MobileVoice - 2010

Customizable Spoken Dialog Question Answering For Mobile Find

Mithun Balakrishna, Ph.D

Research Scientist

Lymba Corporation

mithun@lymba.com

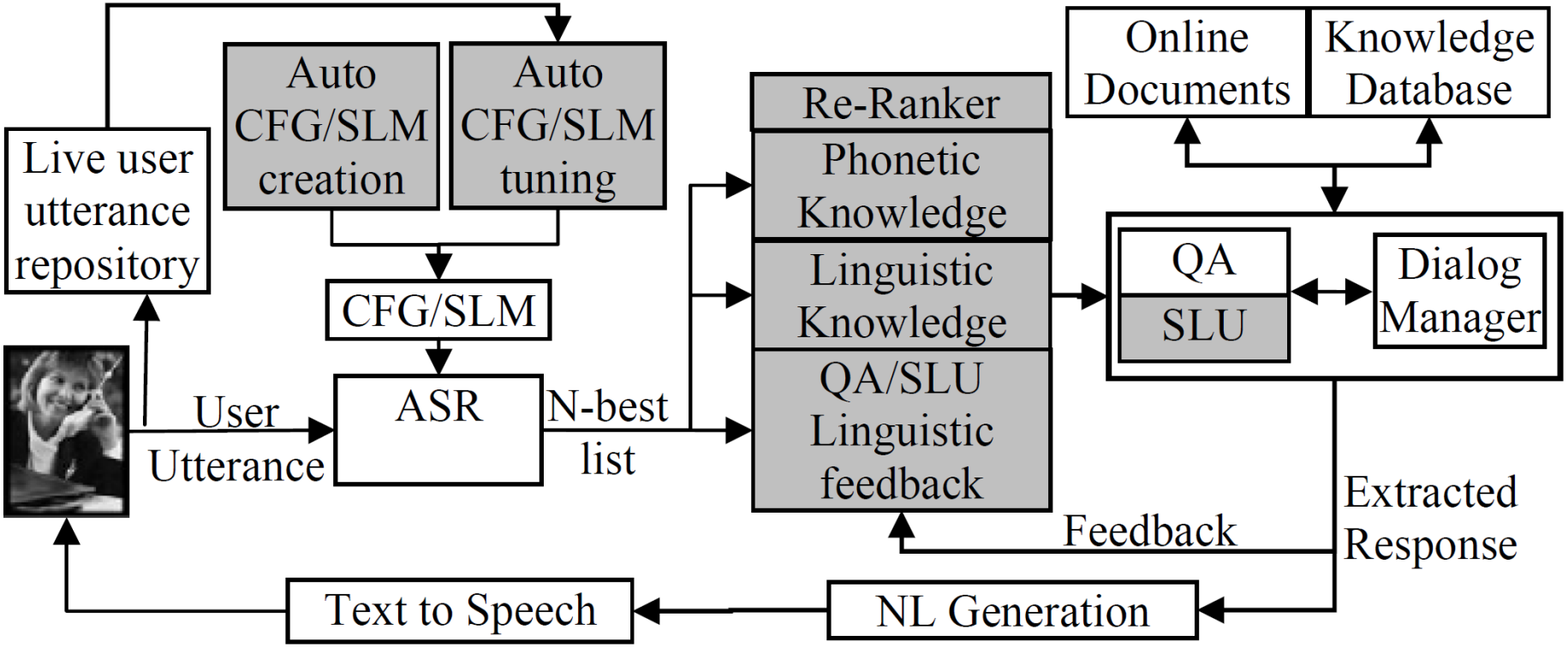
Lymba SDQA - Overview

- Lymba's SDQA focuses on building easily customizable SDQA Mobile Find applications by efficiently integrating the state-of-the-art in
  - Natural Language Processing (NLP)
    - Question Answering (QA)
    - Text Understanding,
  - Speech Technologies
    - Automatic Speech Recognition (ASR)
    - Spoken Language Understanding (SLU)
    - Spoken Dialog Management.
- Focus on designing techniques to get the best ASR/SLU result at each state in a human-machine dialog while decreasing the development/operational overhead
- Technical feasibility study funded by National Science Foundation's SBIR program

