words $\underline{q}^{(0)}$ in the question to retrieve the top N_w news articles from, say, the CNN or AltaVista news websites. We then extract the terms in these articles that are highly correlated with the query words within a multi-sentence window in the full web document. For each term $q_i^{(0)} \in \underline{q}^{(0)}$, we extract the list of nearby non-trivial words, $\underline{w}_i$, that are within the local context. We further compute the weights for all terms $w_{ik} \in \underline{w}_i$ based on the probabilistic support of their occurrences with $q_i^{(0)}$ as:

$$\frac{d_s(w_{ik} \wedge q_i^{(0)})}{d_s(w_{ik} \vee q_i^{(0)})} \tag{9}$$

where $d_s(w_{ik} \wedge q_i^{(0)})$ gives the number of sentence-windows that contain both w_{ik} and $q_i^{(0)}$; and $d_s(w_{ik} \vee q_i^{(0)})$ gives the number that contains either w_{ik} or $q_i^{(0)}$. We merge all $\underline{w}_i$ to form context word list $\underline{C}_q$ for $\underline{q}^{(0)}$. For question "*What are the symptoms of atypical pneumonia*?", the original query is "*symptoms, atypical, pneumonia*", and the expanded query after using the web knowledge becomes *"symptoms, pneumonia, spread, virus, fever, cough, hospital, atypical, doctor, Asia"*.

Various studies (Clarke et al 2002, Brill et al 2002) have shown that the Web is useful at finding the world knowledge by providing the words that occur frequently with the original query terms in the local context. However, it lacks information on lexical relationships among these terms, such as synonyms. To extract such knowledge, we use WordNet (Leacock et al 1998). to get the gloss (definition of terms) words $\underline{G}_q$ and synset (synonym sets) words $\underline{S}_q$ for $\underline{q}^{(0)}$ in order to provide more useful terms related to the event. In order to ensure that we do not assign words in $\underline{G}_q$ and $\underline{S}_q$ out of context, we restrict only to those terms in WordNet that appear in the first or popular senses (or usage) of $\underline{q}^{(0)}$ terms.

Next, we combine the external knowledge sources by adding the words in $\underline{C}_q$, and those in $\underline{G}_q$ and $\underline{S}_q$ to form $\underline{K}_q$:

$$\underline{K}_q = \underline{C}_q + (\underline{G}_q \cup \underline{S}_q) \tag{10}$$

At the end of this process, $\underline{K}_q$ contains many terms relating to the implicit event expressed in the user's query. We increase the weights for the terms appear in both the Web and WordNet, i.e., we let WordNet work like a filter to refine the additional terms. The final weight of each term is normalized and the top *m* terms above the cut-off threshold σ are selected to get the context word vector $\underline{q}^{(1)}$:

$$\underline{q}^{(1)} = \underline{q}^{(0)} + \{\text{top } m \text{ terms} \in \underline{K}_q \text{ with weights} >= \sigma\} \tag{11}$$

where *m* is initially set to 20 in our experiments.

Here $\underline{q}^{(1)}$ should contain more context words than $\underline{q}^{(0)}$ and hence more knowledge about the particular query. After the lexical reinforcement by WordNet, the resulting context word ranked list will contain the refined context words. For the *"pneumonia"* example, the refined query is *"symptoms, pneumonia, virus, spread, fever, cough, breath, doctor"* See the example shown in Figure 5.

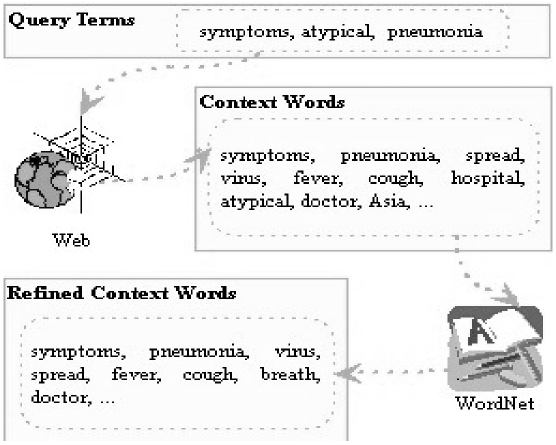

Query Terms
symptoms, atypical, pneumonia
Context Words
symptoms, pneumonia, spread, virus, fever, cough, hospital, atypical, doctor, Asia, ...
Web
Refined Context Words
symptoms, pneumonia, virus, spread, fever, cough, breath, doctor, ...
WordNet

