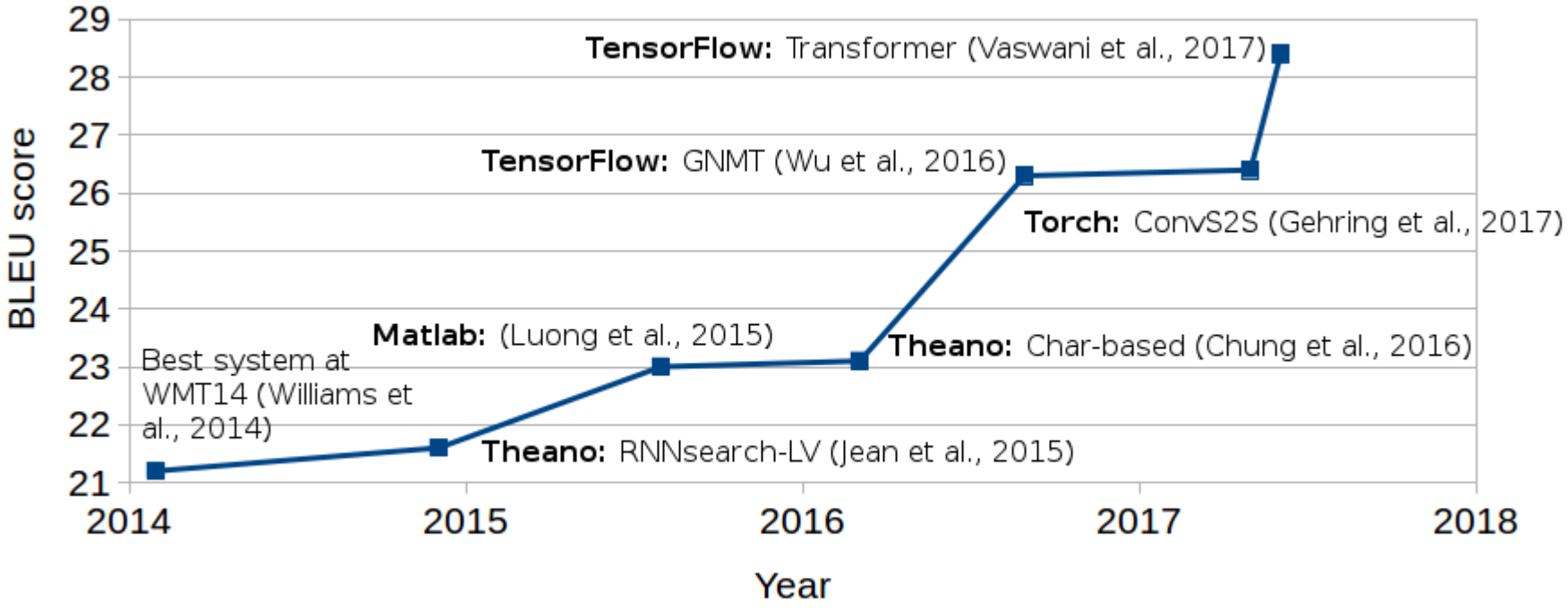
Why not be Versatile? Applications of the SGNMT Decoder for Machine Translation