Lymba SDQA - Overview

- In our feasibility study, we developed synergistic mechanisms to:
  - generate efficient Statistical Language Models (SLMs) with minimum manual intervention and data
  - automatically tune the SLMs using live-user data with no human annotation
  - efficiently understand user speech using a knowledge-based approach while minimizing the manual effort in creating the understanding models
  - improve transcription performance by processing high-level domain dependent/independent phonetic and linguistic information resources, and QA/SLU filtering/feedback in an post-processing (n-best re-ranking) framework

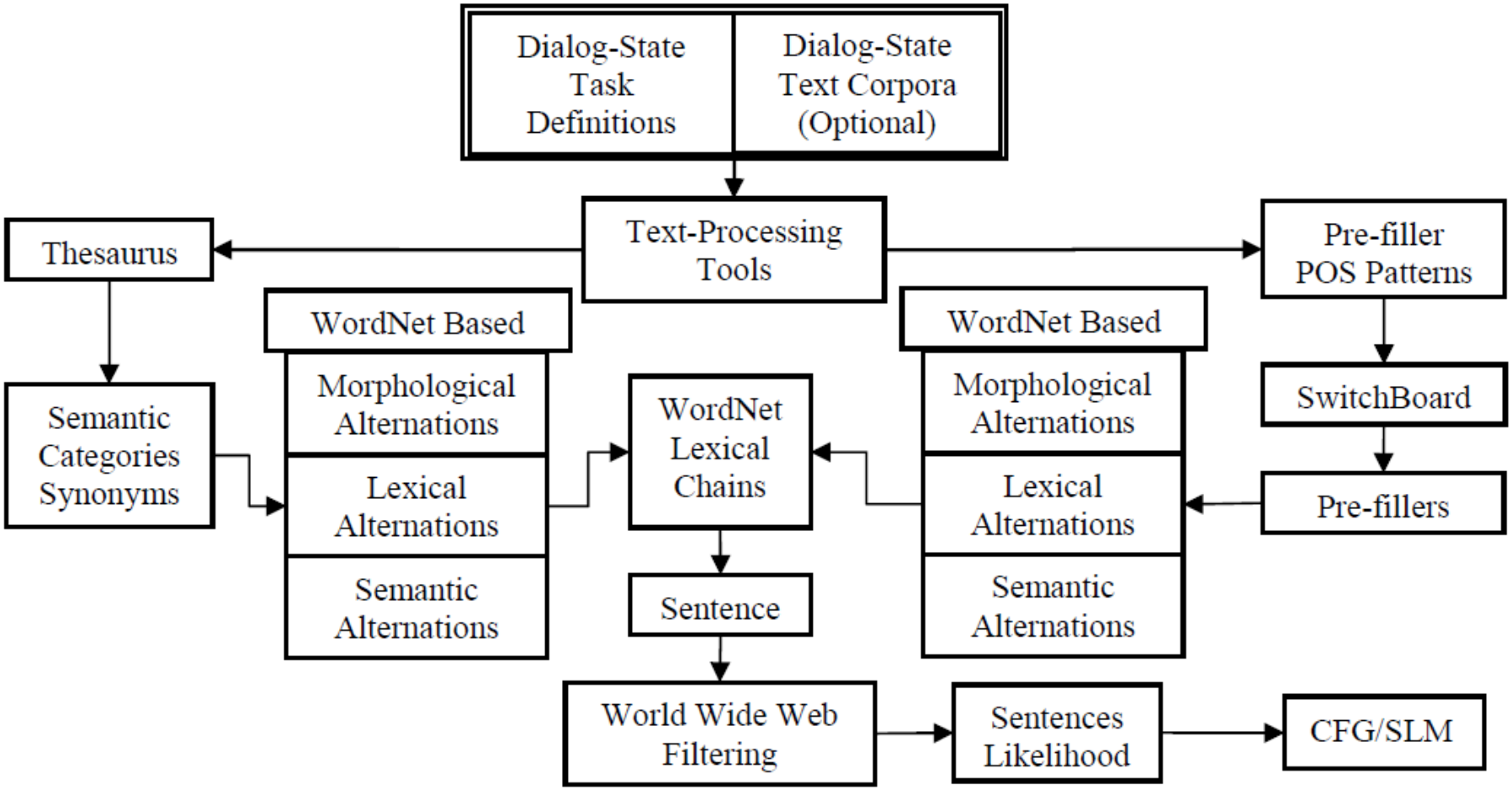
Lymba SDQA - Architecture



Semi-automated CFG/SLM Generation

- **Semi-automatic generation of context free grammars (CFGs) or statistical language models (SLMs) with minimum manual intervention and data requirement while achieving good speech recognition and understanding performance**
  - How do we find statistically valid, accurate, content-word variations that users can utter to perform the various semantic tasks available at a particular dialog state?
  - How do we find the required pre-filler and post-filler variations that users can utter to perform the various semantic tasks available at a particular dialog state?
