Speech Translation

Jan Niehues
27/09/2019
jan.niehues@maastrichtuniversity.nl

Maastricht University

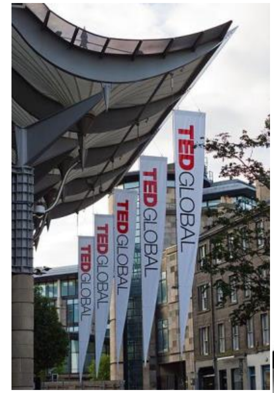
Use cases

- Presentations
  - Conferences/Lectures

- Videos
  - Internet: Youtube, Facebook, …
  - Television

- Every-day interactions
  - Tourist encounters, Medical care, Interactions with authorities
  - Telefon conversations

- Meetings

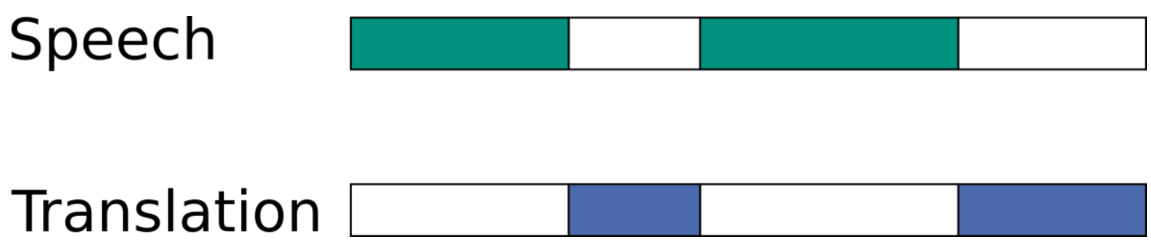


Overview

- Introduction
- Cascaded approach
  - Automatic speech recognition
  - Machine Translation
  - Segmentation and Punctuation
- End-to-End Speech Translation
  - Data conditions
- Challenges:
  - Simultaneous translation
  - Spontaneous speech
  - Speech output

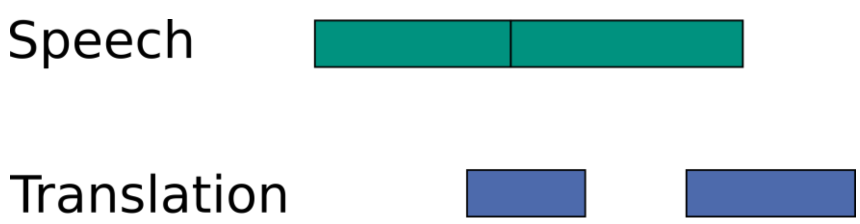
Different Application scenarios

- Sequence
  - Consecutive translation:
    - Speaker speaks a segment
    - Afterwards segment is translated
  - Characteristics:
    - Short Segments
    - Manual segmentation
    - Fixed dialog structure
      - No overlapping speech



Different Application scenarios

- Sequence
  - Simultaneous translation
    - Translation is provided while the speaker speaks
  - Characteristics:
    - Long segments
    - Automated segmentation needed
    - Flexible dialog structure



Different Application scenarios

- Sequence
- Number of speakers
  - Single speaker
    - E.g. Presentations
  - Multiple speaker
    - E.g. Meetings
    - Challenges:
      - Overlapping voice

Different Application scenarios

- Sequence
- Number of speakers
- Online/Offline systems
  - Offline: Translate audio in batch mode
    - E.g., movies
  - Online: Translate during production of speech
    - Real-time translations:
      - Translation as fast as speech input
    - Latency
      - Time that passes between speech and translation
      - Latency should be as minimal as possible

Different Application scenarios

- Sequence
- Number of speakers
- Online/Offline systems
- Presentation
  - Text
  - Audio
    - Additional TTS needed

History

- Speech translation systems for simple dialogs
  - Consecutive
  - Manual segmentation
  - Limited Domain
  - Events:
    - 1992
      - C-Star consortium: Development of several prototypes
    - 2004
      - IWSLT: First benchmark on speech translation

History

- Speech translation systems for simple dialogs

- Presentation translation
  - Simultaneous
  - Open Domain
  - Single speaker
  - Events:
    - 2005: First ever simultaneous translator presented at Carnegie Mellon University and University of Karlsruhe
    - 2010: IWSLT: First benchmark on TED talks
    - 2012: First service with simultaneous translation at Karlsruhe Institute of Technology
    - 2015: Skype Translator

History

- Speech translation systems for simple dialogs
- Presentation translation
- Meeting translation
  - Simultaneous
  - Multiple speaker

Recent Data Resources

- Fisher data [Post et al., 2013]
  - Languages: Spanish to English
  - Domain: Telephone conversation

- MuST-C Corpus [Di Gangi et al., 2019]
  - Languages: English to 8 European Languages
  - Domain: TED

- How2 [, 2018]
  - Languages: English to Portuguese, Multi-modal information
  - Domain: How-To Videos

- LIBRI-Trans [Kocabiyikoglu et al., 2018]MASS [Boito et al, 2019], STC [Shimizu et al., 2014], BSTC, ..

Overview

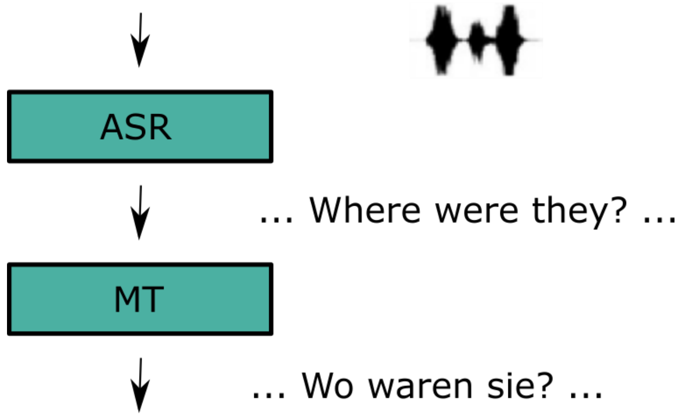
- Introduction
- Cascaded approach
  - Automatic speech recognition
  - Machine Translation
  - Segmentation and Punctuation
- End-to-End Speech Translation
  - Data conditions
- Challenges:
  - Simultaneous translation
  - Spontaneous speech
  - Speech output

Cascade Spoken Language Translation

- Serial combination of several models

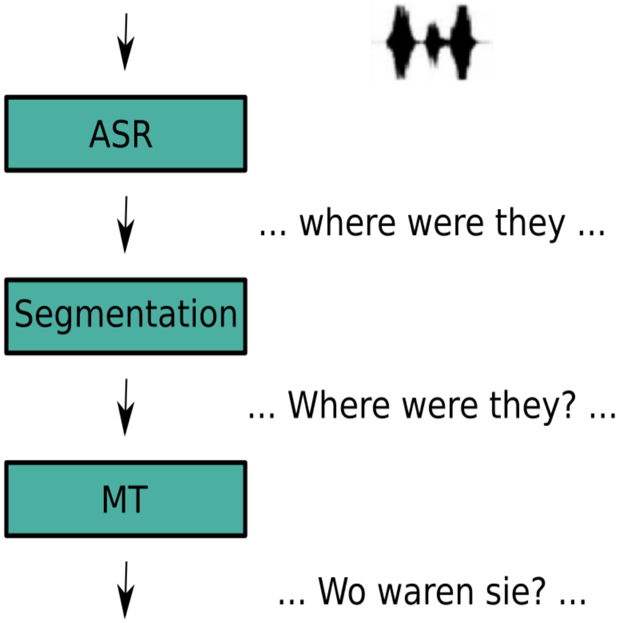
- ASR
  - Audio → Text

- Machine translation
  - Source language → target language



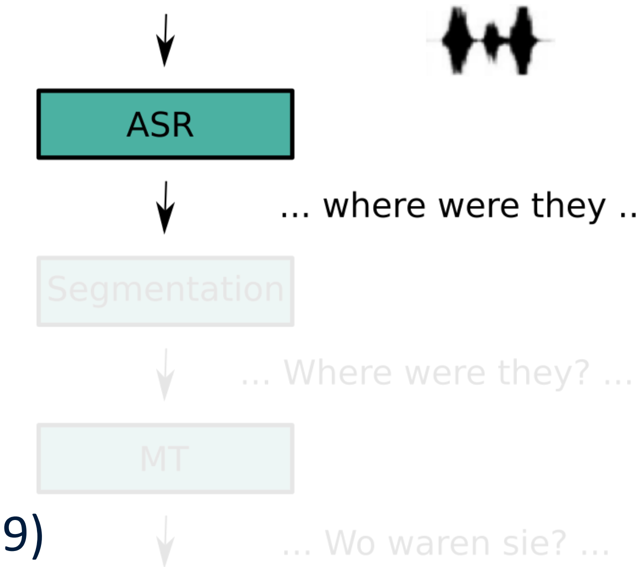
Cascade Spoken Language Translation

- Serial combination of several models
- ASR
  - Audio → Text
- Segmentation
  - Add case information
  - Add punctuation information
- Machine translation
  - Source language → target language



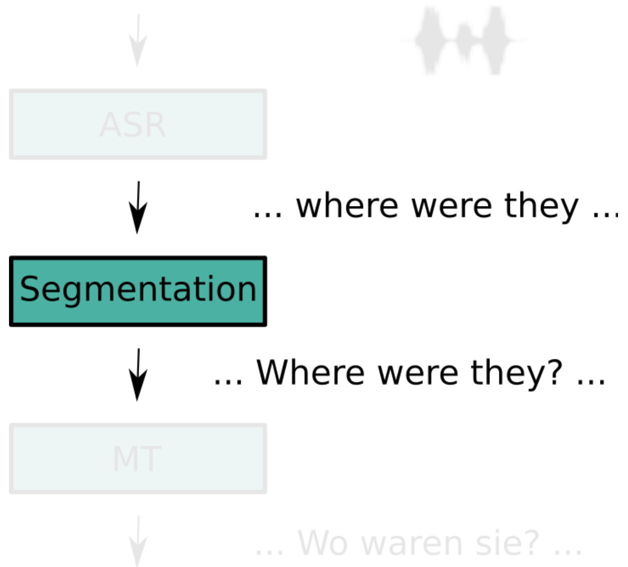
Automatic Speech Recognition

- HMM/DNN-based systems
  - Traditional ASR Systems
  - Still often state-of-the-art
- CTC-based Systems
  - LSTM to predict letters or blank symbol
  - CTC loss function
- Encoder-Decoder Systems
  - Deep networks necessary (Pham et al., 2019)



Segmentation

- Task:
  - Resegment text to sentence-like units
  - Insert punctuation marks
  - Often:
    - Correct casing of words



Segmentation

- Many applications:
  - Continuous audio stream
  - No punctuation in spoken language
- Automatic segmentation and punctuation needed
  - Readability
  - Semantic
    - “Let’s eat Grandpa !”
    - “Let’s eat, Grandpa !”
  - MT often operates at sentence level

ASR Output

- Example:

where
were they and what did they
talk about and now what was the topic of
the discussion as this
emotion of being angry came up now to be able
to answer these questions you will
also realize quite
quickly that this of course…

ASR Output

- Segmentation and punctuation are improve for readability

> **Where were they?**
> **And what did they talk about?**
> **And now what was the topic of the discussion, as this emotion of being angry came up?**
> **Now, to be able to answer all these questions, you will also realize quite quickly, that this of course…**

How do segmentation and punctuation affect machine translation?