Figure 5: Example of query reinforcement

4.3 Transcript Retrieval&Answer Extraction

We use the MG tool (Witten et al 1999) to index the transcripts at story level. Given the expanded query $\underline{q}^{(1)}$, we perform similarity retrieval to obtain a ranked list of transcripts. For each transcript, we use the following criteria to rank the relevance of a sentence to the question: *(Recall that from query processing, we extracted* $\underline{q}^{(0)}$*,* $\underline{n}$*,* $\underline{h}$*)*. For each sentence Sent_j, we match it with:

- noun phrases: W_{nj} = % of phrase overlap between $\underline{n}$ and Sent_j
- named entities: W_{hj} = 1 if there is a match between $\underline{h}$ and Sent_j; and 0 otherwise.
- original query words: W_{cj} = % of term overlap between $\underline{q}^{(0)}$ and Sent_j.
- expanded query words: W_{ej} = % of term overlap between $\underline{q}^{(1-0)}$ and Sent_j , where $q^{(1-0)} = \underline{q}^{(1)} - \underline{q}^{(0)}$.

The final score for the sentence is:

$$S_j = \sum_i \alpha_i W_{ij} \tag{12}$$

where $\sum \alpha_i = 1$ and $W_{ij} \in \{W_{nj}, W_{hj}, W_{cj}, W_{ej}\}$. The top *K* sentences are then selected as the candidate answer sentences based on S_j.

The top sentence returned for the above "*pneumonia*" example is: *"Symptoms include high fever, coughing, shortness of breath and difficulty breathing."*

4.4 Video Summarization

Given on the set of news transcript sentences extracted, we perform dynamic news video summarization to generate the video answer. The list of sentences corresponds to appropriate audio fragments in the video story. This gives the constraint on the duration of the video to be shown. The task here is to extract appropriate visual segments, to be shown along with the (audio) sentences, that are both informative and interesting to the users. These two criteria suggest that even though we should show visual segments correspond to the selected transcript sentences; we should also pack as much variety of video genre types within the news story as possible. Also, we should avoid showing too much *Anchor-person* shots (Christel et al 2002), even though the selected news transcript sentences are likely to come from such shots.

The algorithm for generating visual summary subject to the duration constraint imposed by the selected (audio) transcript sentences is as follows:

a) We remove those shots that are shorter than 4 seconds in duration.

b) We pick those that overlap in duration with the selected transcript sentences and place them in *candidate* list.

c) We group video shots of the same genre type together, and perform clustering of shots within each genre type by using the 64-bin color histogram of the key frame for each shot. For each cluster, we select ***two*** representative shots for inclusion in the *candidate* list. We select one representative shot near the centroid, and the other near the boundary. This is to eliminate duplicate shots, and ensure variety in the selection of candidate shots in each cluster.

d) We compute the weight of each shot in the *candidate* list based on two criteria. First, whether it overlaps with the selected transcript sentences (w_t=1) or not (w_t=0). Second, whether it is of more interesting genre type (w_g). We set the "*interest index*" of each genre type, w_g, to a value ranging from 1 (high interest) to a smaller value. For example, for general news, the shot genres with high "*interest index*" are: *Live-reporting, Meeting/Gathering*; while for sports, the high "*interest index*" genres are: *Live-reporting, Text-scene*. The *Anchor-person, 2-anchor-person* shots have low "*interest index*". In the current system, we pre-defined the "*interest index*" of each genre. Further studies in this area need to be carried out.

e) We assign the weight to each shot in the *candidate* list using the following formula:

$$\mathrm{Wt}(\mathrm{shot}_i) = \beta_t w_t + \beta_g w_g \qquad (13)$$

We set β_t=β_g=0.5 in this test in order to strike a balance between choosing visual segments that fall within the transcript duration and those that are "interesting" to the users.

f) Finally, we select the top n candidate shots, where n is set to the largest integer less than the duration divided by 5. This is a heuristic to ensure that we can pack sufficient interesting shots, each with a duration of about 5 seconds. We trim the shots to fit into the duration of audio constraint.

