intent-bear

AIRBUS ATC challenge

CENTER FOR LANGUAGE AND SPEECH PROCESSING

UNIVERSITY OF WEST BOHEMIA

intent-bear ... WHO ARE WE?

University of West Bohemia - Department of Cybernetics (Pilsen, Czech Republic)

- Luboš Šmídl - smidl@kky.zcu.cz
- Jan Švec - honzas@kky.zcu.cz

Johns Hopkins University - Center for Language and Speech Processing (Baltimore, USA)

- Jan “Yenda” Trmal - jtrmal@gmail.com

intent-bear ... WHY AIRBUS ATC CHALLENGE?

- experience from the IT-BLP project
  - Intelligent technologies for improving air traffic security
  - Supported by GAČR 2011-2015

- cooperation UWB & JHU
  - JHU - CLSP - KALDI developer

CENTER FOR LANGUAGE AND SPEECH PROCESSING

UNIVERSITY OF WEST BOHEMIA

IT-BLP project

Tasks:

- Collect, process, and transcribe approx. 200 h of recordings from ANS/RLP Praha
- ASR (web demonstrator using technologies WebRTC, SIP, WebSockets, LVCSR, Tornado, Python)
- TTS - specific voices with accent (Czech, British, American, Serbian, German, Polish, France, Chinese, …)
- aTT - automatic training tool (video: goo.gl/zn6kU8)
  - Web application for creating teaching/learning material
- aPP - automatic pseudo-pilot (video: https://goo.gl/JwdJCv)
  - Multimodal dialog system designed as a learning tool for air traffic control officer trainees (ATCO)

Demo & technological demonstrator (year 2015):

- itblp.zcu.cz/

CENTER FOR LANGUAGE AND SPEECH PROCESSING

UNIVERSITY OF WEST BOHEMIA

APP - Automatic Pseudo-Pilot

A multimodal dialogue system for ATC trainees

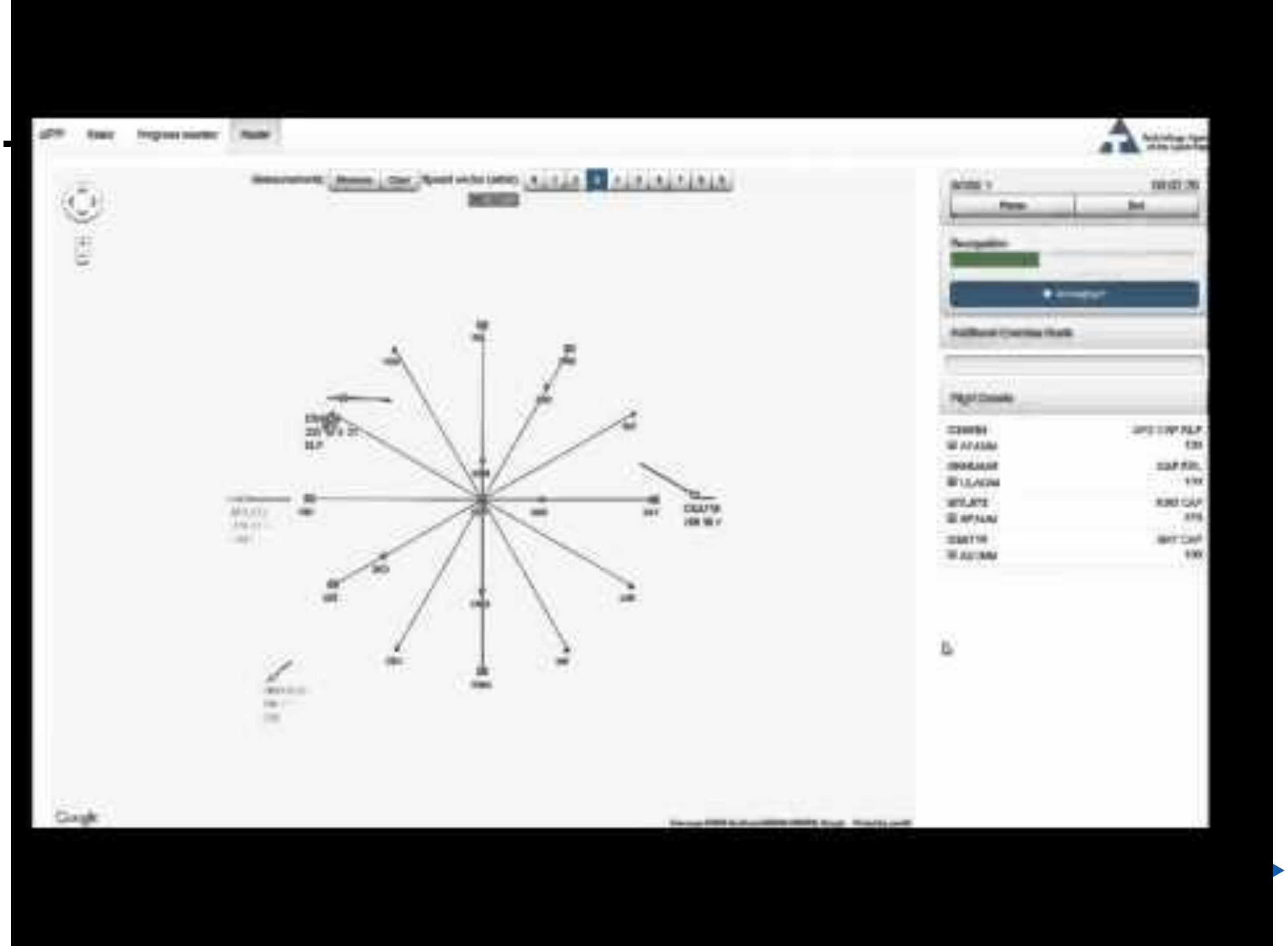
Functionality:

- understand ATC's utterance (ASR+SLU) + answer (TTS +noises)
- control air traffic generator - ATG
- show simulated radar screen - HTML5
- GUI of the dialogue system
- shows output of ATG, connects to ASR and TTS
- evaluate user's performance
- recorded radar screen with timeline of user's actions
- flight statistics for each airplane
- create different situation to exercise
- assign flight plans and additional goals

CENTER FOR LANGUAGE AND SPEECH PROCESSING

UNIVERSITY OF WEST BOHEMIA

APP -



ATC Challenge

Leaderboard results:

- 2nd place (harm mean: 0.98)

Test results:

- 4th place (WER 0.0876, F1 0,7704)

Footnotes:

- We have enjoyed it
- We can do more - another improvement after the competition …
- We are able to train production ASR for a different location (semi-supervised)

CENTER FOR LANGUAGE AND SPEECH PROCESSING

UNIVERSITY OF WEST BOHEMIA

ASR overview

- KALDI-based ASR
- Deployment-ready single system

Overview

- Lexicon preparation
- Language modeling
- Additional data?
- Handling <UNK> and <FOREIGN> tokens

Lexicon preparation

Out of 2500 types in the training list, around 500 were typos.

We checked against CMUdict

- Fixed manually typos
- Generated french pronunciation for french words (cities) using espeak + manually created table IPA->ARPAbet
- Verified specific words do exist (ATC terminology, waypoints)
- Trained G2P for correct, words not present CMUdict (phonetisaurus)
- Added ‘huh’ pronunciations (from WSJ)
- Two possible <UNK>: unknown word ‘_’ and foreign word (or phrase) ‘@’

CENTER FOR LANGUAGE AND SPEECH PROCESSING

UNIVERSITY OF WEST BOHEMIA

Language Modeling

Used srilm toolbox

- 3-gram perplexity: 8.0
- 4-gram perplexity: 5.0 (MaxEnt LM, used for rescoring)

RNNLM didn't help

No other external data

Additional data available ?

Youtube channels (approx 100 hrs recordings)

LiveATC

- Fan-driven community page containing recordings of communication from various airports
- downloaded around 150k hours of recordings
- FR, CZE, SW, US, CAN accents

UWB corpus (proprietary corpus of approx 200 hrs of CZE accented ATC - IT-BLP data

Various sites with additional aux info: phraseology, spelling, aviation-safety, manuals, planecrashinfo, quora, skytalk, tailstrike