- **Translation output** of German to English translation system
- ASR

> We see here is an example from the European Parliament, the European Parliament 20 languages
> And you try simultaneously by help human translator translators the
> Talk to each of the speaker in other languages to translate it is possible to build computers
> The similar to provide translation services

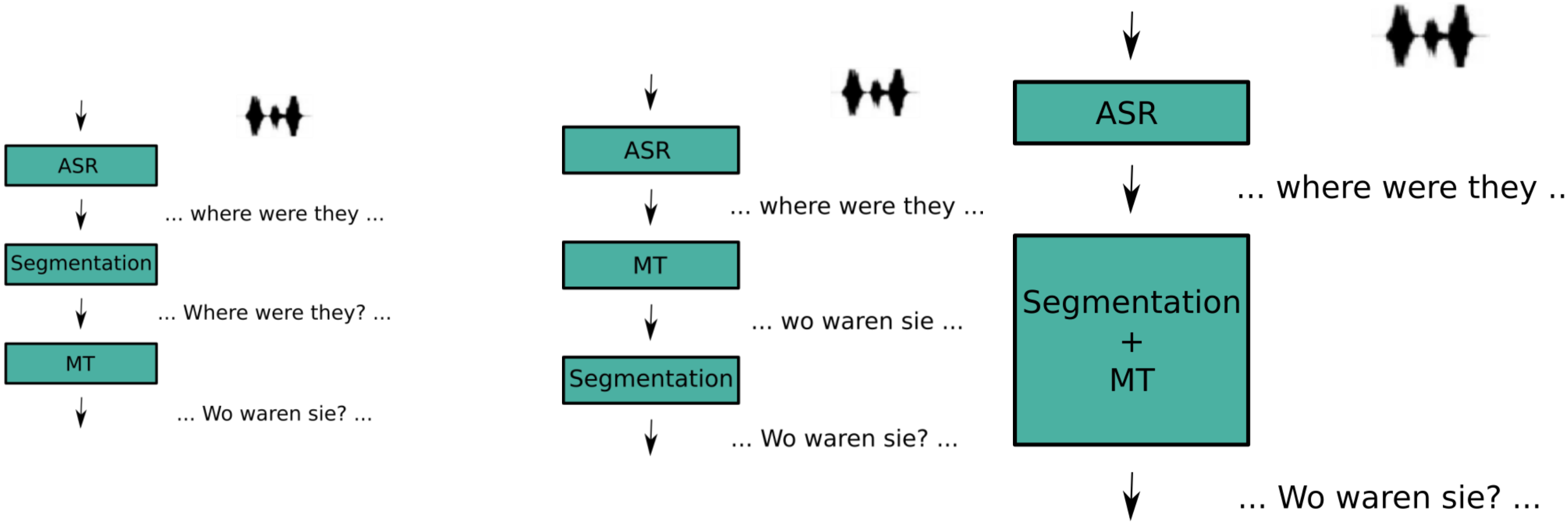
- ASR + correct segmentation and punctuation added manually

> We see here is an example from the European Parliament.
> The European Parliament 20 languages are spoken, and you try by help human translator to translate simultaneously translators the speeches of the speaker in each case in other languages.
> It is possible to build computers that are similar to provide translation services?

Segmentation and Punctuation

- Insertion of right punctuation gets difficult as the speech gets more disfluent
- Example:
  - "I (long pause) uh went to hair salon yesterday"
- Long pause can cause punctuation marks
  - "I."
  - "uh went to hair salon yesterday."
- For translation we need better segmentation and punctuation

Adding Punctuation



- Segmentation difficult in middle and right version
  - Peitz et al., 2011

LM and prosody based model

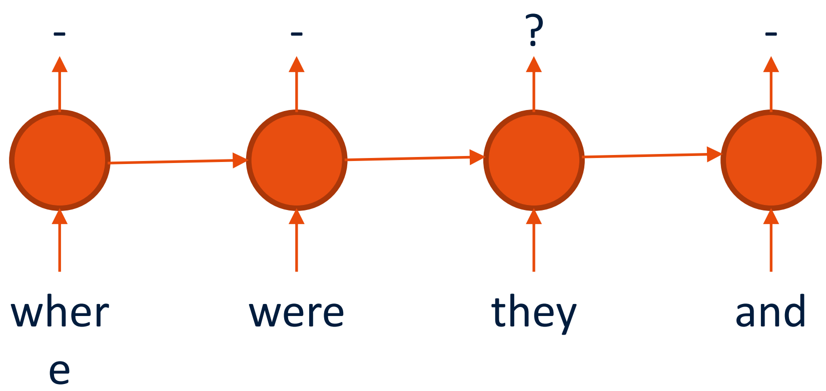
- Consider two prior words and two after the possible punctuation marks

- LM trained on punctuated text
  - Score without an inserted punctuation mark
    - P(Hello Sir how are)
  - Score with a comma
    - P(Hello Sir , how are)
  - Score with a full stop
    - P(Hello Sir . how are)
- Pause longer than n seconds then a new segment

- Fast

Sequence labeling

- Input:
  - Sequence of words
- Output:
  - Following punctuation mark
- Models:
  - CRF, HMM, LSTM, …



Monolingual translation system

- Input:
  - Text without punctuation
- Output:
  - Text with punctuation
- Models:
  - Phrase-based SMT, NMT, …

- Steps:
  - Generate training data
  - Train model
  - Apply model to input data
  - Insert segment boundaries after punctuation

Monolingual MT- Training data

- Parallel text:
  - Remove punctuation from monolingual source text

Where were they **And what did they talk about** **And now what was the topic of the discussion as this emotion of being angry came up** **Now to be able to answer all these questions you will also realize quite quickly that this of course…**
Where were they? **And what did they talk about?** **And now what was the topic of the discussion, as this emotion of being angry came up?** **Now, to be able to answer all these questions, you will also realize quite quickly, that this of course…**

Monolingual MT- Training data

- Parallel text:
  - Remove punctuation from monolingual source text
  - Randomly split text

where
were they and what did they
talk about and now what was the topic of
the discussion as this
emotion of being angry came up now to be able
to answer these questions you will

where
were they? and what did they
talk about? and now, what was the topic of
the discussion, as this
emotion of being angry came up? now, to be able
to answer all these questions, you will
also realize quite
quickly, that this of course

Monolingual MT- Testing

- Sliding window to observe words in longer, various contexts

der	bildet	die	sogenannte	konjunktive	Normalform	wir	haben
bildet	die	sogenannte	konjunktive	Normalform	wir	haben	gesehen
die	sogenannte	konjunktive	Normalform	wir	haben	gesehen	dass
sogenannte	konjunktive	Normalform	wir	haben	gesehen	dass	wir
konjunktive	Normalform	wir	haben	gesehen	dass	wir	diese
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

Monolingual MT- Testing

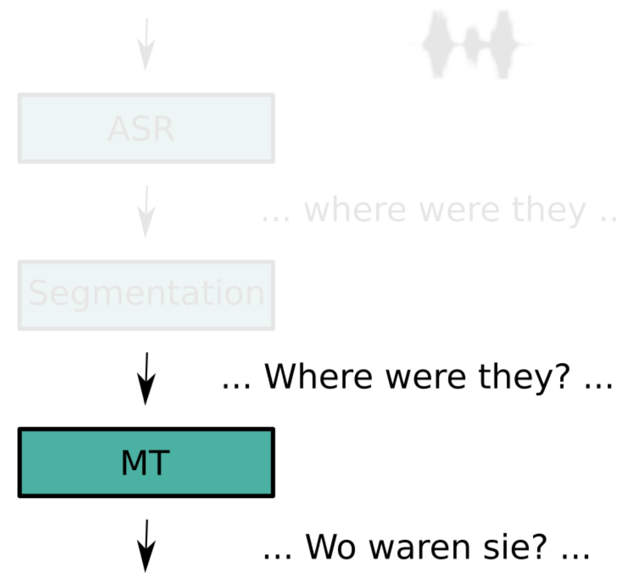
- Sliding window to observe words in longer, various contexts
  - Empirical threshold for inserting punctuation mark

der	bildet	die	sogenannte	konjunktive	**Normalform.**	Wir	haben
bildet	die	sogenannte	konjunktive	**Normalform.**	Wir	haben	**gesehen,**
die	sogenannte	konjunktive	**Normalform.**	Wir	haben	**gesehen,**	dass
sogenannte	konjunktive	**Normalform.**	Wir	haben	**gesehen,**	dass	wir
konjunktive	**Normalform.**	Wir	haben	**gesehen,**	dass	wir	diese
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

Machine translation

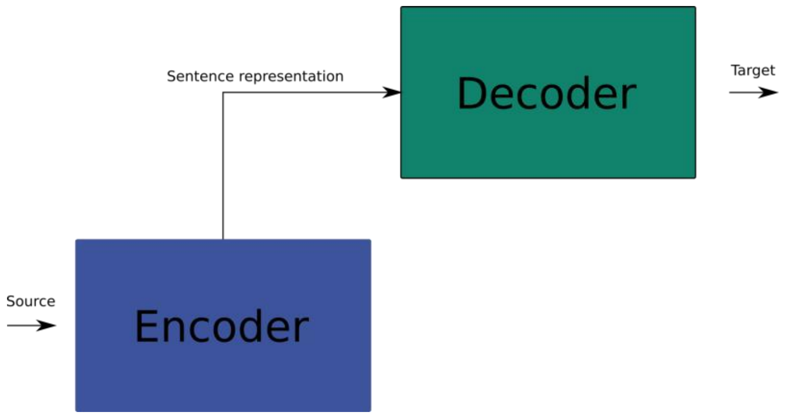
- Baseline
  - Sequence-to-Sequence based models

- Style in speech is different
  - Often adaptation to speech style
  - Continue training



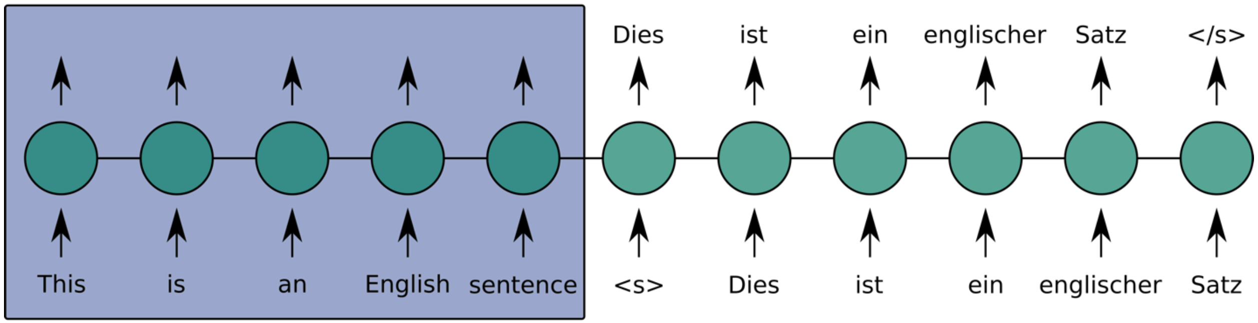
Sequence to Sequence model

- Predict words based on previous target words and source sentence
- Encoder
  - Read in source sentence
- Decoder
  - Generate target sentence word by word



