# Dialogue management in (speech) QA systems

Boris van Schooten,
University of Twente, Netherlands
(IMIX project)

# Contents

- Intro QA systems
- Dealing with follow-up questions
- Dealing with noise
- Confirmation
- Implementation in Ritel
- Future directions

# Dialogue management in (speech) QA systems

Basic strategies:

- user follow-up questions
- system clarification

Other largely unexplored options:

- user feedback
- answer-specific queries

Existing systems: SmartWeb (German), Rits-QA (Japanese), Nara institute (Japanese), KAIST (English), Hitiqa (English)

Existing competitions: Trec, Qac have context tracks (follow-up questions)

# Ritel: example

S: Bonjour et bienvenue sur le service Ritel du Limsi. Souhaitez-vous couter le guide ?

U: quelle est la capitale du Zimbabwe

S: Harare est la capitale du Zimbabwe. Autre chose ?

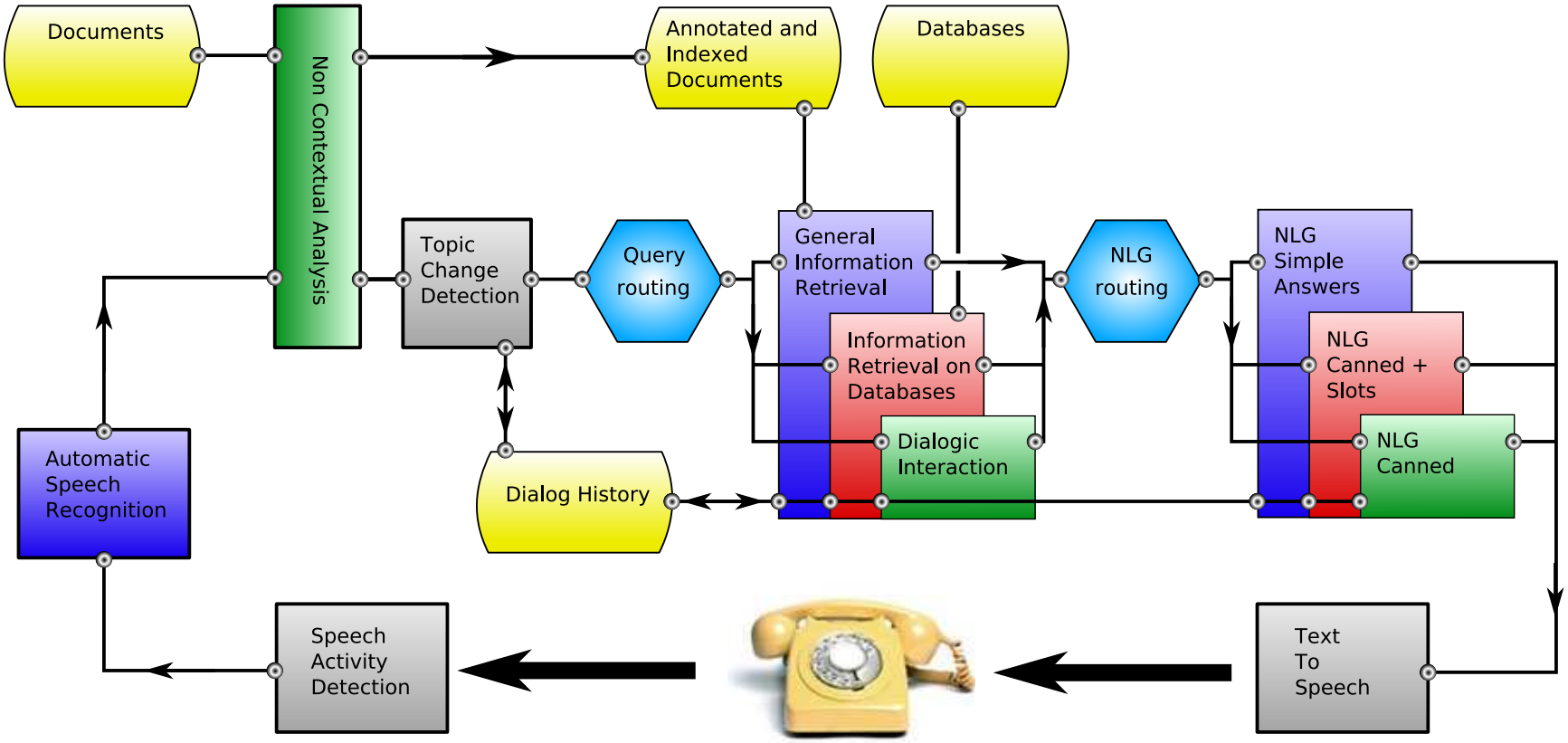
U: c' est de la France

S: Paris est la capitale de la France. Autre chose ?