Figure 6 illustrates the process of summarization. Suppose there are 6 sentences (represented by S1 to S6) and 5 shots (AN is *Anchor-person* shot, MT1, MT2, and MT3 are *Meeting/Gathering* shots, and SP is *Speech/Interview* shot) in the story. Assume that our QA engine selects sentences S2 and S4, and the visual summarizer selects MT1 and MT3 from the cluster of 3 *Meeting* shots, and SP of *Speech* genre. Eventually, we trim visual segments from MT1, SP, and MT3 to provide the summary.

For query "*What are the symptoms of atypical pneumonia?",* the 3-sentence window selected by the QA engine is:

S_1: *"He and his two companions are now in isolation and the one hundred and fifty five passengers on the flight were briefly quarantined."*

S_2: *"Symptoms include high fever, coughing, shortness of breath and difficulty breathing."*

S_3: *"But health officials say there's no reason to panic."*

Figure 7 shows the corresponding video summary extracted.

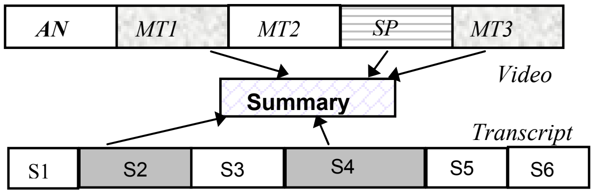


Figure 6: A scenario of general news video summarization

* Only keyframes are shown here.

Figure 7: Video summary of the "*pneumonia*" example

5. EVALUATION

We selected 7 days of CNN news video from 13-19 March 2003 for our test. We used two half-hour news segments per day, giving rise to a total of 350 minutes of news video. After performing news video story segmentation, we extracted a total of 175 video stories. For testing purposes, we also retrieved about 600 news articles per day from the Alta Vista news web site (Alta Vista 2003) during those 7 days. The total number of articles used is about 4,000. They are used as external resource during QA.

We designed 40 TREC-style (Voorhees 2002) questions related to the 7-day news. The questions belong to 8 different question classes at various difficulty levels. The list of questions is given in Figure 8. Among the 40 questions, 28 of them are general questions and are asked everyday during the test period. Most of these general questions will give different answers when posed on different days. The rest of 12 questions are date-specific and are relevant only to the specific day. This gives rise to a total of 208 questions. Most questions have answers in the video corpus. Only 4 questions do not have an answer (NIL-answer questions).

5.1 Impact of Video Transcript Correction

We first test the ability of our technique to correct the text recognition errors using a combination of phonetic-based and video-text matching techniques based on a name-list. From the Sphinx-III output of news transcripts for the 7-day news, we manually counted the number of ATs (3,155) and those that are wrongly recognized (1,227). Out of those in error, 762 can be found in Sphinx-III phonetic dictionary. Hence we can attempt to correct the errors in these ATs using the phonetic matching technique. The rest of those in error that are not found in the Sphinx's phonetic dictionary (465) would have to be corrected using the OCR matching technique.

The results of corrections are summarized in Table 3. Overall, we are able to correct 68% of errors, while introducing 513 new errors (or false positives). The results suggest that our transcripts error correction technique is effective. The ability to correct 68% of errors will greatly hence the ability of subsequent QA process to retrieve precise answers.