Encoder

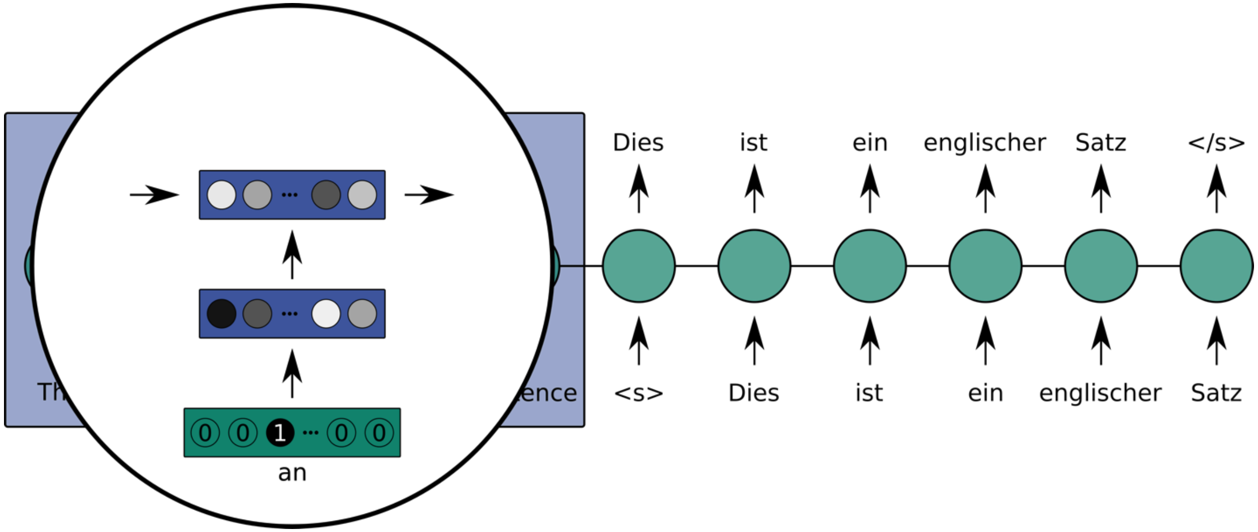
- Read in input
  - Represent content as hidden vector with fixed dimension
- LSTM-based model
- Fixed-size sentence representation



Encoder

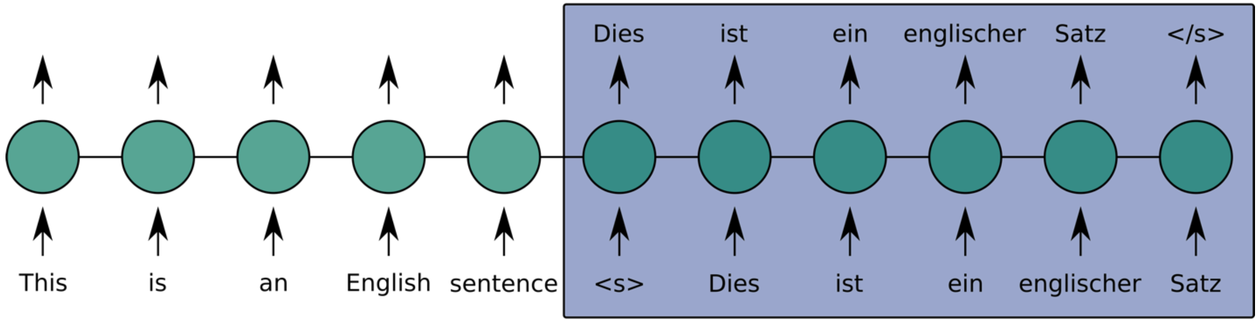
- Read in input
  - Represent content as hidden vector with fixed dimension
- LSTM-based model
- Fixed-size sentence representation

- Details:
  - 1 – hot encoding
  - Word embedding
  - RNN layer(s)



Decoder

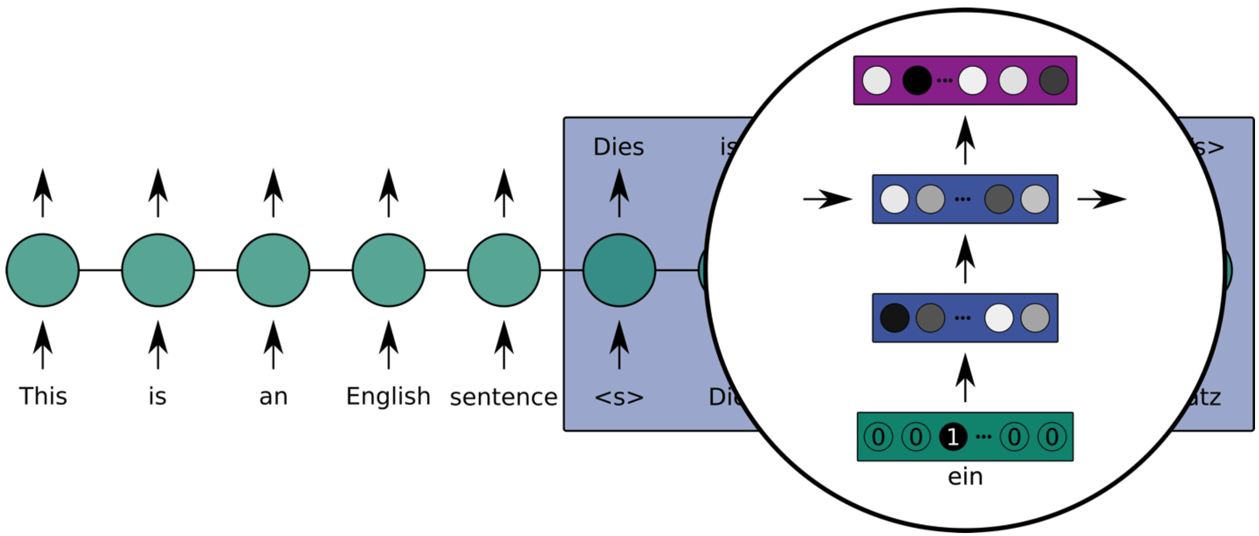
- Generate output
  - Use output of encoder as input
- LSTM-based model
- Input last target word



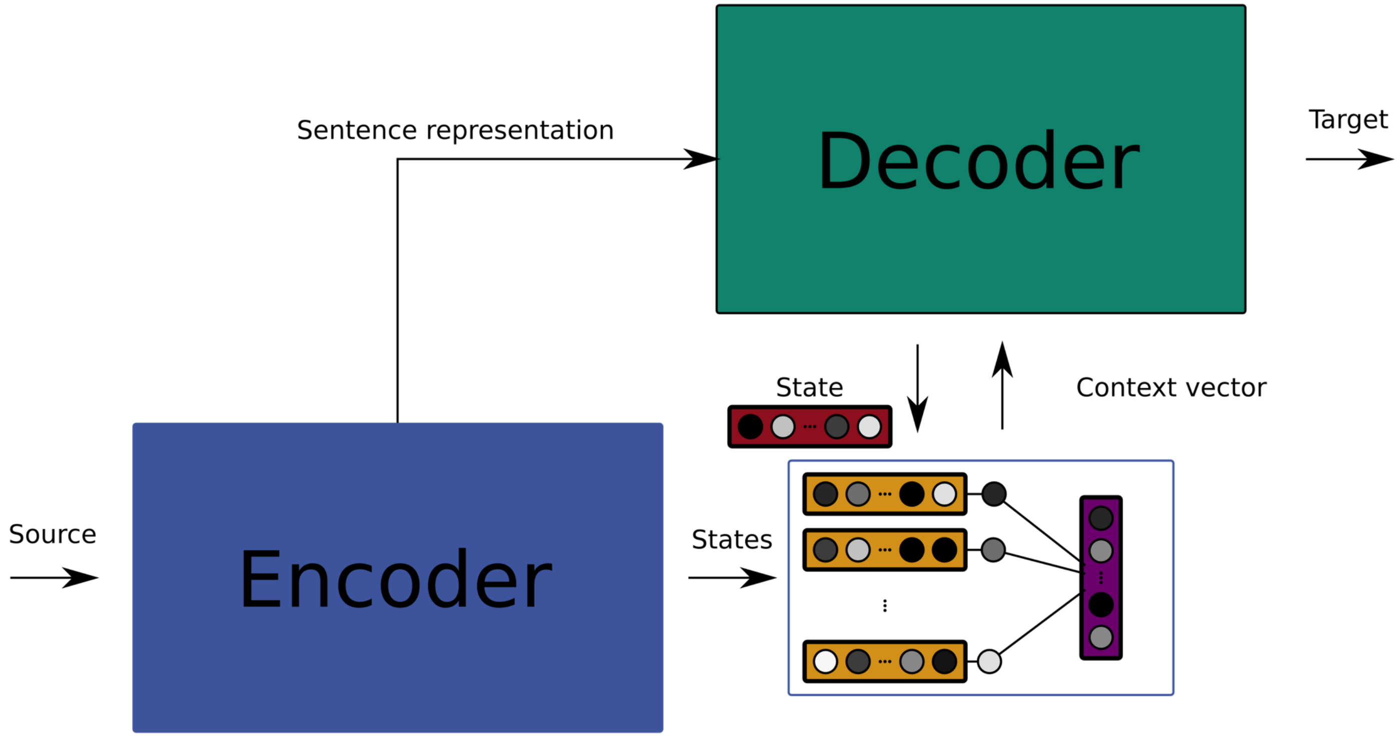
Decoder

- Generate output
  - Use output of encoder as input
- LSTM-based model
- Input last target word

- Details:
  - 1-hot representation
  - Word embedding
  - RNN layer(s)
  - Output layer