Felix Stahlberg, Danielle Saunders, Gonzalo Iglesias, and Bill Byrne

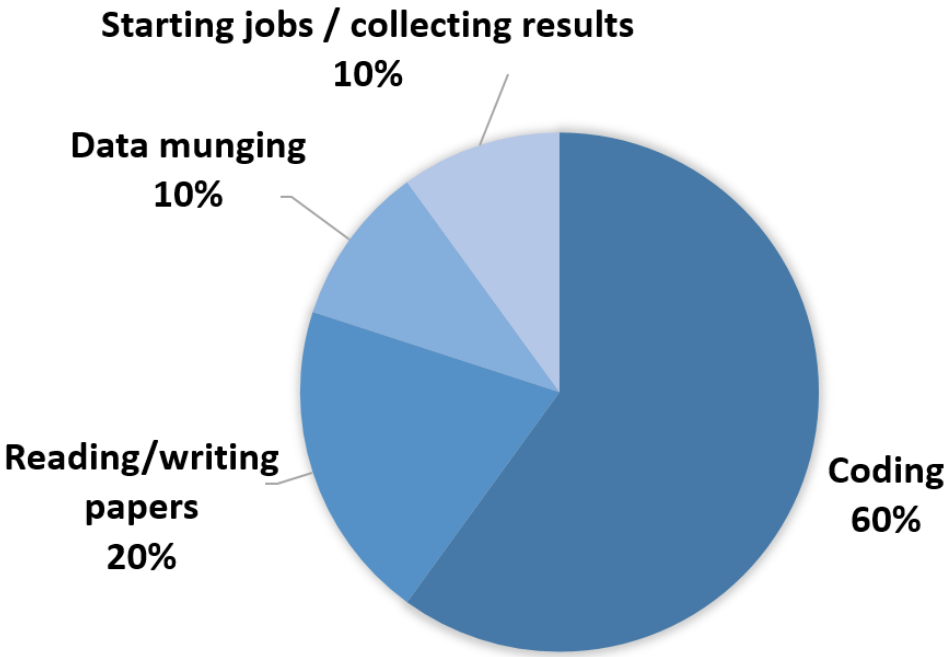
Department of Engineering

Motivation (1): Rapid progress in MT



- **Industry**: Rapid prototyping of new research avenues
- **Teaching**: Identifying suitable material in a quickly changing body of research
- **Research**: Keeping setups up-to-date with the latest models

Motivation (2): Coding is time consuming



- Implementation time is often far more valuable than computation time (for a PhD student).
- **Technical debt** (Sculley et a., 2014) is a major challenge in machine learning

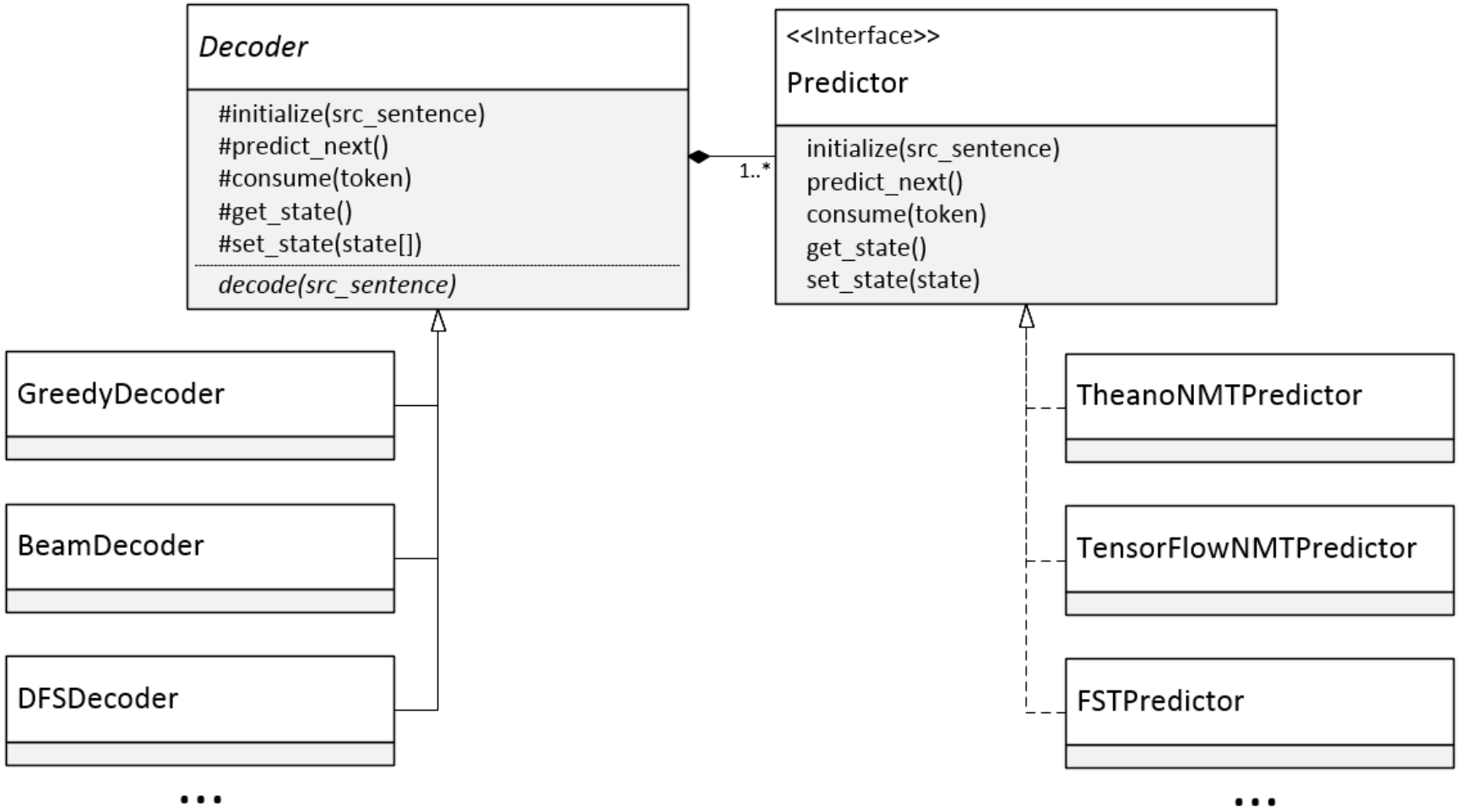
Motivation (3): Research agenda of our group