Table 3: Statistics & performance of transcript correction steps

# of ATs found in 175 news trasncripts	3,155
# of ATs with errors in the news transcripts	1,227
Process 1:	
# of ATs with errors found in Phonetic Dictionary	762
# of AT errors corrected by Phonic Matching	529
Accuracy of Process 1	529/762
Process 2:	
# of ATs with errors not found in Phonetic Dictionary	465
# of additional AT errors corrected by OCR matching	308
Accuracy of Process 2	308/465
% of errors corrected	68.2%
# of false positive	513

1) Who is the British Prime Minister?
2) Who is elected to be China's President?
3) Who is the President of the United States?
4) What is the name of the former Premier of China?
5) What is the name of the new Premier of China?
6) Who will pay the heaviest tallies?
7) Who was arrested in Pakistan?
8) Which musician called off his US tour?
9) When will NASA resume shuttle flights?
10) When will Germany, France and Russia meet?
11) When is the funeral of DjinDjic?
12) Which are the three countries involved in the summit today?
13) Where was the summit held?
14) Which city is the capital of Central African Republic?
15) Which are the three major war opponent countries?
16) To whom US withdrew the aid offer?
17) Which country vowed to veto the resolution today?
18) Which country's compromise proposal was rejected by US?
19) Where is Kashmir Hotel?
20) Where did Iraq invite the chief weapons inspectors to?
21) Which city has the largest anti war demonstration?
22) Where did a AL QUEDA suspect arrested?
23) How many people attended the rally in San Francisco?
24) What is the cost of war?
25) How many people were killed in a Kashmir Hotel?
26) How many people participated in the rally in Madrid?
27) How many people were killed by the new pneumonia?
28) What are the symptoms of the atypical pneumonia?
29) What sanction did President Bush lift?
30) What was the name of the space shuttle broken apart in February?
31) Which rally shows the support for President Bush?
32) What is the official name for the mysterious pneumonia?
33) Which company tests their new passenger profiling system?
34) Name one Jewish holiday.
35) What is British stance?
36) How did Serbs Prime Minister die?
37) How is the anti-war protest in Madrid?
38) How is tomorrow's weather?
39) What is the conflict between US and Turkey?
40) What does the WHO call the new pneumonia?

Figure 8: List of textual questions for news video

5.2 Question Answering Performance

Here we evaluate the ability of the QA system in returning the correct sentence containing the answer. That is, as long as the correct answer is contained in one of the three returned sentences, we consider the answer to be correct. In order to assess the effects of transcript error corrections, we test the performance of our QA system on: (a) the raw transcripts without error correction; and (b) the transcript with error correction.

Tables 4-6 tabulate the results. The results in Table 6 indicate that without transcript error corrections, our QA system could achieve a QA accuracy of about 55%. However, with the corrected transcript, the QA accuracy improves drastically to about 73%. The results demonstrate that our transcript error correction is useful, and it helps in realizing a usable video QA system with an overall accuracy of 73%.

The results in Table 4 reveal that our QA system performs reasonably well on general questions.

Table 4: Accuracy over 196 (28*7) general questions

Transcript	*Correct Answers*	*Accuracy*
without error correction	110	56.1%
with correction	143	73.0%

Table 5: Accuracy of over 12 date-specific questions

Transcript	*Correct Answers*	*Accuracy*
without error correction	6	50%
with error correction	10	93.3%

Table 6: Overall accuracy of 208 questions

Transcript	*Correct Answers*	*Accuracy*
without error correction	116	55.8%
with error correction	153	73.6%

5.3 Discussion of Results

In developing such a system that integrates technologies and research from many fields, many sources of errors may incur and needs to be tackled. Although cares have been taken to minimize errors, many errors do occur and affect the quality of the results. The main remaining sources of errors include:

- Errors in segmenting video sequence into story units, and in identifying sentence boundaries using the statistics of speech pauses. The cumulative error is about 15% for the news video that we are testing.
- Most non-English names are wrongly recognized. For those names that do not have equivalent phonetic representation in Dictionary, and also do not appear or wrongly recognized in video text, there is no means of correcting such errors. In fact, more than 10% of uncorrected ATs fall under this category. To further improve the performance of error correction, we need better video OCR tool and a speech recognizer that can better handle non-English names.
- Certain wrongly recognized words are hard to correct. For example, *Baghdad* is wrongly recognized as *burger*, and *Chile* as *chalet*. Both of these substituted words are meaningful single words and are thus hard to identify. To correct such error, we need to incorporate statistical language model to

determine the probability of occurrence of certain words or ATs in the language context and constructs.