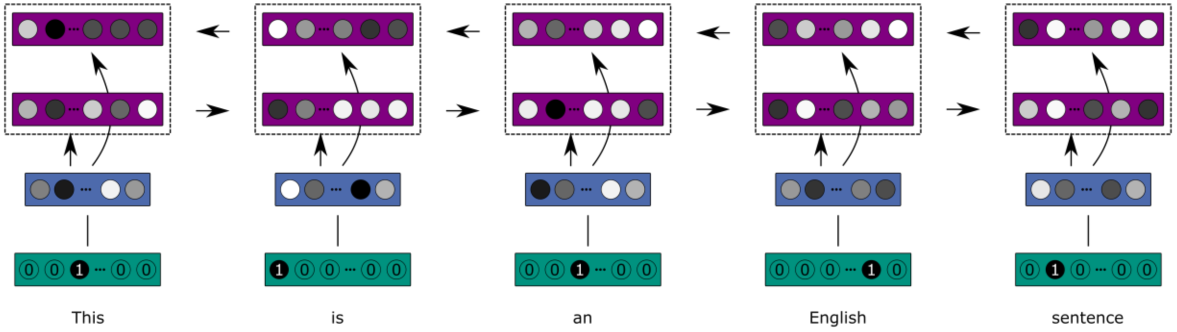


Attention-based NMT



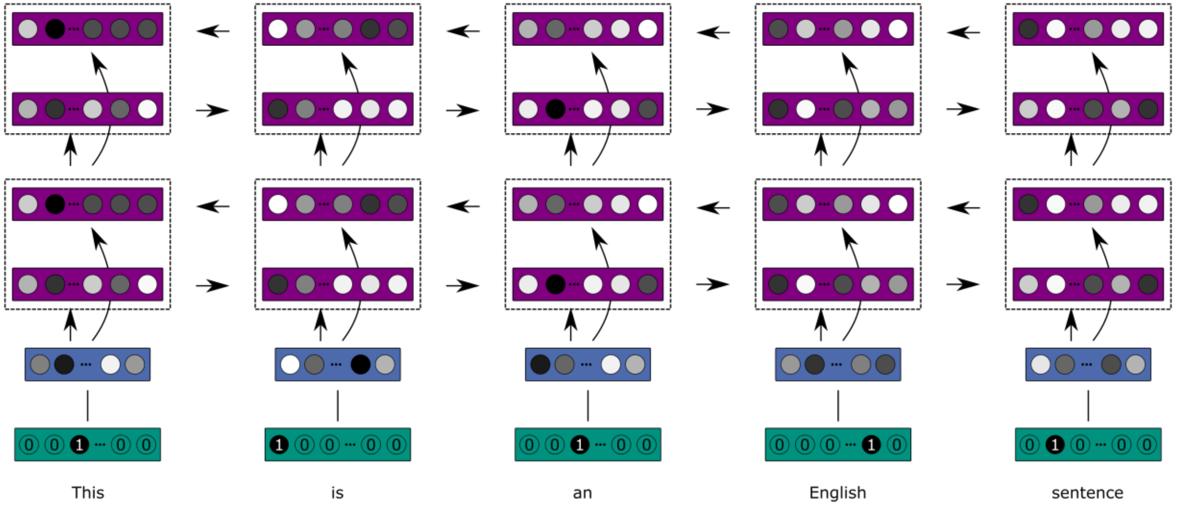
Advanced RNN

- Encoder only
  - Bidirectional RNN
  - Past and future context



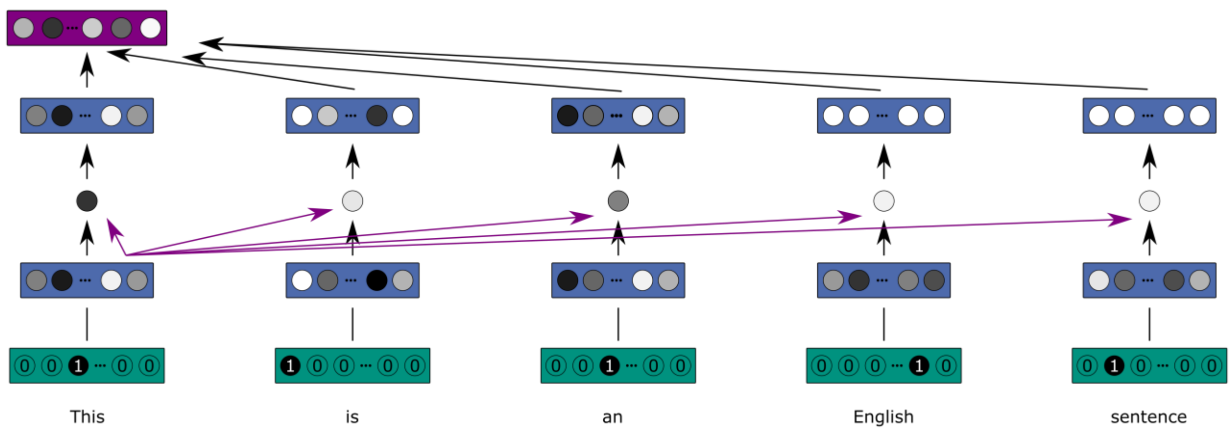
Advanced RNN

- Encoder only
  - Bidirectional RNN
  - Past and future context
- Encoder and Decoder
  - Multi-layer RNNs



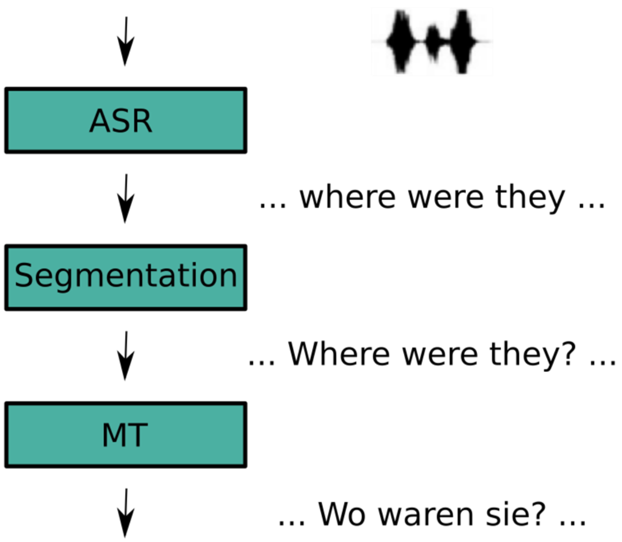
Transformer

- Self-attention
  - Replace RNNs by self attention networks
  - Calculate similarity between state and all other states
  - Calculate weighted sum



Cascade Spoken Language Translation

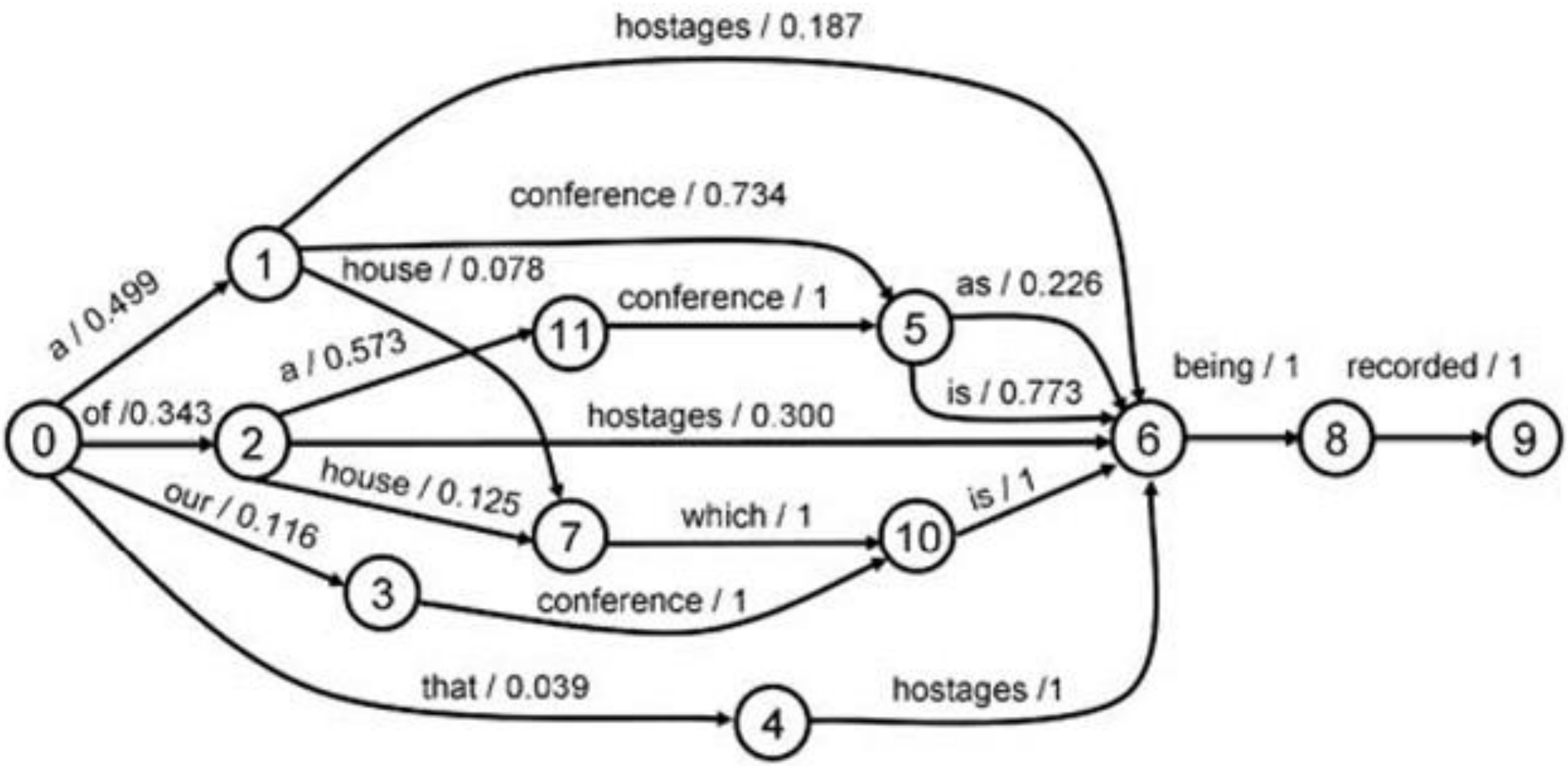
- Serial combination of several models
  - Automatic speech recognition (ASR)
  - Machine translation (MT)
  - Segmentation
- Advantages:
  - Data availability
  - Modular system
  - Easy incorporation of new ASR/MT developments



Cascaded SLT: Challenges

- Error propagation
  - Even the best components lead to errors
  - Solutions
    - Ignore
    - Represent different hypotheses
      - N-Best lists
      - Lattices [Saleem et al, 2005; Matusov et al, 2005]
    - Make MT robust to errors [Tsvetok et al. 2014; Lewis et al., 2015; Sperber et al, 2017]

ASR lattices



a conference is being recorded

Tight integration

- Find most probable translation for path in the lattice
  - Adapt SMT or NMT
- Use score to model confidence of ASR system
- Problems:
  - MT might translate easier sentence, not correct one

Robust MT

- Introduce errors to parallel training data
  - Artificial noise:
    - Randomly insert/remove/substitute words
  - Real noise:
    - Replace source text by ASR output
    - Challenge:
      - Alignment between audio and target text needed

Cascaded SLT: Challenges

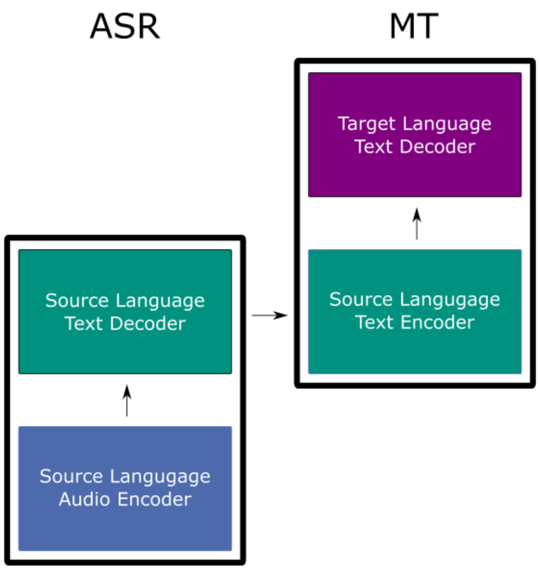
- Error propagation
- Separate optimization
- Script for source language is needed
- Computational complexity