- We often see NMT as one component of a larger system
- We often work with different constraints and decoding strategies
- We often use multiple ways of scoring translations, e.g. n-gram posteriors, FSTs, …

SGNMT design principles

- Easy integration of new models, constraints, or NMT tools
- Easy implementation of new search strategies
- Easy combination of diverse scoring modules
- Computation time is secondary
  - Decoding is easily parallelisable on inexpensive CPUs (unlike training)

SGNMT software architecture



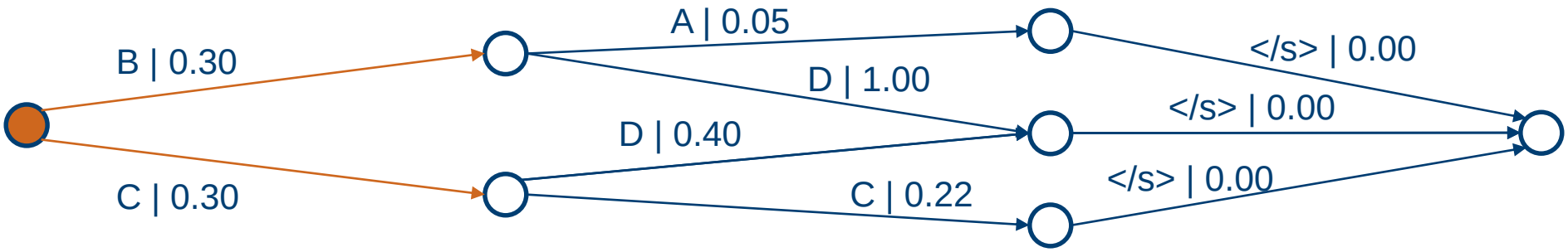
Example: Greedy lattice rescoring in SGNMT

nmt predictor:

A	0.40
B	0.70
C	0.52
UNK	1.30
</s>	1.30

fst predictor:

B	0.30
C	0.30



Example: Greedy lattice rescoring in SGNMT

nmt predictor:

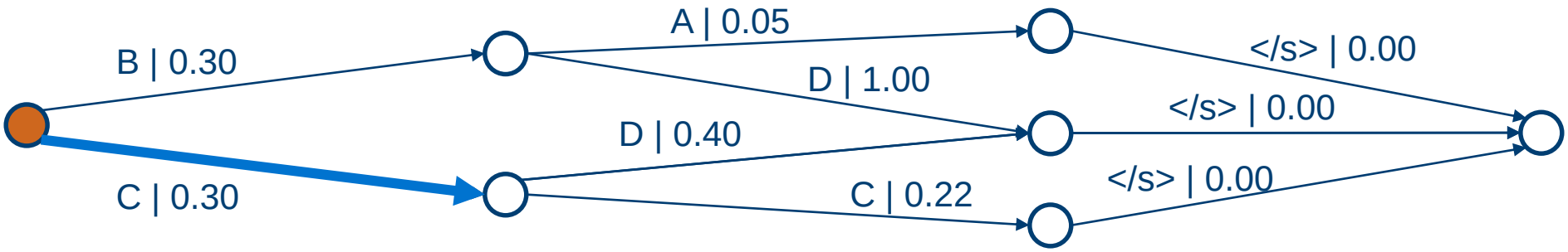
A	0.40
B	0.70
C	0.52
UNK	1.30
</s>	1.30

fst predictor:

B	0.30
C	0.30

combined:

B	1.00
C	0.82



Example: Greedy lattice rescoring in SGNMT

nmt predictor:

A	0.40
B	0.70
C	0.52
UNK	1.30
</s>	1.30

fst predictor:

B	0.30
C	0.30

combined:

B	1.00
C	0.82

nmt predictor:

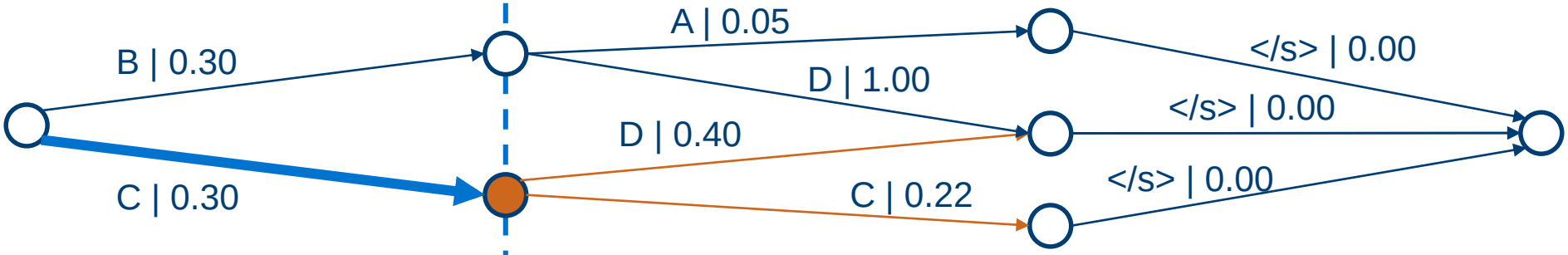
A	1.30
B	0.70
C	1.00
UNK	0.22
</s>	1.30

fst predictor:

C	0.22
D	0.40

combined:

C	1.22
D	0.64



Example: Greedy lattice rescoring in SGNMT

nmt predictor:

A	0.40
B	0.70
C	0.52
UNK	1.30
</s>	1.30

fst predictor:

B	0.30
C	0.30

combined:

B	1.00
C	0.82

nmt predictor:

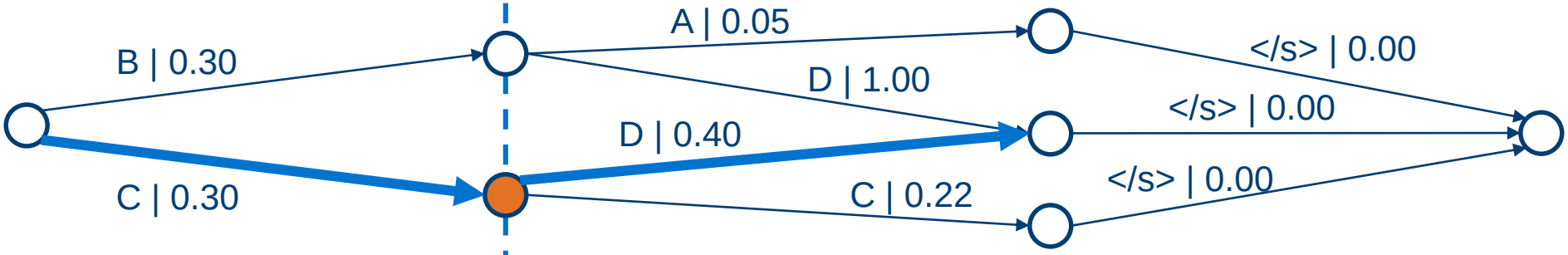
A	1.30
B	0.70
C	1.00
UNK	0.22
</s>	1.30

fst predictor:

C	0.22
D	0.40

combined:

C	1.22
D	0.64



Example: Greedy lattice rescoring in SGNMT

nmt predictor:

A	0.40
B	0.70
C	0.52
UNK	1.30
</s>	1.30

fst predictor:

B	0.30
C	0.30

combined:

B	1.00
C	0.82

nmt predictor:

A	1.30
B	0.70
C	1.00
UNK	0.22
</s>	1.30

fst predictor:

C	0.22
D	0.40

combined:

C	1.22
D	0.64

nmt predictor:

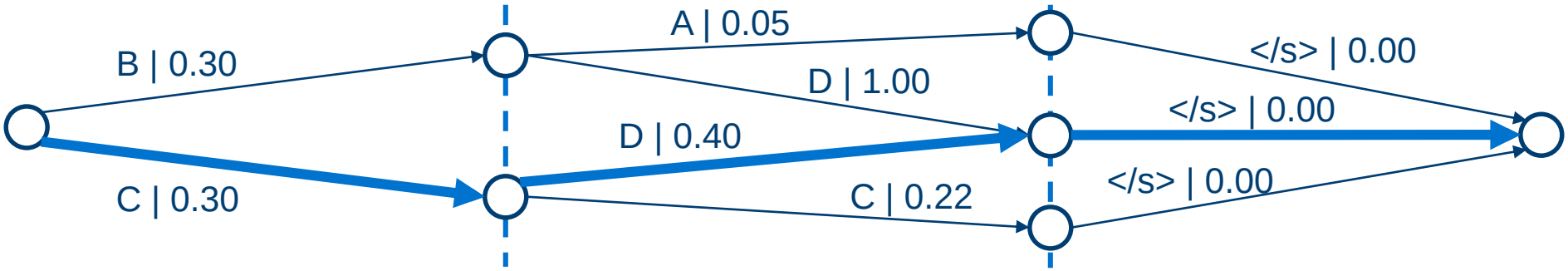
A	1.00
B	1.00
C	0.40
UNK	1.00
</s>	0.52

fst predictor:

</s>	0.00

combined:

</s>	0.52



Example configuration file: Lattice rescoring

```
predictors: fst,t2t


src_test: ./data/bpes/test.bpe.ids.ja


fst_path: ./lattices.test/%d.fst

t2t_src_vocab_size: 35786
t2t_trg_vocab_size: 32946
indexing_scheme: t2t

t2t_problem: translate_jaen_kyoto32k
t2t_checkpoint_dir: ./t2t_train/transformer/
t2t_model: transformer
t2t_hparams_set: transformer_base

outputs:  text,nbest,fst
```

Predictors

Path to source sentences

Path to lattices

General T2T settings

T2T model specification

Output plain text, n-best lists, and lattices

Search errors in beam search (lattice rescoring)

Japanese-English KFTT (Neubig, 2011)

	Average number of node expansions per sentence	Sentences with search errors	BLEU score
Exhaustive enumeration	652.3K	0%	21.7
Depth-first search with admissible pruning	3.0K	0%	21.7
Beam search (beam=20)	250.5	20.3%	21.9
Beam search (beam=4)	64.8	41.9%	21.9
Greedy decoding	18.0	67.9%	22.1

- Beam search yields a significant amount of search errors, but exhaustive search leads to a drop in BLEU score.