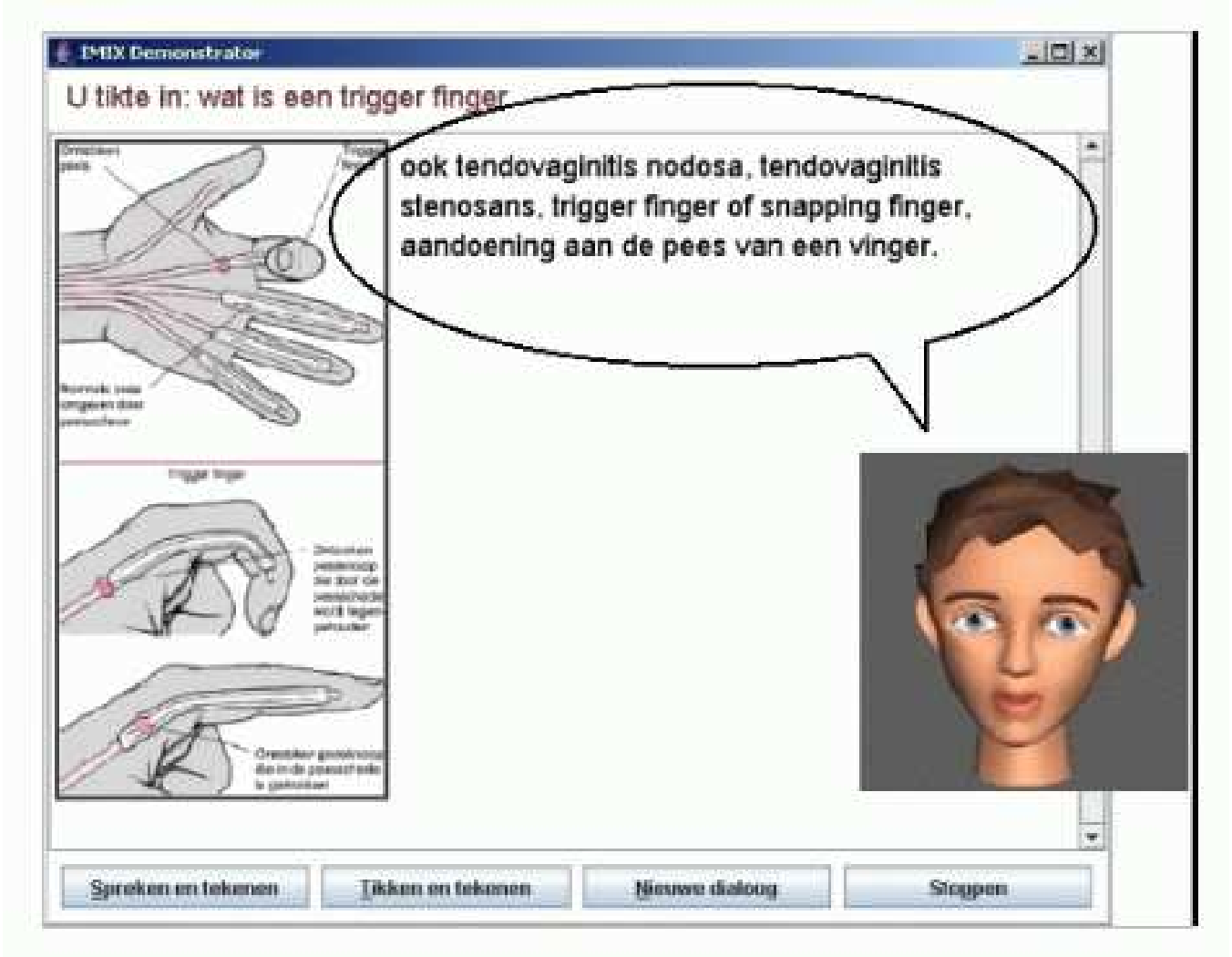
U: et quelle est sa superficie

S: la superficie de la France est de 1007329 km2 y compris les DOM-TOM. Autre chose ?