Overview

- Introduction
- Cascaded approach
  - Automatic speech recognition
  - Machine Translation
  - Segmentation and Punctuation
- End-to-End Speech Translation
  - Data conditions
- Challenges:
  - Simultaneous translation
  - Spontaneous speech
  - Speech output

End-to-End SLT

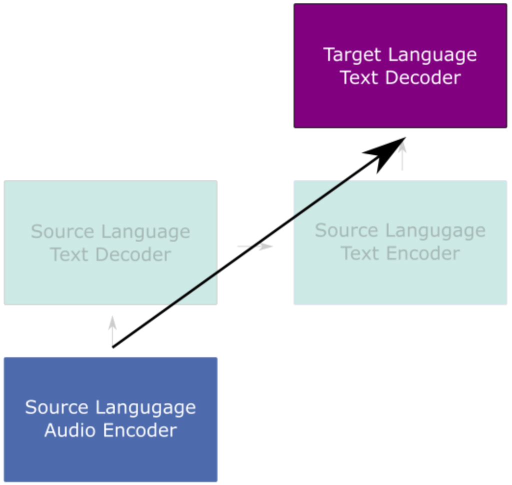
- Opportunity:
  - Sequence to Sequence models successfully applied to both tasks



End-to-End SLT

- Opportunity
- Directly learn mapping to target language text
  - [Duong et al., 2016;Berard et al., 2016; Weiss et al., 2017]

- IWSLT 2018 Evaluation:
  - Significant worse than cascaded models



End-to-End SLT

- Encoder:
  - Source side audio encoder
- Decoder:
  - Text-based decoder

- Comparison to ASR:
  - Decoder generated target language text

- Comparison to MT:
  - Source language audio instead of source language text

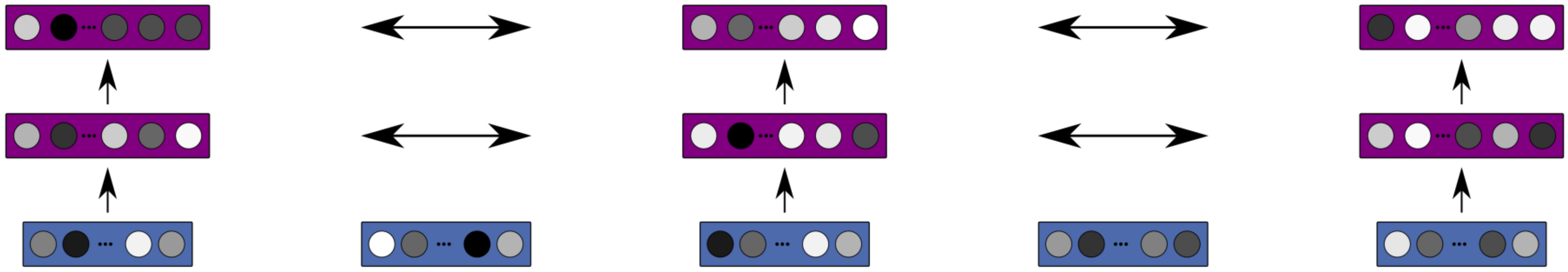
E2E SLT - Challenges

- Input is audio signal
  - Longer sequences difficult to handle for NNs
  - Dependencies in time and frequency dimension

- Data availability
  - Few end-to-end speech translation corpora available
  - Often considerably smaller than MT and ASR training data
- Complicated mapping between source and target sequence
  - Source transcript can be intermedia supervised signal

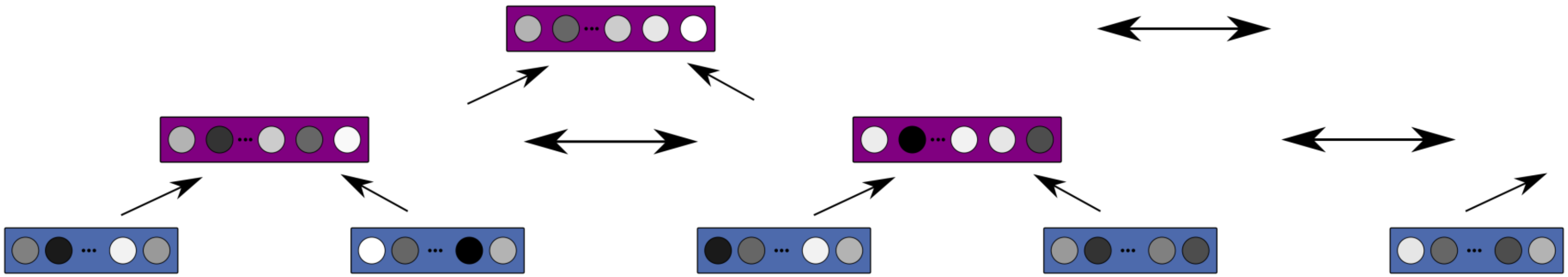
Input Sequence Length

- Large input sequence
  - Consecutive elements contain redundant information
    - Skip features vectors



Input Sequence Length

- Large input sequence
  - Consecutive elements contain redundant information
    - Skip features vectors
  - Pyramidal architecture (Chen et al., 2016)
    - Less states for higher layers



SLT Data

- Main challenge:
  - Few SLT corpora available

- Synthetic data:
  - Automatic generation by using TTS
  - [Berad et al, 2016; Kano et al, 2018]

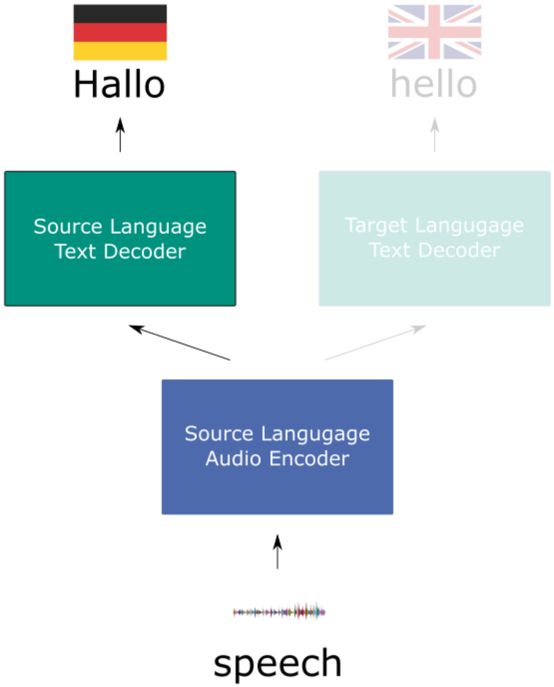
- Challenge:
  - Generalization from TTS output to real audio signal

Exploit other data sources

- Available data:
  - Speech data
  - Parallel MT data
- Exploit using multi-task learning
- Idea:
  - Share parts of the network
  - Train SLT system using speech or MT data

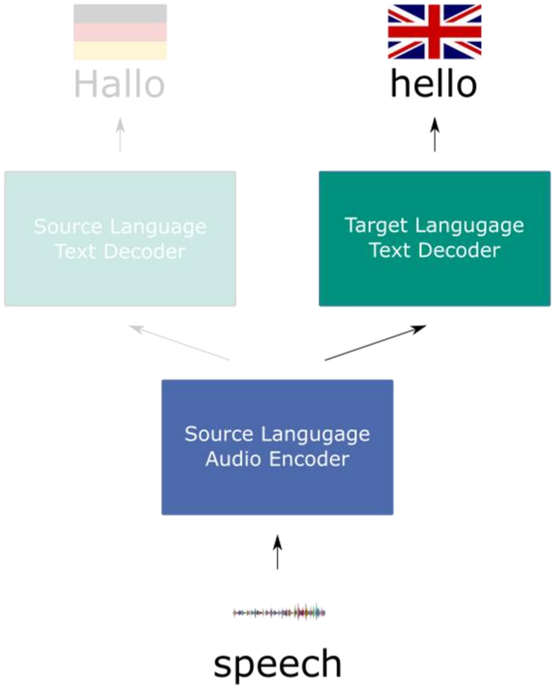
Multi-task learning

- Pre-training (Kano et al., 2018):
  - Train encoder on ASR task
  - Reuse on SLT task



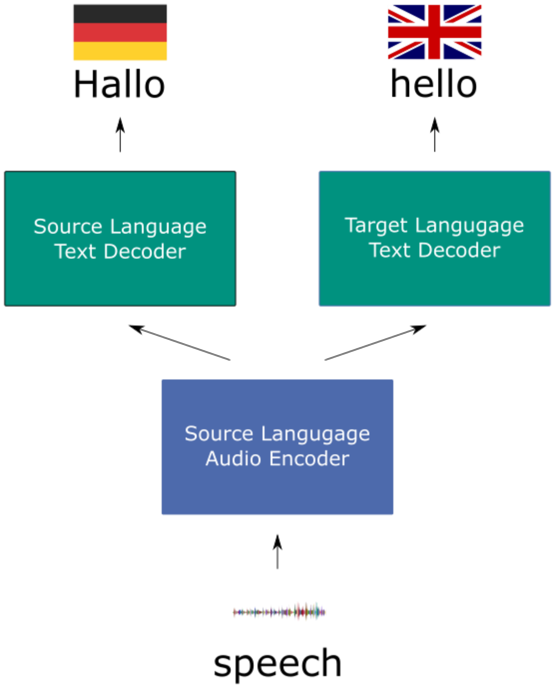
Multi-task learning

- Pre-training (Kano et al., 2018):
  - Train encoder on ASR task
  - Reuse on SLT task



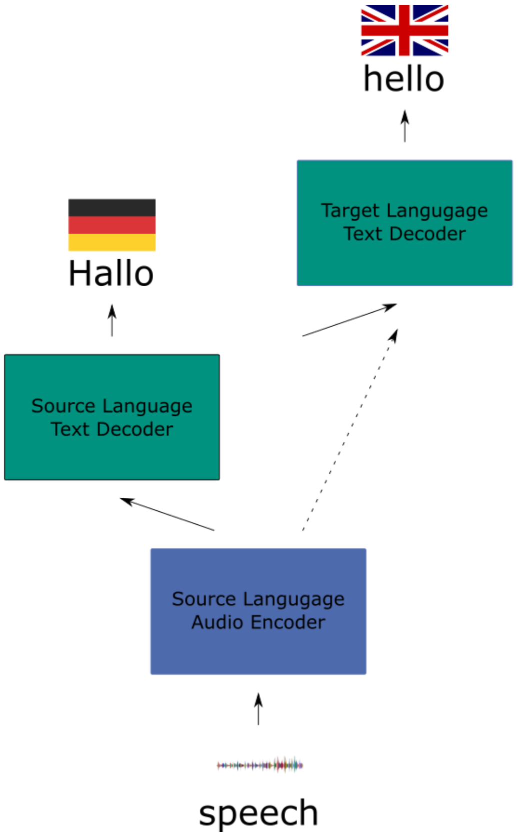
Multi-task learning

- Pre-training (Kano et al., 2018):
  - Train encoder on ASR task
  - Reuse on SLT task
- Multitasking (Weiss et al.,2017):
  - Train SLT and ASR jointly
- Challenge:
  - Data efficiency
  - How much gain from ASR/MT data?