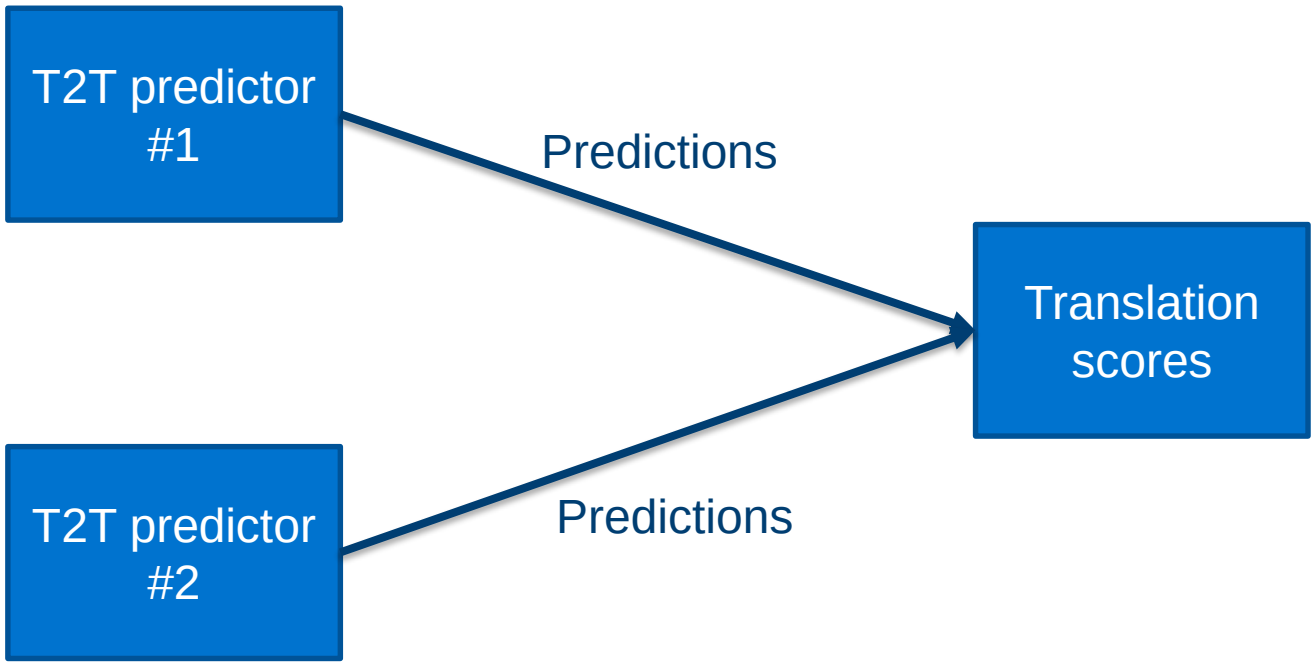
Example configuration file: T2T ensembles

```
predictors: t2t,t2t

src_test: ./data/bpes/test.bpe.ids.ja

t2t_src_vocab_size: 35786
t2t_trg_vocab_size: 32946
indexing_scheme: t2t
t2t_problem: translate_jaen_kyoto32k
t2t_model: transformer
t2t_hparams_set: transformer_base

t2t_checkpoint_dir: ./t2t_train/transformer/

t2t_checkpoint_dir2: ./t2t_train/transformer.2/

outputs:  text,nbest,fst
```