  - How do we find all the valid sequence variations for all the possible argument-labels that can be filled by a user utterance for the same semantic category at a particular dialog state?
  - How do we assign sensible and practical statistical numbers to all the possible utterance variations for all the possible semantic categories?
  - How do we perform the entire above task automatically or semi-automatically (minimizing the human involvement in design, data collection and modeling)?

Semi-automated CFG/SLM Generation



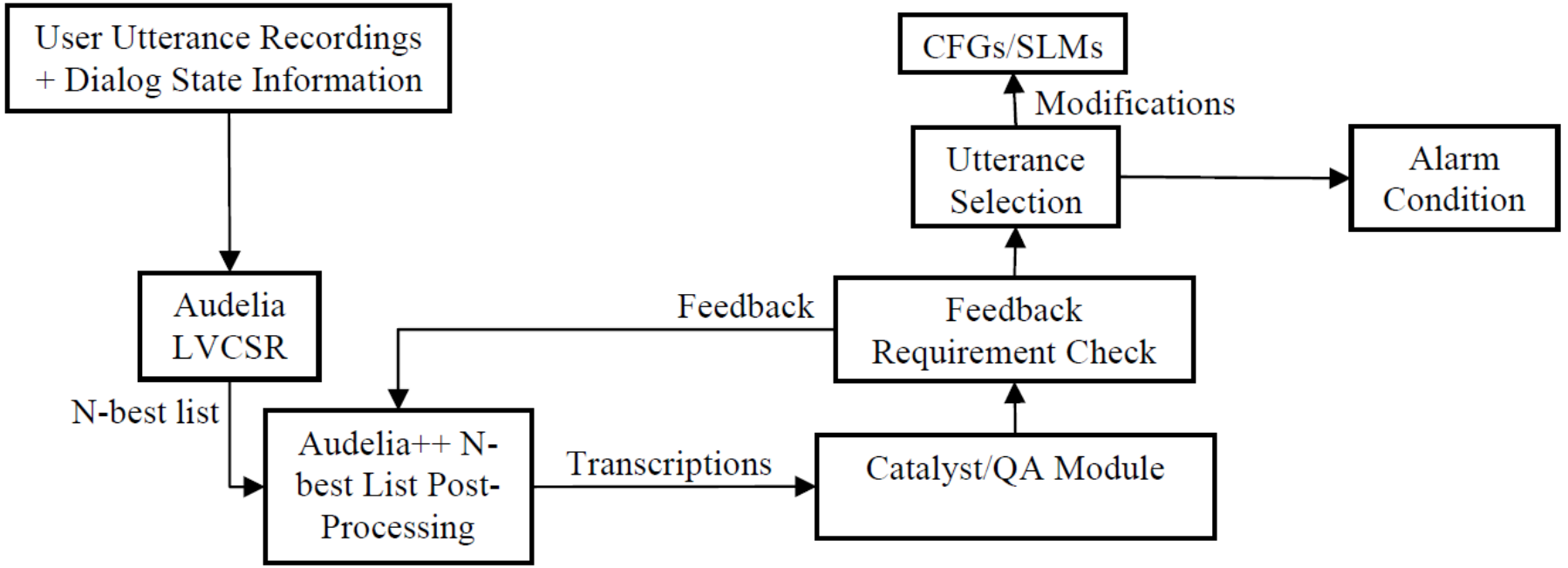
Semi-automated CFG/SLM Generation

- EVALUATION:
  - 28,436 live-user utterances for 5 different domain-constrained, telephony speech applications covering 38 unique dialog-states with around 12.5 choices on average at each state
  - Baseline acoustic models along with the manually created SLMs and SLU rules, for each of the 38 unique dialog states
  - The baseline system produced an overall 21.2% WER and 12.4% SemER (real-time speed)
  - Our performance was consistently close to the performance of the baseline system at most of the dialog-states
    - overall performance was 25.4% WER and 13.3% SemER
    - real-time speed
    - Catalyst is used as the SLU module using the same set of input, manual, description-seeds
  - Manually generating the SLMs (1.2 person-days Vs. 59.4 person-days for the baseline system)