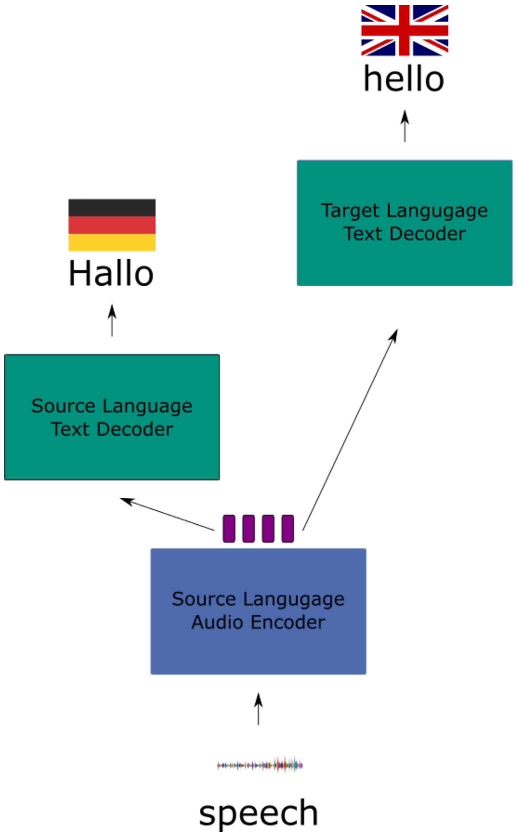
2-stage NN Model

- SLT needs to learn complicated mapping
  - Supervised intermediate signal available
- Stack different decoders
  - Attend to source language decoder hidden states
- Triangle version:
  - Attend to source audio and source text
  - [Anastasopoulos Chiang, 2018]
- Compared to cascade model:
  - No hard decision on words



Shared context vector

- Decoder is auto-regressive
  - Erroneous ASR words encoded in decoder states
- Attend to context vectors instead of decoder states
  - [Sperber et al, 2019]



Overview

- Introduction
- Cascaded approach
  - Automatic speech recognition
  - Machine Translation
  - Segmentation and Punctuation
- End-to-End Speech Translation
  - Data conditions
- Challenges:
  - Simultaneous translation
  - Spontaneous speech
  - Speech output

Challenges – Simultaneous Translation

- Generate translation while speaker speaks
- Tradeoff:
  - More context improves speech recognition and machine translation
    - Wait as long as possible
  - Low latency is important for user experience
    - Generate translation as early as possible
- Challenge:
  - Different word order in the language
    - SOV vs SVO

German	Ich	melde	mich	zur	CCMT	2019	an
Gloss	I	regester/ cancel	myself	to	CCMT	2019	
English	I	????					

Challenges – Simultaneous Translation

- Reasons for latency
  - Computation time → fast servers with multiple cores, parallelized computations, smaller, faster models..
  - Communication time → fast connection, low overhead between components
  - Required context length?

Challenges – Simultaneous Translation

- Approaches:
  - Learn optimal segmentation strategies
  - Stream decoding
    - Dynamically learn when to generate a translation
  - Re-translate
    - Update previous translation with better ones

Simultaneous Translation: Learn optimal segmentation strategies

- Idea:
  - Create segments that optimizing tradeoff between segment length and translation quality

- Advantages:
  - No changes to the NMT system
- Disadvantage:
  - Shorter context during translation

- E.g.:
  - Oda et al., 2014

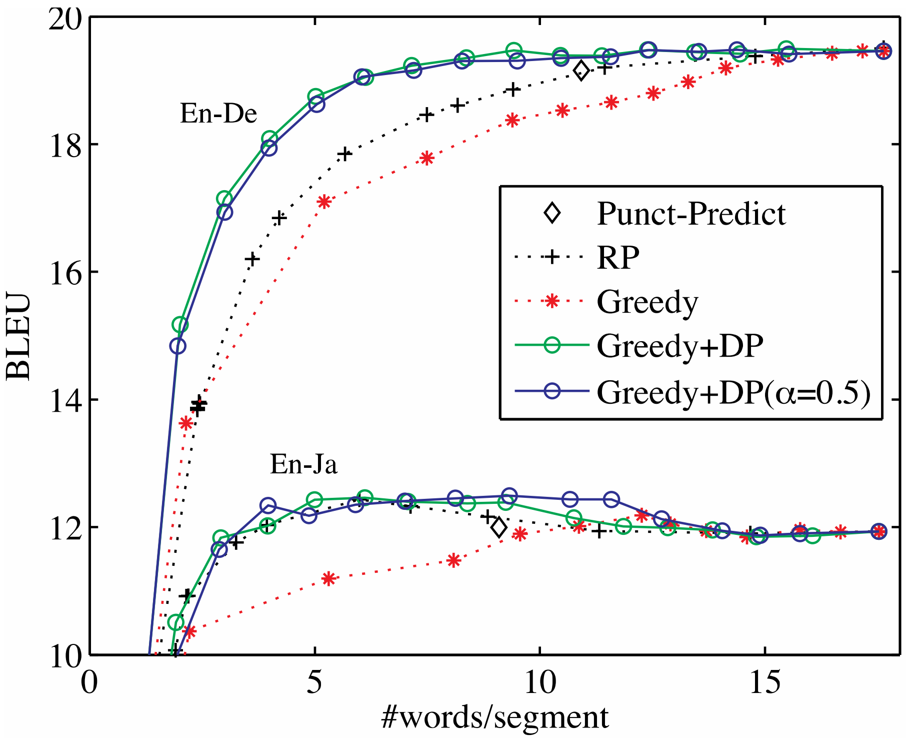
Example:

Ich melde mich

zur CCMT 2019 an

Simultaneous Translation: Learn optimal segmentation strategies

- Baseline:
  - Try to segmented into sentence
- Parameter for trade-off
  - Latency
  - Translation quality



Oda et al., 2014

Simultaneous Translation: Stream decoding

- Idea:
  - At each time step:
    - Decided to output word
    - Wait for additional input

- Methods:
  - Dynamic decision (Cho et al, 2016; Gu et al, 2017; Dalvi et al, 2018)
  - Fixed schedule (Ma et al, 2019)

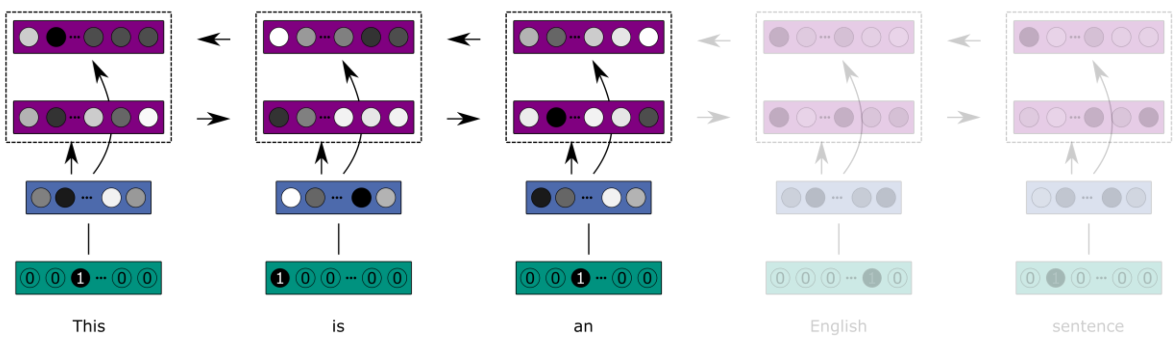
- Advantage:
  - Longer context into the past is available

- Disadvantage:
  - Major changes to the architecture
  - Balance between latency and quality

Stream decoding - Encoder

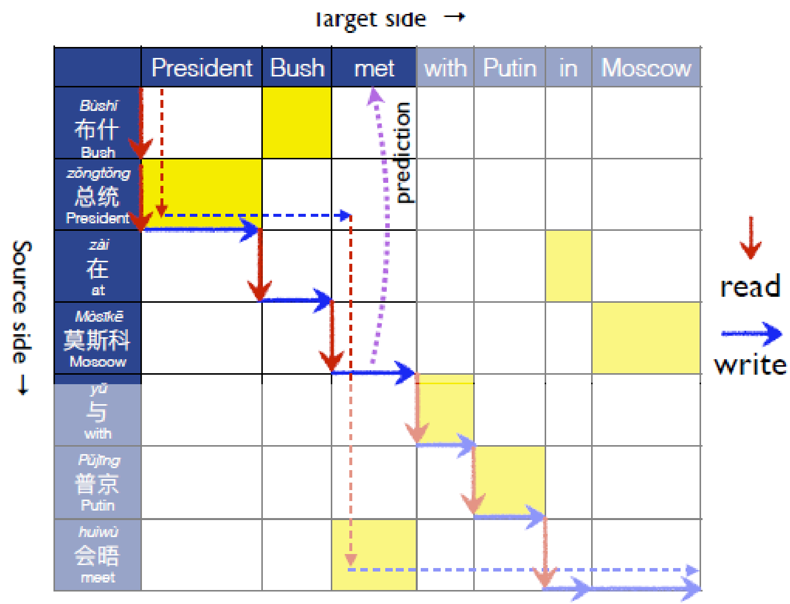
- Encoder:
  - No information of the future

  - LSTM:
    - Unidirectional
  - Attention:
    - Only attend to pervious states



Stream decoding - Decoder

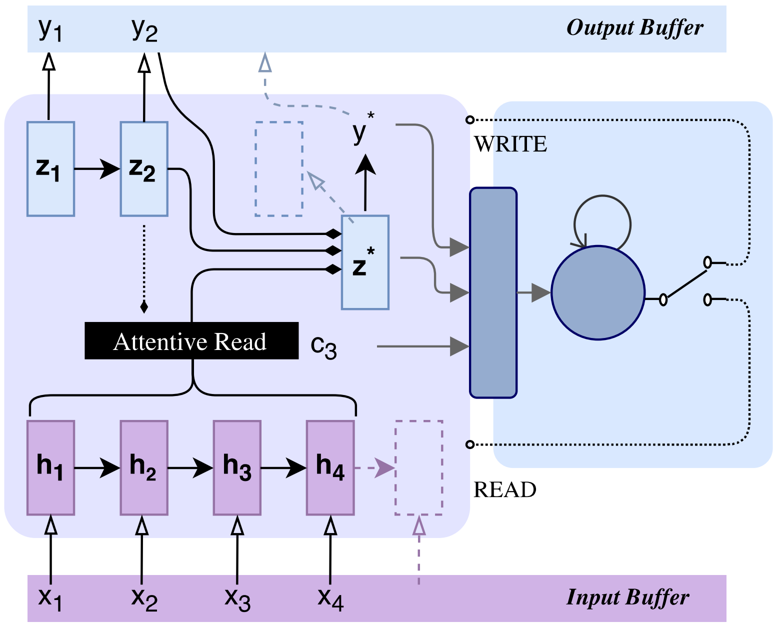
- Decoder:
  - Static delay
    - Wait-k policy
      - Ma et al., 2019