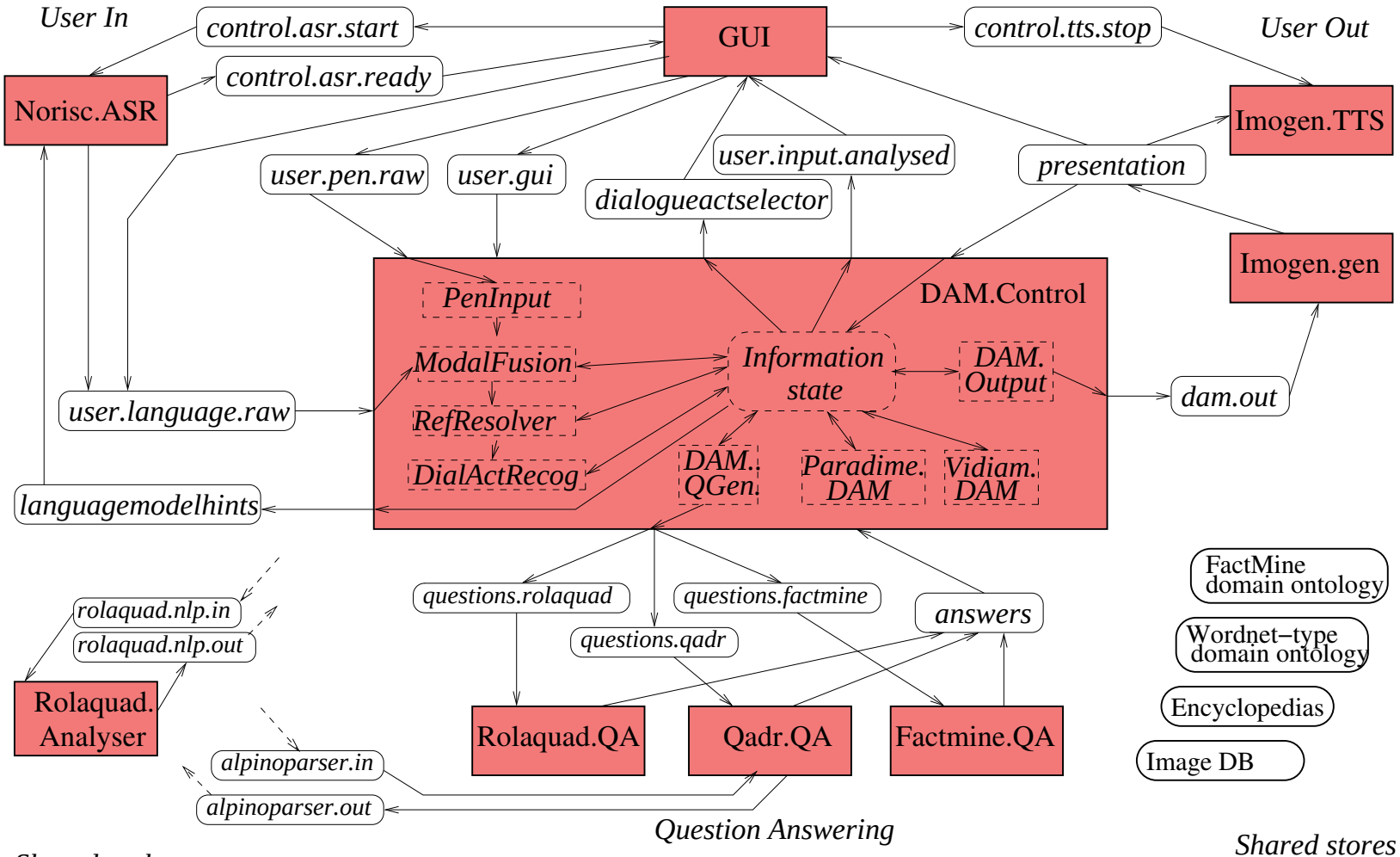
# Ritel: architecture



# IMIX: example



# IMIX: architecture




**ARCHITECTURE IMIX DEMONSTRATOR 3, 27 SEP 2006**

# QA issues: what is a question, what an answer?

| | |
|---|---|
| Factoid (ritel) | In: question type, keywords Out: word or phrase matching question type |
| Encyclopedic / definitionoid (imix) | In: question syntactic pattern Out: answer with matching pattern |
| List | |
| Complex, Analytic | |
| ... | |

Ritel keywords: person, location, event, time, ... (about 50)

IMIX keywords: disease, person, treatment. Relations: causes, treats, ... (about 15 in all)