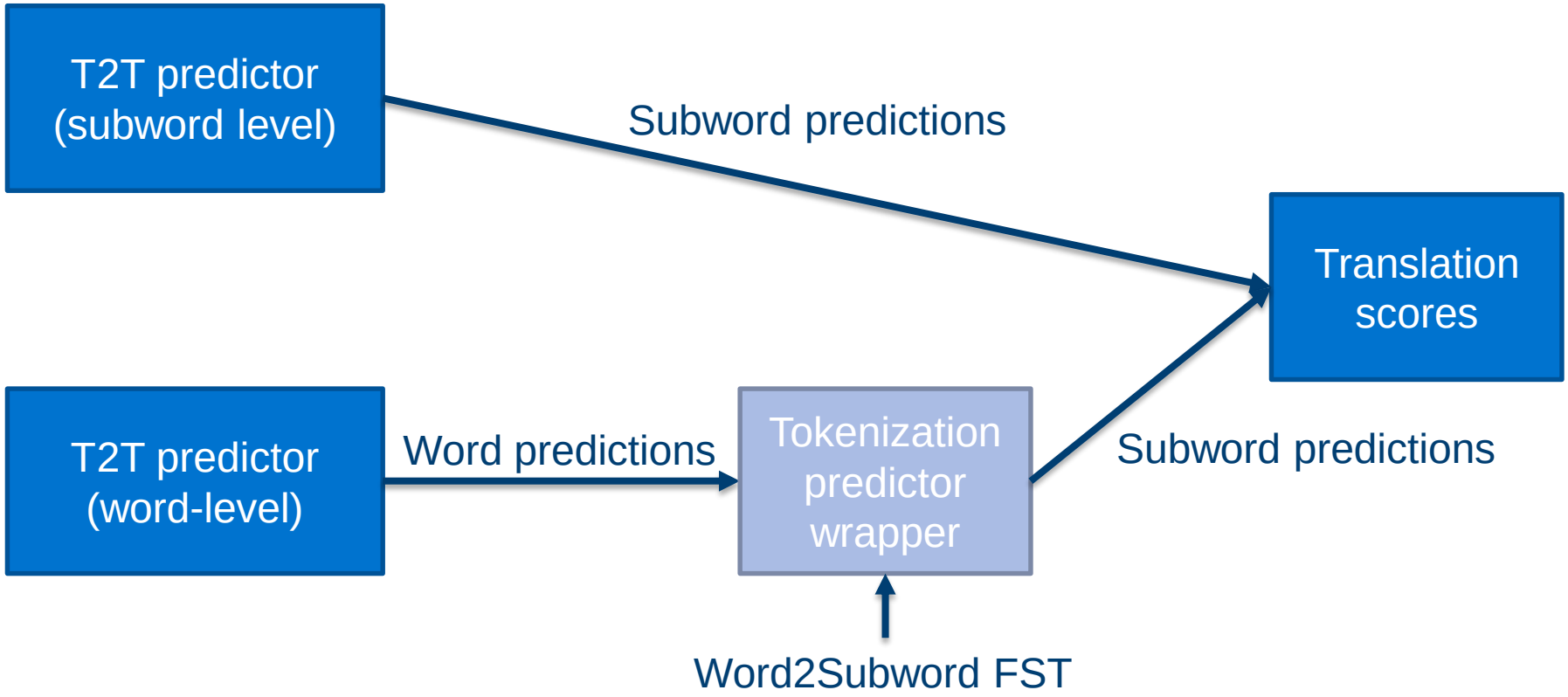
Two t2t predictors

Two checkpoint directories

T2T ensembing with SGNMT



T2T ensembing with SGNMT (word+subword)



Example configuration file: Mixing BPEs and words

```
predictors: t2t,fsttok_t2t

fsttok_path: word2bpe.fst

t2t_checkpoint_dir: ./t2t_train/bpe_transformer/
t2t_checkpoint_dir2: ./t2t_train/word_transformer/

...
```

Mixing words and subwords

BLEU scores on the Japanese-English KFTT test set (Neubig, 2011)

NMT (Word)	NMT (Subword)	SMT (MBR-based)	BLEU
✓			21.7
✓		✓	22.0
	✓		21.7
	✓	✓	22.5
✓	✓	✓	23.3

SMT baseline: 18.1 BLEU

MBR-based NMT-SMT hybrids: Felix Stahlberg, Adria de Gispert, Eva Hasler, Bill Byrne. Neural machine translation by minimising the Bayes-risk with respect to syntactic translation lattices. In EACL, 2017

NMT-SMT hybrids with different NMT backends

BLEU scores on the Japanese-English KFTT test set (Neubig, 2011)

	Pure NMT	SMT lattice rescoring	MBR-based NMT-SMT hybrid
Theano: Blocks (van Merriënboer et al., 2015)	18.4	18.9	19.0
TensorFlow: seq2seq tutorial[6]	17.5	19.3	19.2
TensorFlow: NMT tutorial[7]	18.8	19.1	20.0
TensorFlow: T2T Transformer (Google, 2017)	21.7	19.3	22.5

SMT baseline: 18.1 BLEU

- MBR-based combination of NMT and SMT yields gains across all investigated NMT implementations/models.

MBR-based NMT-SMT hybrids: Felix Stahlberg, Adria de Gispert, Eva Hasler, Bill Byrne. Neural machine translation by minimising the Bayes-risk with respect to syntactic translation lattices. In EACL, 2017

Impact

- 30 predictors and 15 search strategies currently available
  - Compatibility with Tensor2Tensor, Blocks/Theano, and the TF NMT tutorial
- Research: 8 publications using SGNMT so far
- Teaching: Used in the MPhil in Machine Learning, Speech and Language Technology at Cambridge
  - Course work (recasing experiments and NMT decoding strategies)
  - Student theses
    - Jiameng Gao. Variable length word encodings for neural translation models, MPhil dissertation
    - Marcin Tomczak. Bachbot. MPhil dissertation
- Industry: Part of the prototyping process at SDL plc.

Thanks

Code available at http://ucam-smt.github.io/sgnmt/html