Ma et al., 2019

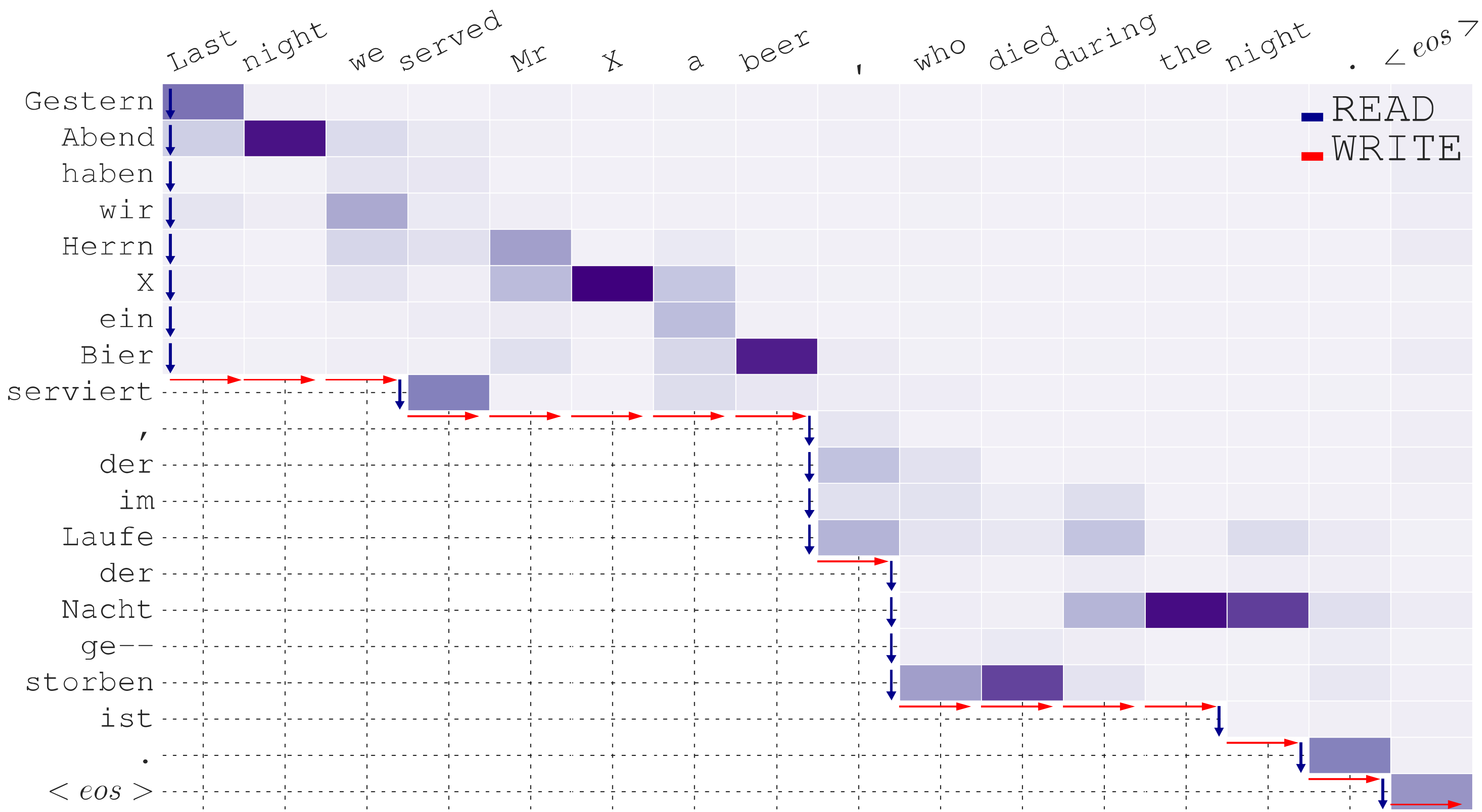
Stream decoding - Decoder

- Decoder:
  - Static delay
  - Dynamic delay:
    - At each time step:
      - Decided to output word
      - Wait for additional input
    - e.g. Train agent by reinforcement learning
      - Cho et al, 2016; Gu et al, 2017; Dalvi et al, 2018



Gu et al., 2017

Jointly predicting Segments and Translation



Gu et al., 2017

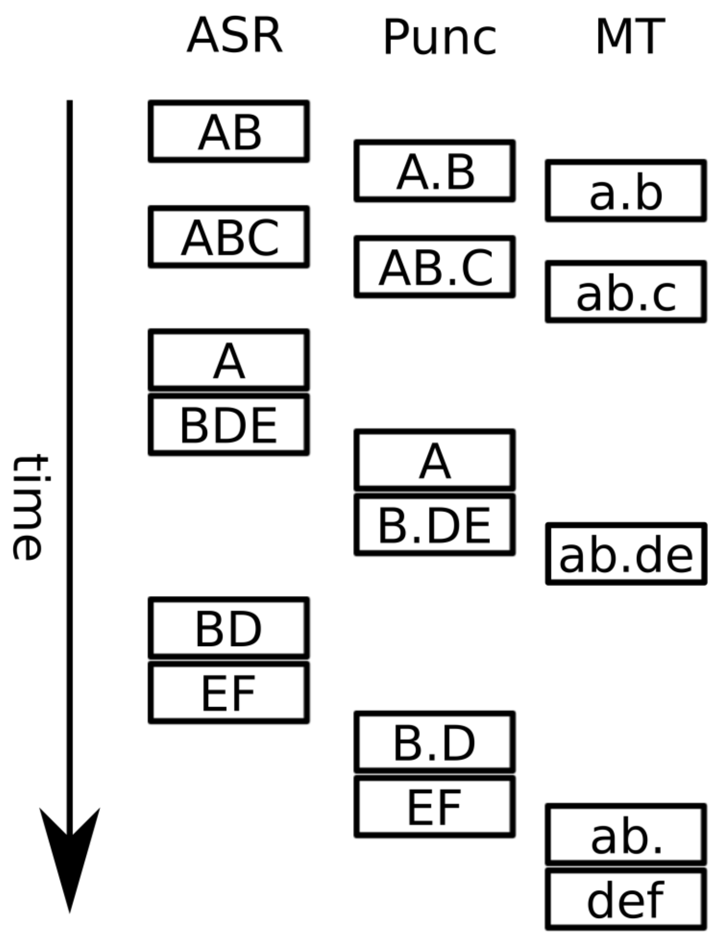
Stream decoding - Decoder

- Decoder:
  - Static delay
  - Dynamic delay
  - Beam search:
    - Adjust beam according to outputted words

Simultaneous Translation: Re-translation

- Idea:
  - Directly output first hypothesis (low latency)
  - If more context is available
    - Update with better hypothesis (high quality)
  - Not only for MT, but for all components [Niehues et al, 2016]
  - Example:
    - Ich melde mich → I register
    - Ich melde mich von der Klausur ab → I withdraw form the exam
- Advantages:
  - Low latency and high quality
- Disadvantages:
  - Bad user experience if there are many updates
  - High computation cost

Update Protocol



- Difficulty:
  - Also input gets updated
- Message goes through the 3 components
- Hypothesis constantly getting updated

Results

- En→Fr
  - 7.5 average seconds → 1.8 seconds for initial output, 3.3 seconds for the final output
- De→En
  - 8.6 average seconds → 2 seconds for initial output, 5.3 seconds for the final output
  - Reordering
- Analysis (subset)

n	1	2	5	10	Full sentence	Update
Latency(s)	5.3	5.4	6.0	7.3	7.9	6.0
BLEU	8.5	9.3	10.2	11.2	11.4	11.4

  - Partial sentences (n words)
  - Same latency as n=5 system
  - Outperforms the same latency system by 1.2 BLEU

Challenges for NMT

- NMT will always generate full sentences

Input	Output
now,	ahora ,
now, I should	ahora debería , debería , debería .
now, I should men	ahora debería hombres hombres .
now, I should mention that this	ahora debería mencionar esto .

Challenges for NMT

- NMT will always generate full sentences
- Train also on partial sentences

Input	Output
now,	ahora ,
now, I should	ahora debería
now, I should men	ahora debería.
now, I should mention that this	ahora , debo mencionarlo .

Challenges – Spontaneous speech

- We are speaking spontaneously usually in our lives
  - Except for formal speeches, talk,…
- Almost all of speech in normal situations
- Speaker is not reading scripts
- Natural, relaxed
- Daily life
- Meetings, phone call
  - Multiple speakers

Characteristics of spontaneous speech

- Frequent use of filler words
  - "uh", "uhm", "hmm"
  - "ja", "well"

- (rough) Repetition of phrases/words
  - "I mean, I mean I saw him there"
  - "there is, there was a cat"
  - "I would like to have a ticket to Denver, no, to Houston"

- Change of idea about what/how to speak
  - "We have here, uh, these fossils were discovered in Argentina…"
  - "How can you do that without, oh, what time is it now?"

Disfluency

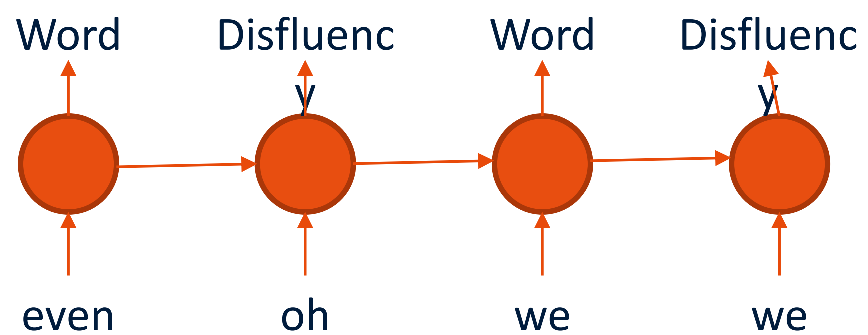
- Why is it so difficult?
  - Rough copies
    - The communication between man and machine, which we **customarily traditionally** always see, is the...
  - Some filler words, which can be filler, but sometimes not
    - “ja” in German
    - “well” in English
      - “we can’t even well we’re not even…”
      - “You did it very well”
  - Nearly no training data
  - ASR output may contain errors
  - Dangerous to remove to much

Challenges – Spontaneous speech

- Speech often spontaneous
  - Disfluencies
- Cascaded approach
  - Special model to generate clean text
  - E.g., as sequence labeling task [Cho et al, 2014]
- End to End:
  - Jointly learn to translate and remove speech disfluencies [Salesky et al, 2019]
  - Challenge:
    - Data resources

Approaches

- Sequence labeling
  - Input: words
  - Output: Labels
- Difficulties:
  - No word changes possible