6. RELATED WORK

The main emphasis of this work is on generating and correcting errors in video transcripts, and performing QA to extract precise answers. Our work is related to other research on speech and video retrieval. One of the most related works is the well-known *Informedia* project, which covers most aspects of feature extraction, segmentation, and retrieval of news video (Wactlar et al 2000, Christel et al 2002). Similar to our approach, they also utilized news transcripts and external news articles to help correct feature extraction errors. In particular, they used the name lists extracted from the news transcripts and external news articles to improve the accuracy in video OCR (Wactlar et al 2000), and in associating names to face (Satoh et al 1999). Our work differs from this in that we use the name list to constraint the phonetic search and OCR matching in correcting the speech recognition errors in the transcripts.

In the more recent work under the *Informedia* project, Christel et al (2002) introduced the idea of video collages as an effective interface for browsing and interpreting video collections. They extracted video stories, key phrases of news transcripts comprising mainly of names of person, location and organization; and other structured information. The system supports queries by users to retrieve information through map, text and structured information like date range. Differing from this work that supports database style search for information, we try to generate correct transcript at the sentence level with the aim to perform flexible question-answering.

Our work is also similar to other research in retrieving audio documents using speech queries. As one of the main sources of errors in speech recognition come from substitution, confusion matrix has been used to record confused sound pairs in an attempt to eliminate this error. Confusion matrix has been employed effectively in spoken document retrieval (Singhal et al, 1999) and to minimize speech recognition errors (Shen et al, 1998). However, such a method will bring in many irrelevant terms, when they are used directly to correct speech recognition errors (Ng 2000). Because important terms in a long document are often repeated several times, there is a good chance that such terms will be correctly recognized at least once by a speech recognition engine with a reasonable level of word recognition rate. Many spoken document retrieval (SDR) systems (Chen et al 2001) have taken advantage of this fact in reducing speech recognition and matching errors. In contrast to SDR that uses confused sounding pairs to expand spoken language query, or statistics of repeated sounds to correct substitution errors, we use a known name list, along with phonetic and OCR text matching techniques to correct substitution errors.

7. CONCLUSION

Many users are interested in searching for *information*, while the current video retrieval engines are designed to return only *video documents*. There are many simple factoid questions posed over news video collection where the users expect to acquire the video segments containing the short precise answers. Here we raise the topic of video question answering to support personalized news video retrieval. Users interact with systems using short natural language text with implicit constraints on contents, duration, and genre of expected videos. Our system VideoQA returns the relevant news video fragments as the answers, supplemented by text version of latest news, summarized to the duration constraint as specified by the users.

The realization of VideoQA system requires the integration of a range of technologies including: video story segmentation, speech recognition and correction to generate news transcripts, question-answering analysis to generate precise multi-sentence output, and summarization of news stories. The main contributions of this work are: (a) the extension of question answering technology to support QA in news video; and (b) the use of external knowledge and visual content analysis to help correct speech recognition errors and to perform precise question answering. Test on the 7-day of CNN news demonstrate that the approach is feasible and effective.

This work is only the beginning, more research needs to be carried out as follows. First, we need to further reduce the speech recognition errors using name lists and other multimedia cues. Next, we need to extract better mid-level features and to associate these features with concepts in audio transcripts. This will help in identifying interesting video segments as answers and summary. Finally, we need to explore interactive QA and permit users to retrieve answers of type video, text and other multimedia sources.

8. REFERENCES

[1] Alta-Vista news web site (2003). http://news.altavista.com/

[2] E. Brill, J. Lin M. Banko, S. Dumais, and A. Ng (2001). “Data-intensive question answering”, In Proceedings of the Tenth Text REtrieval Conference (TREC’2001), 393-400.