# Follow-up questions: What do users do after they asked a question?

"Trec" follow-up questions

*Qui est le réalisateur de Titanic ?*
*Qui a écrit le scénario ?*

Non-factoid follow-up questions

*What do ribosomes produce?*
*All types of proteins?*
*And how does that work?*

Other "real life" follow-up utterances

*c'est une ville je vous ai déjà expliqué je peux pas mieux faire*
*I asked for effects, not causes!*
*So the answer is "no"?*

# Follow-up questions in IMIX and Ritel

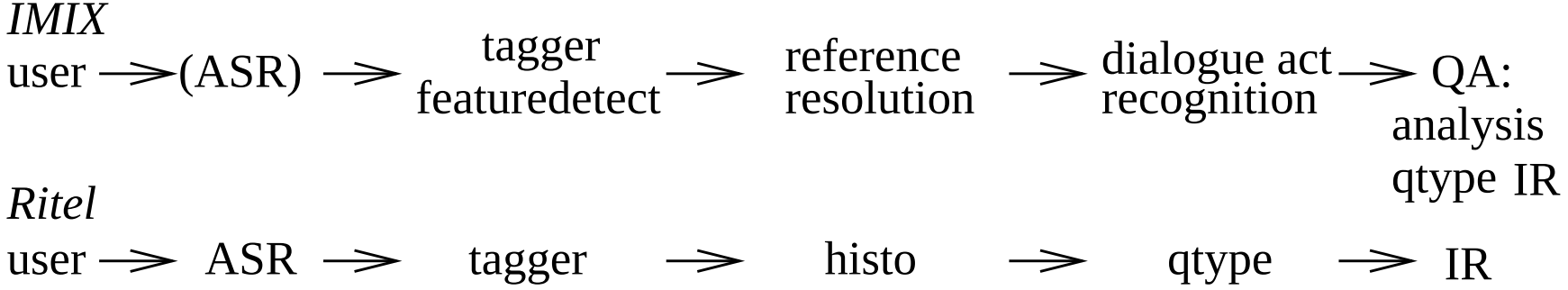
Both IMIX and Ritel use a history of salient keywords as context for follow-up questions.

Differences: reference resolution v context completion

**IMIX**: resolve explicit references or rewrite sentence to add most salient complementary keywords (suitable for black-box QA system).

**Ritel**: complete keyword set by adding type-complementary keywords from history (suitable for keyword-based system)

*Noise issues: the QA pipeline in IMIX and Ritel*




Approximate performance:

- Ritel asr:70%
- Ritel tagger:80-90% for named entities, mixed for other.

  Imix tagger/feature detector: approx 60-80%
- Ritel histo: unknown

  IMIX reference resolution: 40%
- IMIX dialogue act recogniser: 60-75%
- Ritel/IMIX question type recognition + QA: 30-50%

## Noise issues: ASR

The most serious problems with the current Ritel lie in the quality of ASR. In IMIX, we don't even have serious ASR.

Real life ASR:

- performance constraints means no proper word graph
- no proper confidence
- no proper out-of-vocabulary detection (however we found there were almost no out-of-vocabulary words in the ritel corpus)
- We found no obvious ways to improve the ASR by postfiltering

## Noise issues: ASR

But some useful findings:

- When a user repeats him/herself, a keyphrase that was recognised twice is 80% likely to be correct
- 92% of the general topics mentioned by users are recognised by the ASR.
- Named entities were in vocabulary and language model, with few exceptions. We found that "Hitchcock" and "James Dean" had very high priority in the language model.

## Confirmation

Other stages of the pipeline have similar problems. More general: we have noise and we don't know where it is; we wish to handle this using feedback from the user. That is:

- Confirmation of key phrases and question type
- Confirmation of answer
- Confirmation of topic or language model

But: we should keep the length of the question-answer cycle minimal, and the dialogue natural.

# Confirmation: history model

**Old Ritel**: no confirmation, mostly non-specific user elicitation combined with a single history completion model used for both follow-up questions and confirmation

**New adaptation**: 2 levels of dialogue context and a confirmation model

- current-question confirmation context. Used for both ASR error repair and formulation problems.
- follow-up question context

# Confirmation: history model

Why separate contexts for confirmation and follow-up questions?