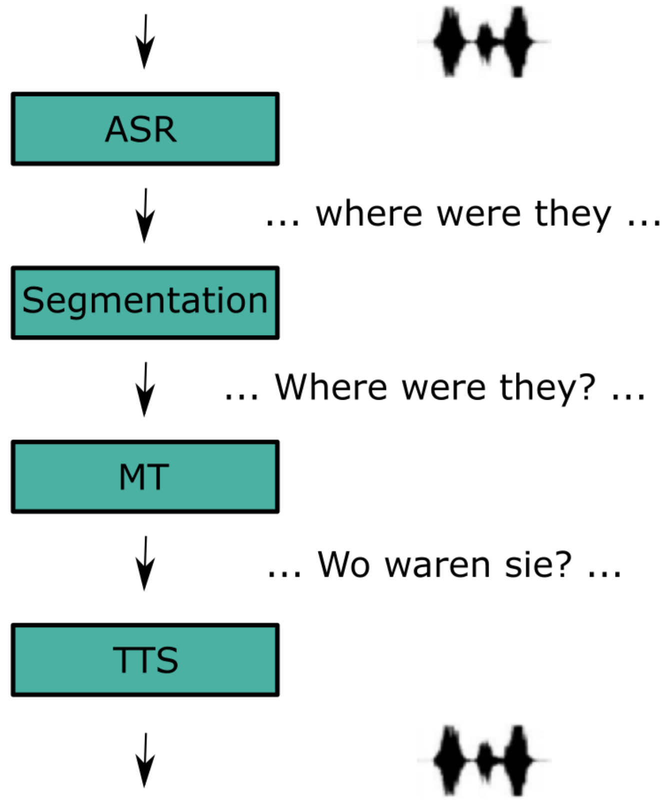


Speech output

- Until now:
  - Presentation as subtitles
- Human interpreter:
  - Speech output
- Challenges:
  - Delivery to audience
  - Error propagation
  - Combination with original voice

Baseline - Cascaded

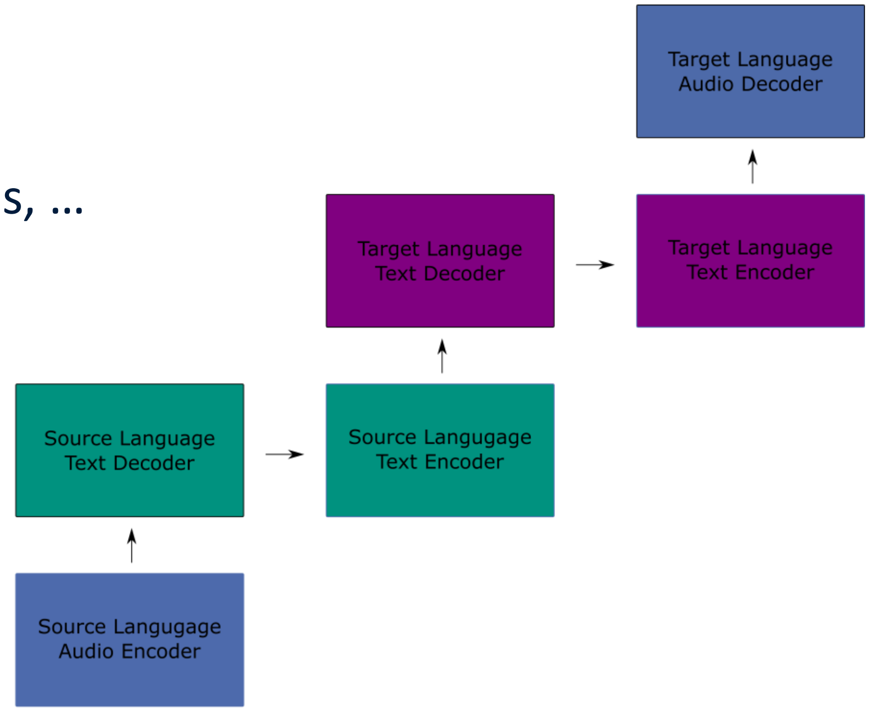
- Combination
  - Automatic speech recognition
  - Machine translation
  - Text-to-Speech



End2End models

- Jointly train ASR, MT and TTS
- Opportunities:
  - Retaining paralinguistic and non-linguistic information
    - Maintain source speaker voice
    - Emotion
    - Prosody

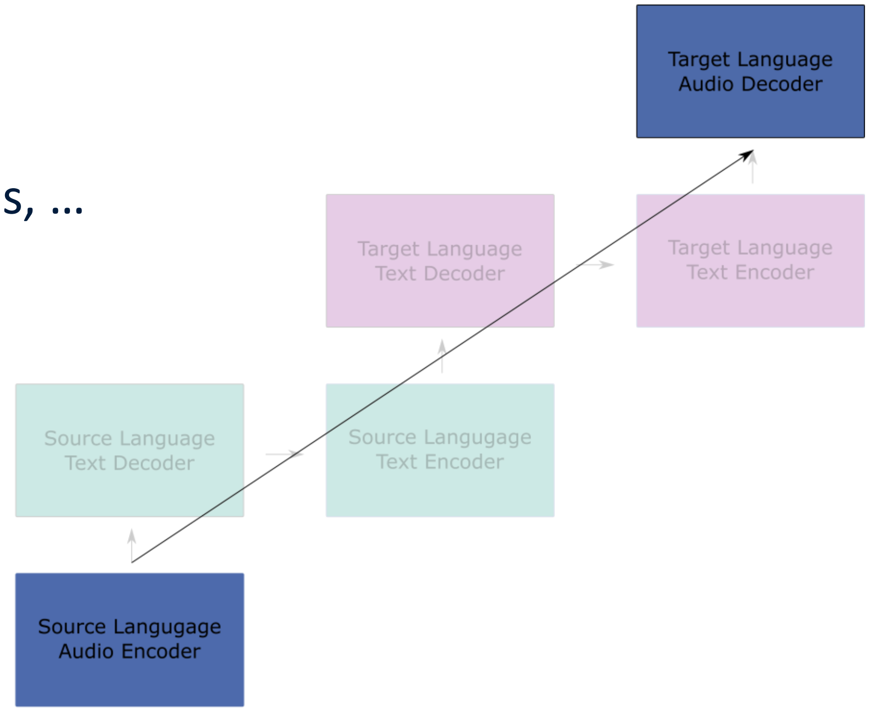
  - Fluent pronunciations of names, …



End2End models

- Jointly train ASR, MT and TTS
- Opportunities:
  - Retaining paralinguistic and non-linguistic information
    - Maintain source speaker voice
    - Emotion
    - Prosody

  - Fluent pronunciations of names, …

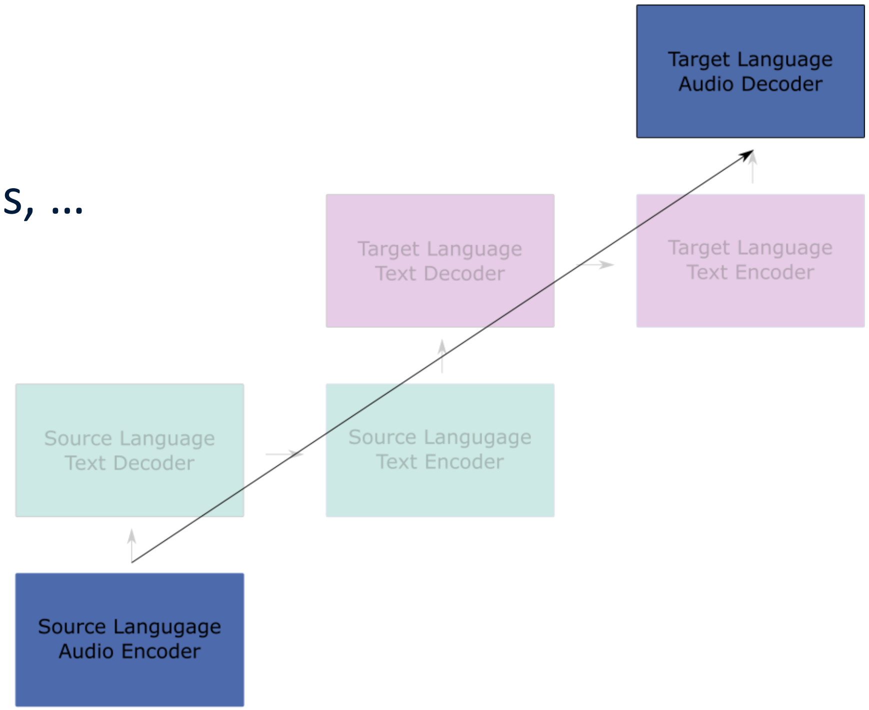


End2End models

- Jointly train ASR, MT and TTS
- Opportunities:
  - Retaining paralinguistic and non-linguistic information
    - Maintain source speaker voice
    - Emotion
    - Prosody

  - Fluent pronunciations of names, …

- First approach:
  - Jia et al, 2019



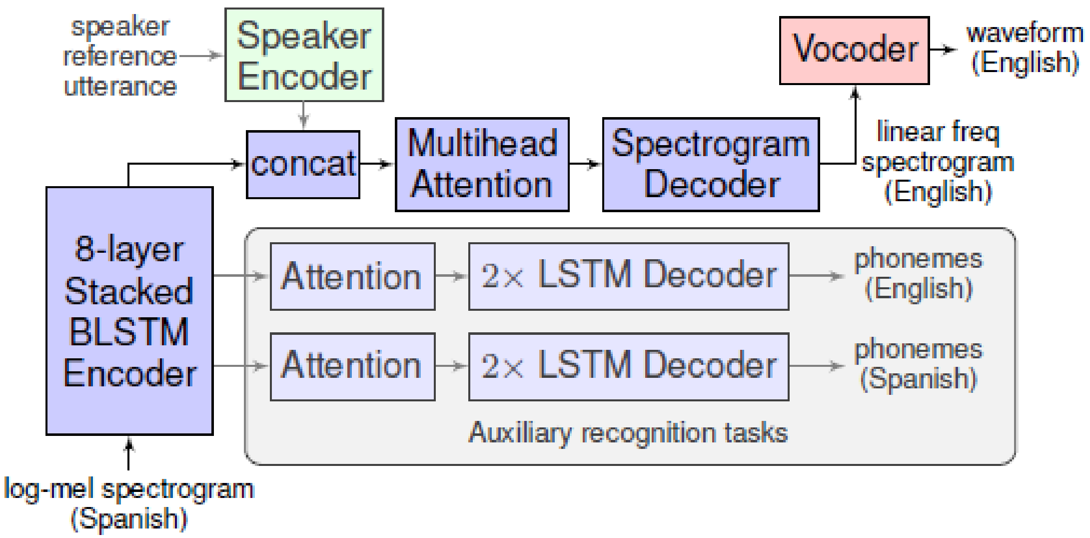
Audio decoder

- Two step approach:
  - Generate Spectrograms using sequence to sequence model
  - Vocoder
    - Spectogram -> wave form

- State-of-the-art approach for TTS
  - E.g. Tacotron2

Multi-tasking

- Essential for good quality
- Two additional task:
  - ASR
  - MT
  - Additional task use intermediate representation of the encoder



Jia et al, 2019

Evaluation

- Manual:
  - MOS evaluations
    - 5-point subjective listening test
    - Expected to be independent from translation quality
      - But translation might lead to “not understandable” output
- Automatic:
  - Run ASR separate ASR on speech output
  - Calculate BLEU score

Summary

- Speech translation adds additional difficulties
  - Segmentation
  - Disfluencies
  - Simultaneous translations

- Cascade models often still state of the art

- Significant improvements in end-to-end models

Future research directions

- Simultaneous E2E Speech Translation
  - Segmentation
  - Stream decoding
- Different data conditions
  - Multilingual models
  - Low/Zero resource models
- Prosody
- Manual interaction

Test it

- https://github.com/isl-mt/SLT.KIT

- Toolkit to build SLT system
  - MT:
    - RNN, Transformer
  - ASR:
    - CTC,RNN, Transformer
  - End2End:
    - RNN, Transformer
- Data from IWSLT available
  - Systems for How2

Maastricht University