[3] Lekha Chaisorn, Tat-Seng Chua, and Chin-Hui Lee (2002). "The Segmentation of News Video into Story Units", Proceeding of IEEE Int’l Conference on Multimedia and Expo-ICME 2002, Lausanne, Switzerland, Aug 26-29, 2002

[4] Berlin Chen, Hsin-min Wang, and Lin-Shan Lee (2001). “Improved Spoken Document Retrieval by Exploring Extra Acoustic and Linguistic Cues”, Proceedings of the 7th European Conference on Speech Communication and Technology.

[5] M.G. Christel, A.G. Hauptmann, H.D. Wactlar and T.D. Ng (2002). "Collages as Dynamic Summaries for News Video", In the Proceedings of ACM Multimedia 2002, Juan-les-Pins, France, December 2002.

[6] T.S. Chua and J.M. Liu (2002). “Learning Pattern Rules for Chinese Named-Entity Extraction. AAAI’2002. Edmonton, Canada, Jul/Aug 2002. 411-418.

[7] C. Clarke, G. Cormack and T. Lynam (2001). “Web reinforced question answering.” In Proceedings of the Tenth Text REtrieval Conference (TREC’2001),673-680.

[8] Thomas H. Cormen, Charles E. Leiserson and Ronald L. Rivest (1990). “Introduction to algorithms”, published by McGraw-Hill Book Company.

[9] W. H.-M. Hsu and S.-F. Chang (2003). "A Statistical Framework for Fusing Mid-level Perceptual Features in News Video", (invited paper), ICME 2003, Baltimore, USA, July 6-9, 2003.

[10] C. Leacock & M. Chodorow & G. Miller (1998). Using Corpus Statistics and WordNet for Sense Identification. Comp. Linguistic, 24(1), 147-165.

[11] C.H. Lee (1998). "On stochastic feature and model compensation approaches to robust speech recognition", Speech Communication, 25, 29-47.

[12] Kenney Ng (2000). "Information Fusion For Spoken Document Retrieval", Proceedings of ICASSP'00, Istanbul, Turkey, Jun.

[13] Dan Roth (2003). "A SnoW-based Shallow Parser", a software down-loaded from: http://l2r.cs.uiuc.edu/~cogcomp/

[14] G. Salton, M.C. McGill (1983). Introduction to Information Retrieval", McGraw Hill.

[15] Shin'ichi. Satoh, Yuichi Nakamura & Takeo Kanade (1999). "Name-it: Naming and Detecting Faces in News Videos". IEEE Multimedia, Jan 1999, 22-35.

[16] K. Seymore, S. Chen, S. Doh, M. Eskenazi, E. Gouvea, B. Raj, M. Ravishhankar, R. Rosenfeld, M. Siegler, R. Stern, and E. Thayer (1998). "The 1997 CMU Sphins-3 English Broadcast News Transcription System", Proceedings of the 1998 DARPA Speech recognition Workshop.

[17] Liqin, Shen, Haixin Chai, Yong Qin and Tang Donald (1998). "Character Error Correction for Chinese Speech Recognition System", Proceedings of International Symposium on Chinese Spoken Language Processing Symposium Proceedings, 136-138

[19] A. Singhal and F. Pereira (1999). "Document Expansion for Speech Retrieval", Proceedings of the 22nd Annual International ACM SIGIR conference on Research and Development in Information Retrieval, 34~41.

[20] E.M.Voorhees (2002). "Overview of the TREC 2002 Question Answering Track." In notebook of the Eleventh Text REtrieval Conference (TREC'2002), 115-123.

[21] Howard D. Wactlar, Alaxander G. Hauptman, Micahael G. Christel, Ricky A. Houghton, and Andreas M. Olligschlaeger (2000). "Complementary video and audio analysis fro Broadcast News Archives", Communications of the ACM, February 2000, Vol 43. No. 2, 42-47

[22] I. Witten, A. Moffat, and T. Bell (1999). "Managing Gigabytes", Morgan Kaufmann.

[23] Hui Yang and Tat-Seng Chua (2003). Structured use of external knowledge for event-based open-domain question-answering". 26th Int'l ACM SIGIR Conference' 03. Jul/Aug. Canada. To appear.