Automated CFG/SLM Tuning

- **Automatic tuning of CFGs or SLMs, using minimal live-user data and no human annotation, to improve ASR transcription accuracy**
  - How do we accurately transcribe the collected live user utterance, without any manual intervention, to enhance the language models with new data or knowledge?
  - Given the dialog-state and its list of possible tasks, how do we map the transcribed utterances to the semantic categories/tasks and their corresponding variable slots?
  - How do we filter out invalid user utterances and extract valid variations to the possible semantic categories?
  - How do we assign sensible and practical statistical numbers to the new utterance variations for all the possible semantic categories?
  - How do we perform the entire above task automatically and efficiently? (no human involvement in transcription and extracting valid responses; the utterance extraction/addition/deletion mechanism should be efficient to improve the overall ASR performance)

Automated CFG/SLM Tuning



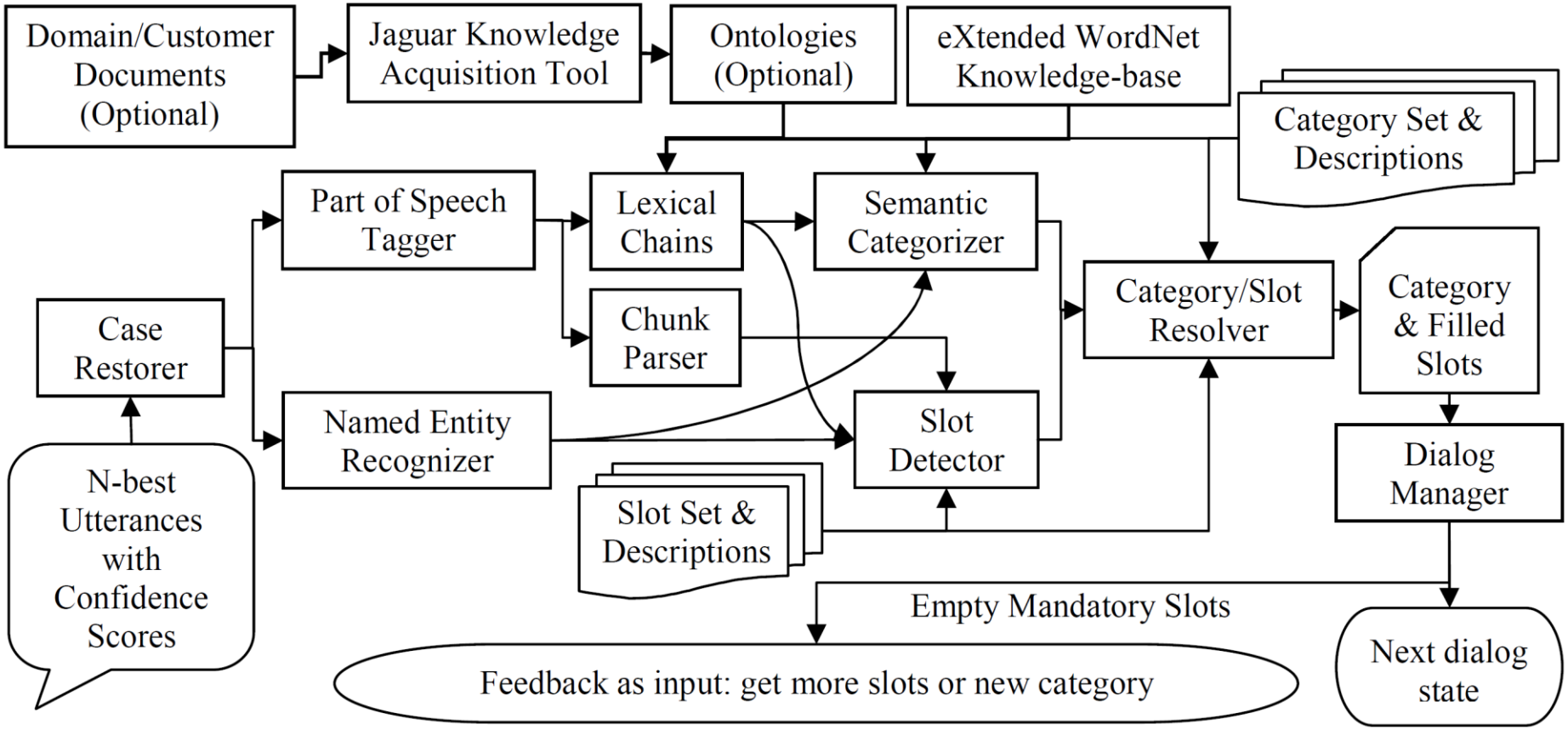
Automated CFG/SLM Tuning

- EVALUATION:
  - 28,436 live-user utterances for 5 different domain-constrained, telephony speech applications covering 38 unique dialog-states with around 12.5 choices on average at each state
    - 50% data for development/tuning
    - 50% for testing
  - Baseline WER and SemER results are produced using the baseline acoustic models along with the manually tuned SLMs and SLU rules for each of the 38 unique dialog-states
    - models and rules from the DiDaktos evaluation baseline used as a starting point
  - The real-time baseline system produced an overall 19.6% WER and 10.5% SemER
  - Our performance was consistently close to the performance of the baseline system and beating the human-tuning at some of the dialog-states
    - overall real-time performance was 22.6% WER and 11.6% SemER
    - Catalyst is again used as the SLU module using the original set of input, manual, description-seeds
  - These are very good results because Harmony tuned the SLMs automatically while the manual tuning effort took 47.2 person-days (manual transcription + manual categorization + manual LM modification)

Catalyst- Knowledge-based Spoken Language Understanding (SLU)

- **Automatic extraction of semantic content from ASR transcriptions; minimizing the human involvement and data required for creating/designing the understanding models.**
  - How do we accurately create understanding models, without any manual intervention to create annotated data samples or understanding rules?