Intuitively, they are very different. Formally, requirements are different too:

| | confirmation | follow-up |
|---|---|---|
| resolve reference? | N | Y |
| Which keywords to keep | confirmed | referred-to |
| Include answer? | N | Y |

*Sur quel continent se situe le Botswana ?*
*(ASR: sur quel continent se situe {fw} de)*

**confirmation context: kw = {continent, situe} qtype = location**
**follow-up context: (empty)**

S: Vous recherchez un lieu, mais plus précisément,
que souhaitez-vous savoir ?

*U: Le continent du Botswana.*
*(ASR: le continent du Botswana)*

**confirmation context: kw = {**$continent_{confirmed}$**, situe, Botswana }**
**qtype=**$location_{confirmed}$
**follow-up context: (empty)**

Ainsi, vous recherchez un lieu lié à Botswana et situe.
Je pense que c'est Afrique du Sud. J'écoute votre question.

*U: Quelle est sa superficie ?*
*(ASR: quelle est sa superficie)*

**confirmation context: kw = {superficie} qtype = nombre**
**follow-up context: kw = {continent, situe, Botswana}**

# Confirmation

In QA it is not necessarily possible to paraphrase the question to the user. In Ritel, question is actually a set of keywords with types, plus an expected answer type.

**Implicit or explicit confirmation?** Our paradigm is to enable users to ask questions at any time, and assume that any utterance can be understood as (partial) question. (this is basically the same as the original)

Which implies implicit confirmation.

# Confirmation: prompts

**Confirming question type.** Question type is intuitive enough to make it transparent to users.

*For example: "You are looking for a person. Could you be more specific?"*

**Confirming key words/phrases.**

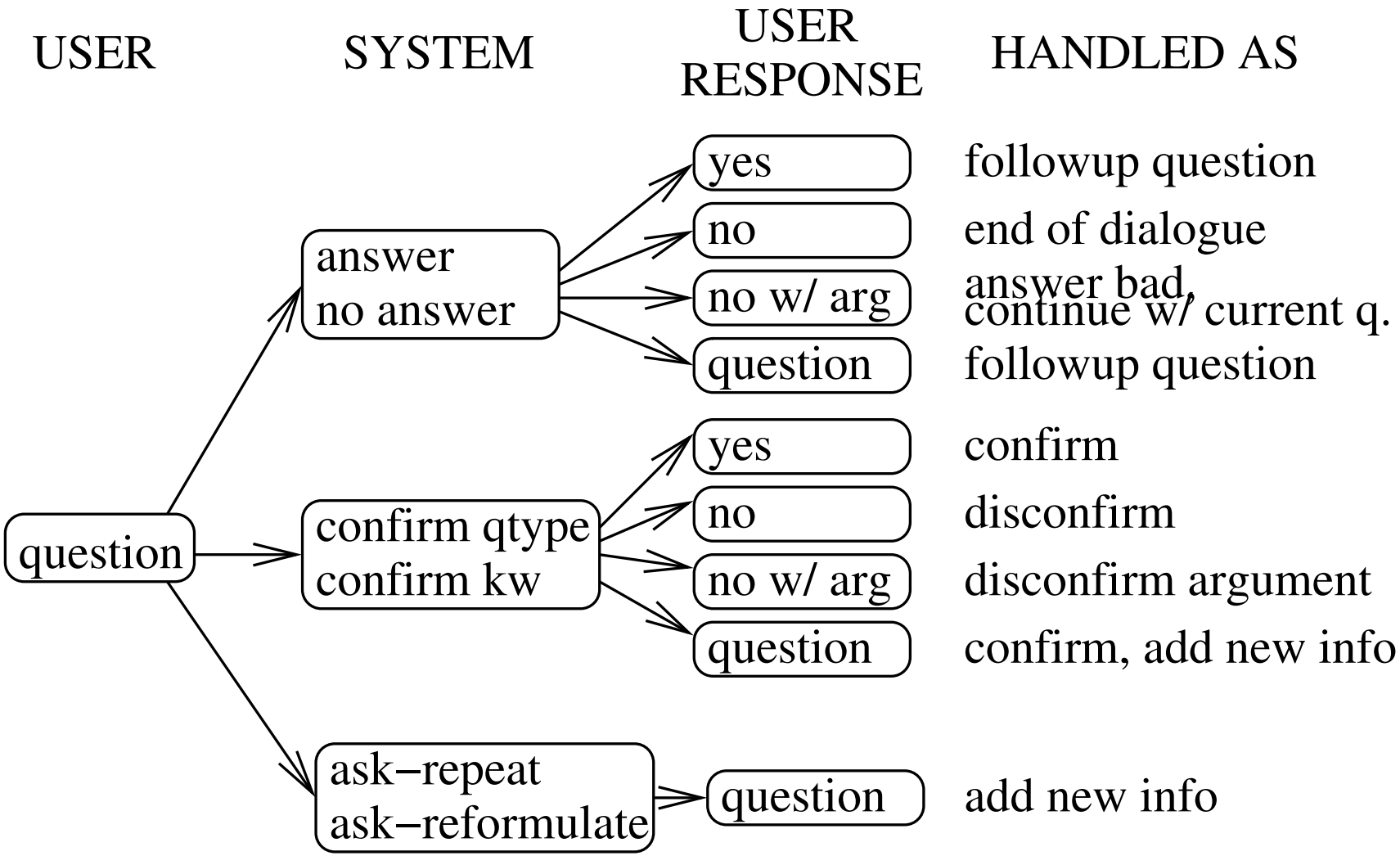
*For example: "You are looking for information about France. What kind of information are you looking for?"*