CENTER FOR LANGUAGE AND SPEECH PROCESSING

UNIVERSITY OF WEST BOHEMIA

Handling the <UNK>

- Typically, detecting UNKs is fairly hard task
- Normally, you'd see something like this in a lexicon
  - <UNK> <unk>
  - I.e. word '<UNK>' maps to a single unit '<unk>'
  - This way, the training procedure is able to use the sentence for training, but the model of '<UNK>' won't be very good
- For decoding, it is a better idea to replace the pronunciation of '<UNK>' by a phoneme graph
  - Either all probabilities constant
  - Or you can train a LM on alignment of the training data

```
if [ $stage -le 4 ] ; then
  utils/lang/make_unk_lm.sh data/local/dict exp/make_unk

  utils/prepare_lang.sh \
    --unk-fst exp/make_unk/unk_fst.txt --phone-symbol-table data/lang/phones.txt \
    data/local/dict "<UNK>" data/local/lang_test data/lang_test
```

Handling the <UNK>

First idea: map both ‘_’ and ‘@’ to <UNK>

Second idea from listening to audio: map ‘@’ to <FOREIGN> with pronunciations of French greetings.

```
<FOREIGN> b oh n jh uw r ah
<FOREIGN> b ah n sh uh r
<FOREIGN> b oh n jh uw r
<FOREIGN> b aa n
<FOREIGN> b oh n
<FOREIGN> jh uh r n ea
<FOREIGN> ow r ah v w aa
<FOREIGN> ow r ah v w aa r
<FOREIGN> oh r eh v uh aa r
```

```
if [ $stage -le 4 ] ; then
  utils/prepare_lang.sh \
    --unk-fst exp/make_unk/unk_fst.txt --phone-symbol-table data/lang/phones.txt \
    data/local/dict_foreign/ "<UNK>" data/local/lang_foreign_test data/lang_foreign_test

  utils/format_lm.sh \
    data/lang_foreign_test data/srilm_foreign/best_3gram.gz data/local/dict_foreign/lexicon.txt data/lang_foreign_test

  utils/build_const_arpa_lm.sh \
    data/srilm_foreign/best_4gram.gz data/lang_foreign_test data/lang_foreign_test_fg
```

Adding <FOREIGN>

- Hypothesis easy to test -- generate new lexicon and decoding graph, decode again
  - Make sure you use the '--phone-symbol-table' parameter for make_lang.sh
- Can we train? Remember not all <FOREIGN> can be salutations
  - Yes, we can
  - Utterances that fail the alignment will get removed automatically

- Too many utterances dropped? Add line

  <FOREIGN> <foreign>

  Into the lexicon (and into phone list)

  We tried this and for this case it made results worse

CENTER FOR LANGUAGE AND SPEECH PROCESSING

UNIVERSITY OF WEST BOHEMIA

Pronunciation probabilities

- Most lexicons do not specify which pronunciation variant is more probable.
- For some words, the silence is more probable than after other (this probability is not modeled by LM)
- We can use our alignments to estimate these probabilities
- In practice, the conditional silence probability seems to be more important

PRONUNCIATION AND SILENCE PROBABILITY MODELING FOR ASR

Guoguo Chen[1], *Hainan Xu*[1], *Minhua Wu*[1], *Daniel Povey*[1,2], *Sanjeev Khudanpur*[1,2]

[1]Center for Language and Speech Processing
[2]Human Language Technology Center of Excellence
The Johns Hopkins University, Baltimore, MD 21218, USA

guoguo@jhu.edu, hxu31@jhu.edu, mwu56@jhu.edu, dpovey@gmail.edu, khudanpur@jhu.edu

Pronunciation probabilities

```
steps/get_prons.sh --cmd "$train_cmd" data/train_nodup data/lang_nosp exp/tri3b

utils/dict_dir_add_pronprobs.sh --max-normalize true \
  data/local/dict_nosp exp/tri3b/pron_counts_nowb.txt exp/tri3b/sil_counts_nowb.txt \
  exp/tri3b/pron_bigram_counts_nowb.txt data/local/dict

utils/prepare_lang.sh data/local/dict "<unk>" data/local/lang data/lang
```

1. First, get the stats from the alignments (of the training data)
2. Create a new dict dir
3. Generate lang directory the usual way
4. Add G.fst and regenerate decoding graph (not shown)