  - Given the dialog-state and its list of possible tasks, how do we map the transcribed utterances to the semantic categories/tasks and their corresponding variable slots?

Catalyst- Knowledge-based Spoken Language Understanding (SLU)



Catalyst- Knowledge-based Spoken Language Understanding (SLU)

- Dialog State: Cable Account Change
- ASR Transcription: *I settled my bill through post*
- Semantic Category (SC): *payment method mail*
- SC Description: *Users can mail to pay their bills*
- Best 3 Lexical Chains:
  - settle#v#15 to pay#v#3 . [HYPERNYM,0.14 (pay#v#3 pay off#v#4 make up#v#3 compensate#v#5)] DISTANCE: 0:14
  - bill#n#1 to bill#n#1 . DISTANCE: 0:00
  - post#n#8 to mail#v#1 .[DERIVATION,0.14 (mail#v#1 get off#v#9)] DISTANCE: 0:14
- Semantic Similarity Measure: 0:28

```
For each user utterances A
  For each B in top ten ASR-transcriptions
    For each C in Semantic Categories set
      For each description D for C
        If ValidMapping(B,D)
          UpdateBestSimilarityMeasure(B,C)
          Similarity(A,C)=+BestMeasure(B,C)
  For each C in semantic categories set
    NormalizeSimilarityMeasure(A,C)
```

Catalyst- Knowledge-based Spoken Language Understanding (SLU)

- EVALUATION:
  - 28,436 live-user utterances for 5 different domain-constrained, telephony speech applications covering 38 unique dialog-states with around 12.5 choices on average at each state
  - Manual SLU rules vs.Catalyst, on manual transcriptions
    - Catalyst outperformed manual SLU rules by producing 3.6% SemER against 6.5% SemER from manual rules
  - Manual SLU rules vs. Catalyst, on random-error manual transcriptions (15% of words in the manual transcriptions were randomly changed)
    - Catalyst outperformed manual SLU rules by producing 4.3% SemER against 7.7% SemER from manual rules
  - These are very good results when considering that the human involvement in seeding the Catalyst system was very small when compared to the amount of time spent on manually generating the SLU rules
    - 1.2 person-days for Catalyst vs. 6.8 person-days for the manual SLU rules

Conclusion

- A novel methodology to use different knowledge sources to produce reliable domain-specific SLMs
- Minimal human intervention for creating good utterance alternatives
- Automatic tuning of CFGs or SLMs, using minimal live-user data and no human annotation, to improve ASR transcription accuracy
- Novel lexical chain based semantic category classification and feedback based on semantic classification strength