# Implementation: user act recognition

User acts (may be multiple ones in one utterance):

- question
- inform (question keywords or information)
- yes, acknowledge (confirm)
- no, negative (disagree, disconfirm specific things)
- bye
- (hello)

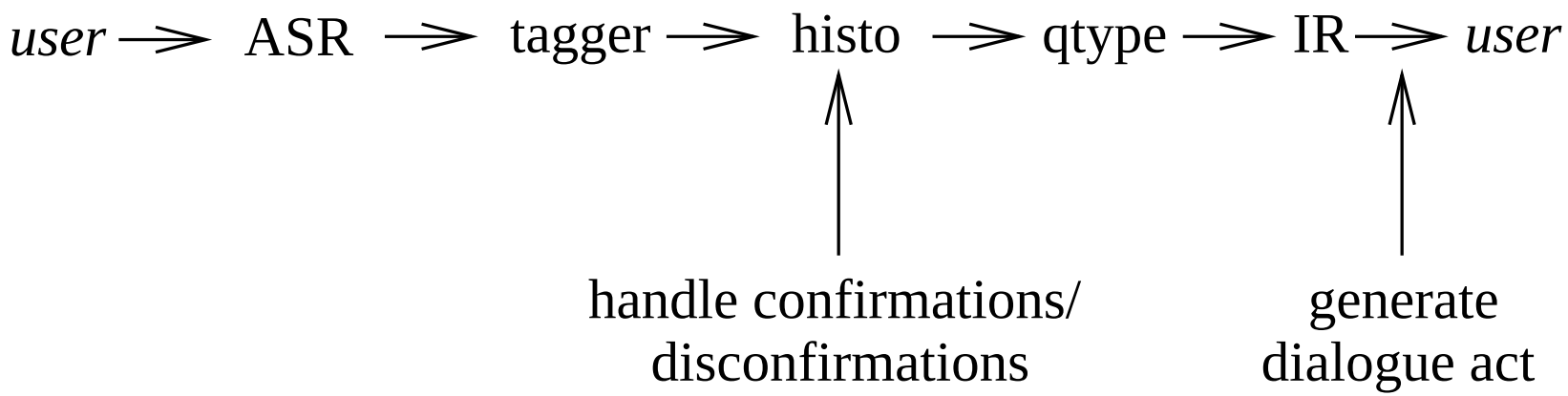
# Implementation: dialogue strategy



# Implementation: Input features for dialogue management

- ranked keywords
- question type or failure to find it; cues that lead to the question type hypothesis (as certain nonspecific cues are less certain)
- negative/positive feedback detector
- topic change detector
- QA success/failure and confidence?
- "verify question" detector? (for users questioning the system's answer)
- morphological/subject similarity detector? (for "No, I meant ..." type questions)

# Implementation: dialogue manager augmentations



# Future directions: porting to IMIX

Main problem: multiple QAS; QAs are black boxes.

- Keyword confirmation model can be ported
- Use a generalised substitute for question type?
- Passing the confirmed result to the QA is problematic. This would require generating a sentence with a structure that is useful for the QA.

# Future directions: HCI issues

- What prompts to use? Confirm what, how much, when?
- Is implicit confirmation reacted to properly?
- Difference between asking for a repeat and for a reformulation?

# Future directions: various ideas

- treat user speech as incremental: wait until there is enough information
- use multiple parallel ASRs with one language model for each topic
- indicate the backoff level by a prompt.