CENTER FOR LANGUAGE AND SPEECH PROCESSING

UNIVERSITY OF WEST BOHEMIA

Data cleanup

- The transcribed data will often contain transcription errors, the segments are not correct, audio can be so noisy, that it causes harm using it…
- Idea: recognize using biased LM and use only those parts that were recognized correctly
- Used fairly often in kaldi egs
- Typically done before DNN training to get nice/correct alignments
- Script local/run_cleanup_segmentation.sh

```
# This does the actual data cleanup.
steps/cleanup/clean_and_segment_data.sh --stage $cleanup_stage \
  --nj $nj --cmd "$cmd" \
  $data $langdir $srcdir $dir $cleaned_data
```

Acoustic model

- Chain model (LF-MMI), factorized TDNN
- 12-layer, dim=1280, bottleneck=256, dropout
- Unconstrained egs
- Data cleanup (10 % of the data thrown away)
- Data augmentation: volume and speed (final system had 5-way, but performed only marginally better than “standard” 3-way)
- i-vectors (fairly small gain), tested two-pass i-vector estimation, again very tiny gain
- UNK = 4-gram phoneme loop
- Online decoder

CENTER FOR LANGUAGE AND SPEECH PROCESSING

UNIVERSITY OF WEST BOHEMIA

Internal Results (train split into 30+5(dev)+5(test))

Baseline 9.28

+	Cleanup	9.02
+	iVectors	8.98
+	Pronprobs	8.83
+	LM Rescoring	8.45
+	<FOREIGN>	7.69
+	Two-stage ivectors	~0.03 (not included)
+	7-way augmentation	~0.00

CENTER FOR LANGUAGE AND SPEECH PROCESSING

UNIVERSITY OF WEST BOHEMIA

ASR submissions details

Three different submissions

- Single system, TDNN -- driven by our philosophy, that the competing submission should reflect deployable solution -- real-time decoder, no (many)system combination, no (B)LSTM
- Three different submissions had the same AM two LMs
    - <FOREIGN> mapped to <UNK>
    - <FOREIGN> modelled as French phrases
- For Eval run, we have included dev and test, i.e. we trained on 40 hrs of speech. This gave 0.3 % improvement on leaderboard data.

CENTER FOR LANGUAGE AND SPEECH PROCESSING
UNIVERSITY OF WEST BOHEMIA

Call-sign detection - initial experiments

Reuse the semantic entity detection method from IT-BLP project

Many drawbacks in the challenge:

- Designed to work with ASR lattices
- Outputs the unified description of entity
- Uses expert-defined context-free grammars

Advantages not usable in the challenge:

- Allows to sum-up multiple ASR hypotheses with the same meaning
- Multiple output hypotheses with posterior scores

J. Švec, P. Ircing and L. Šmídl, "Semantic entity detection from multiple ASR hypotheses within the WFST framework," 2013 IEEE Workshop on Automatic Speech Recognition and Understanding, Olomouc, 2013, pp. 84-89. doi: 10.1109/ASRU.2013.6707710

