

Multipass Decoding

in Moses

with Continuous Space LM (CSLM)

Loic Barrault
Fethi Bougares
Jacob Dlougach
Hieu Hoang

```
git clone git@github.com:moses-smt/mosesdecoder.git  
git checkout mtm13_multipass
```

CSLM

Score contiguous phrases

Issues

- First users outside of Le Mans/LIMSI

 - Problems compiling

- Dependent on Intel MKL

 - 30 evaluation license

- Change all weights in each pass

Hypothesis recombination

- System AR/EN, dev nist-nw09
- baseline : with recombination (default) : 52.41
- without-recombination (SS=200) : 51.73
- without-recombination (SS=300) : 51.74
- without-recombination (SS=400) : 51.99
- without-recombination (SS=1000) : 52.07
- without-recombination (SS=2000) : 52.12
- without-recombination (SS=3000) : 52.26
- without-recombination (SS=6000) : 52.47

Multipass Decoding

Rescoring hypothesis

- Look in hypothesis stack

New feature function

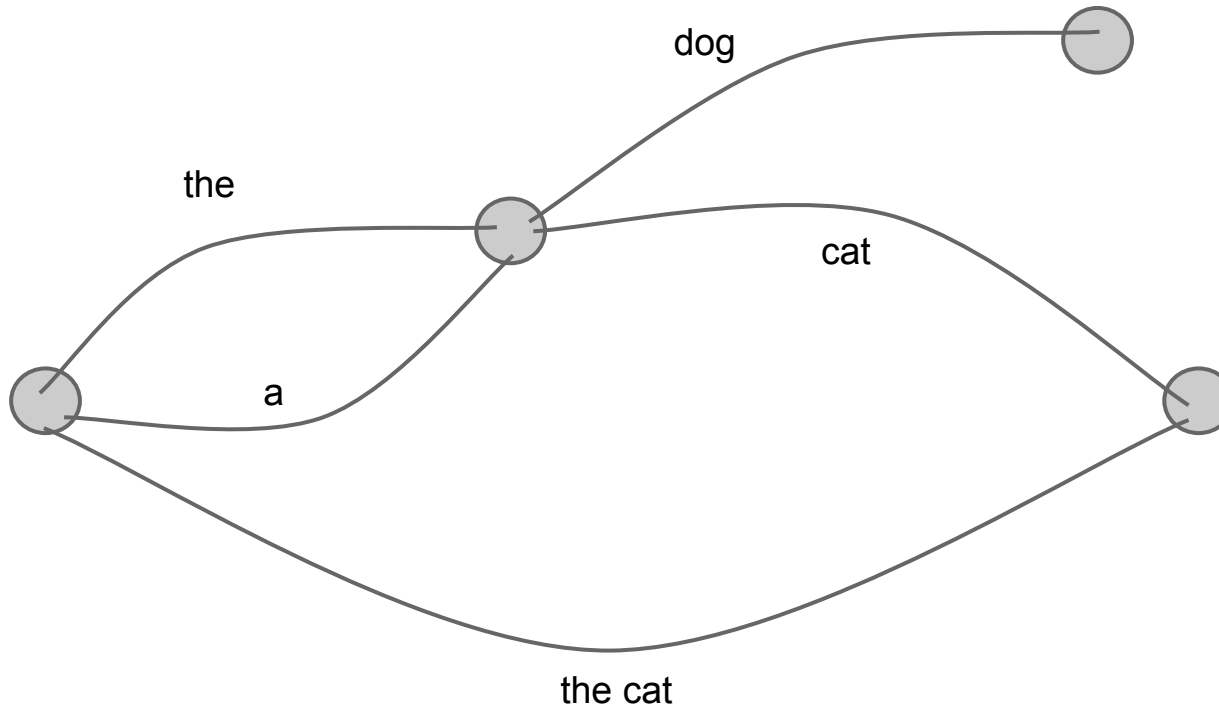
Issues:

Asynchronous calculation

State Splitting

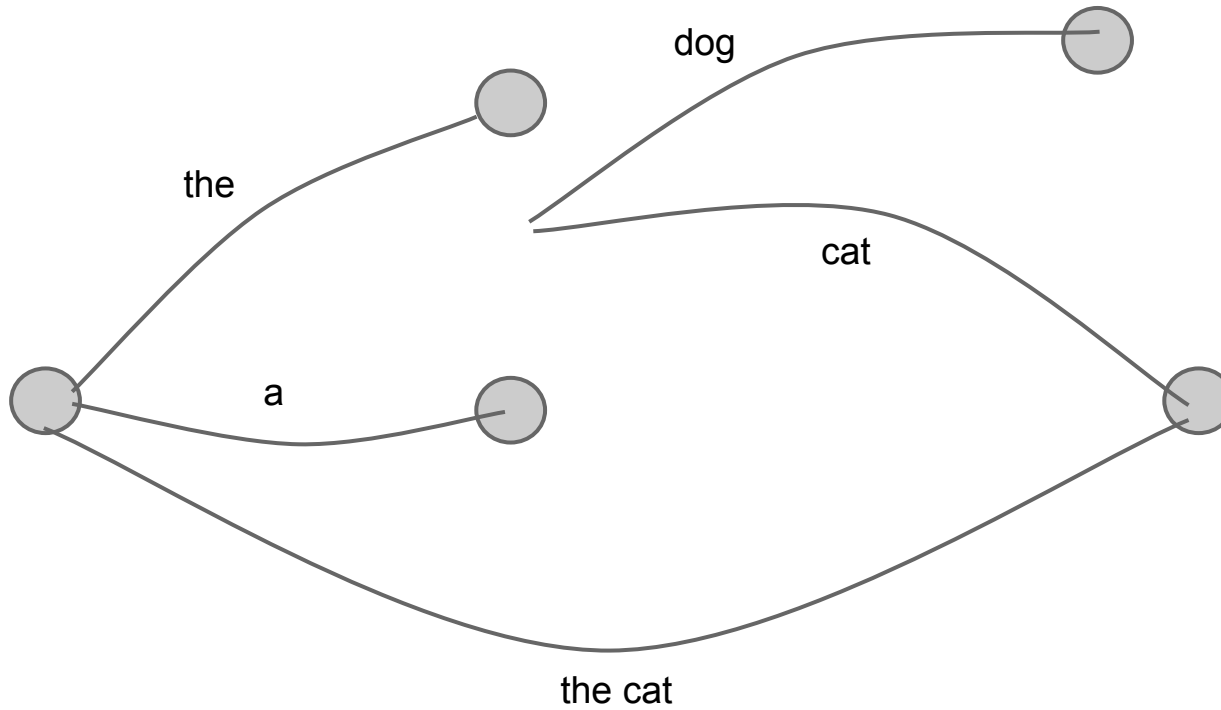
Lattice Rescoring

Input: le chat



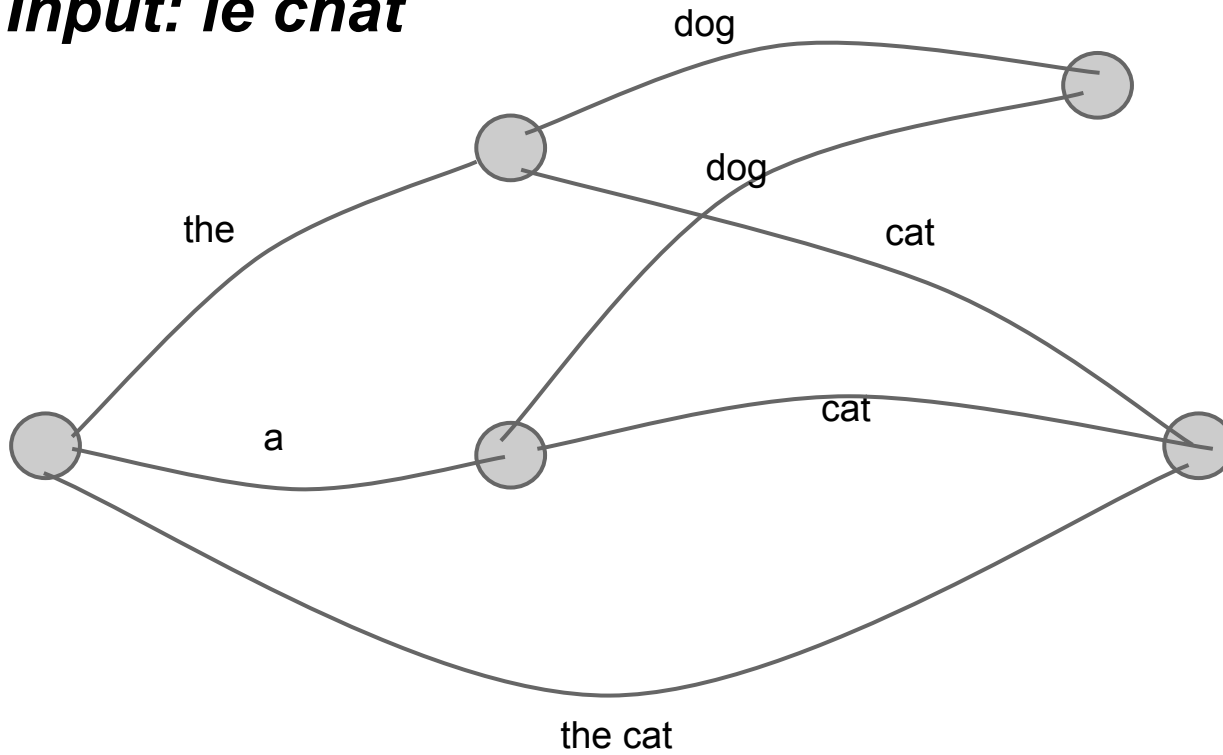
Lattice Rescoring

Input: le chat



Lattice Rescoring

Input: le chat



Tasks

Jacob

Created search graph

Allow state splitting

Hieu

Multipass decoding in Moses

Asynchronous feature functions

Loic

CSLM

Fethi

Experiments

Search graph structure

Moses has many different search graph structures for different purposes.

Abstract view of search graph:

- Each edge matches a hypothesis
- Each vertex matches a winning hypothesis
- Copying all information from stacks instead of referencing
- Main graph operation - iteration through edges

Search graph construction

1. Construction from decoder's stacks: DFS on hypotheses starting from last stack to get topologically sorted graph.
2. Construction from other graph with rescoring on new set of stateless features: reuse existing scores and calculate new ones.
3. Construction from other graph with stateful features: make search graph a *new input method* for decoding. Every edge is now a translation option.

Conclusion

Interesting Project
Ambitious
To be continued...