CENTER FOR LANGUAGE AND SPEECH PROCESSING

UNIVERSITY OF WEST BOHEMIA

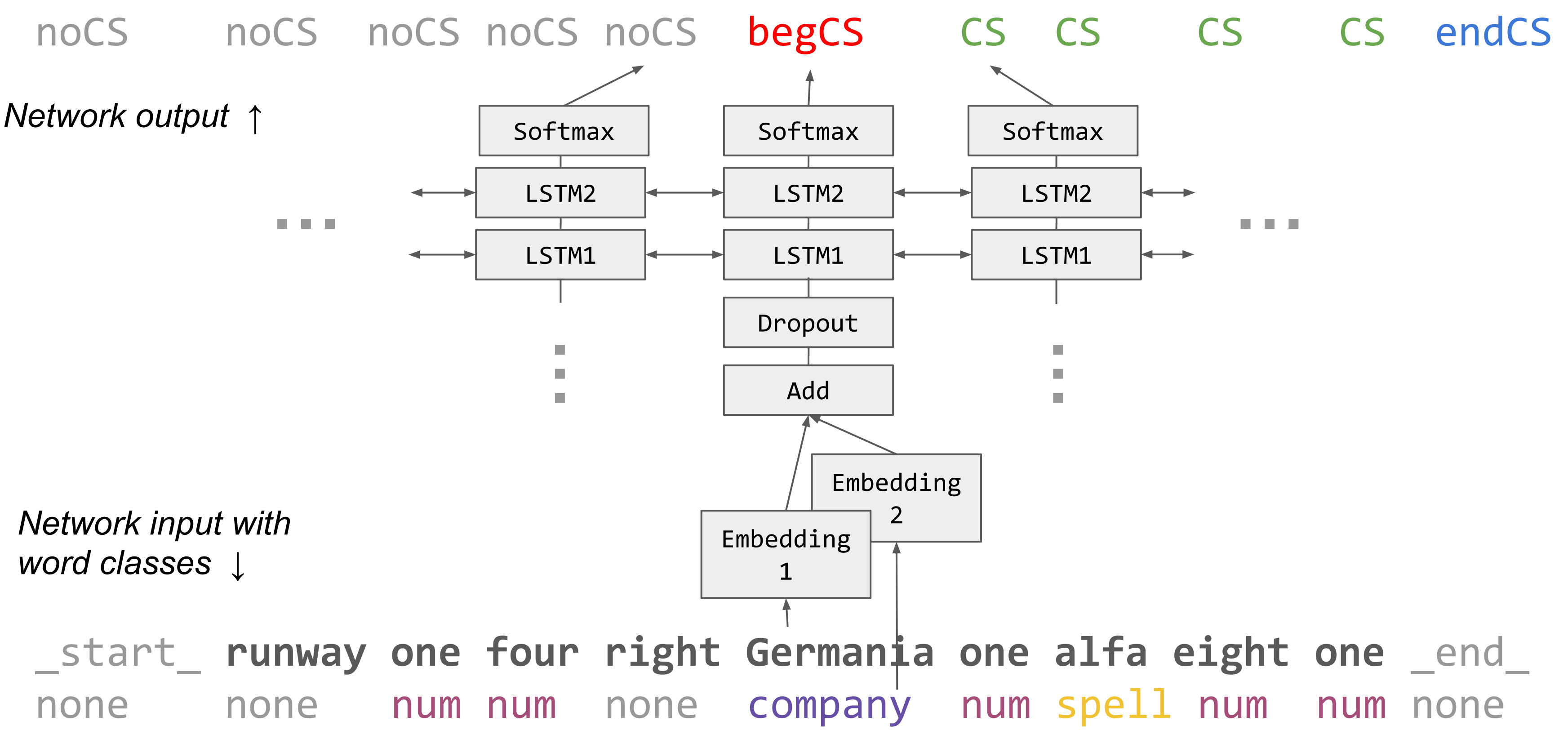
Call-sign detection - trainable model

2-layer bidirectional LSTM

- Training data
  - Recognized ASR hypothesis with ground truth callsign (alignment!)
  - Transcribed train partition
  - Recognized train partition
  - Recognized dev & test partitions
- LSTM tagging
  - Output classes: no CS, beginning of CS, middle of CS, end of CS
- Expert knowledge (word classes) by additional embedding layer
  - Company name, numbers, spelling alphabet
- Ensembling to average over different initializations of LSTM training

CENTER FOR LANGUAGE AND SPEECH PROCESSING
UNIVERSITY OF WEST BOHEMIA

Network architecture



Submissions details

We are using different LMW & WIP weights for ASR submission and CS detection

- Optimized on dev data
- Typically, the CS detection performs better with higher LMW

LSTM ensemble (3-5 averaged networks)

- to minimize the noise from different LSTM initializations

Improvement in WER ⇏ improvement in F1

- esp. for our train/dev/test split and leaderboard data

CENTER FOR LANGUAGE AND SPEECH PROCESSING

UNIVERSITY OF WEST BOHEMIA

Call-sign detection results

F1 metrics on leaderboard data

Semantic entity detection (expert-based)	0.7021
Initial experiment with LSTM (1 LSTM layer)	0.7984
Full-featured LSTM model	**0.8340**

CENTER FOR LANGUAGE AND SPEECH PROCESSING

UNIVERSITY OF WEST BOHEMIA

intent-bear ... WHO ARE WE?

University of West Bohemia - Department of Cybernetics (Pilsen, Czech Republic)

- Luboš Šmídl - smidl@kky.zcu.cz
- Jan Švec - honzas@kky.zcu.cz

Johns Hopkins University - Center for Language and Speech Processing (Baltimore, USA)

- Jan “Yenda” Trmal - jtrmal@gmail.com