University of Glasgow

AI4BioMed

Accelerating Cross-Encoders in Biomedical Entity Linking

Javier Sanz-Cruzado & Jake Lever

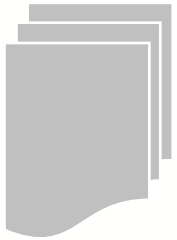
WORLD CHANGING GLASGOW

A WORLD TOP 100 UNIVERSITY

BioNLP @ ACL 2025, 1st August 2025

1. Biomedical entity linking

What is biomedical entity linking?



Biomedical documents

“Varicella” is a highly contagious **“viral infection”** that causes an acute **“fever”** and **“blistered rash”**, mainly in children. **“Immunocompromised patients”** infected with the **“virus”** need **“intravenous treatment”** with the **“antiviral”** **“aciclovir”.**

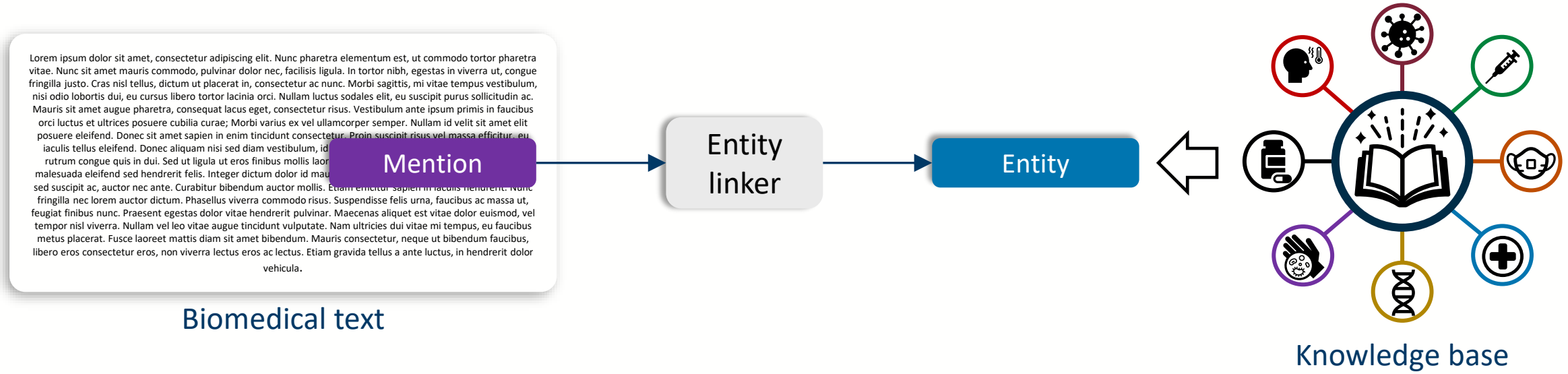
“Varicella” →

- Varicella
- Chickenpox
- Varicella-Zoster Virus

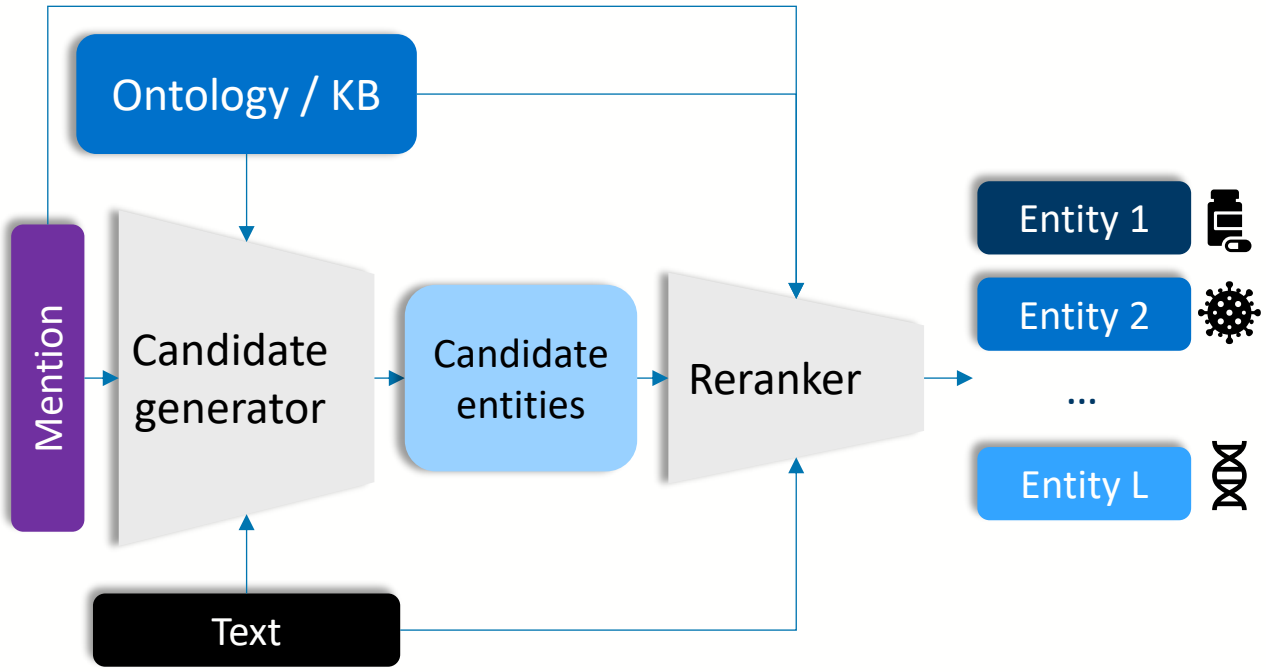
Knowledge base

What is biomedical entity linking?

Biomedical entity linking matches mentions of biomedical concepts (diseases, chemicals) in text with unique entities within a knowledge base



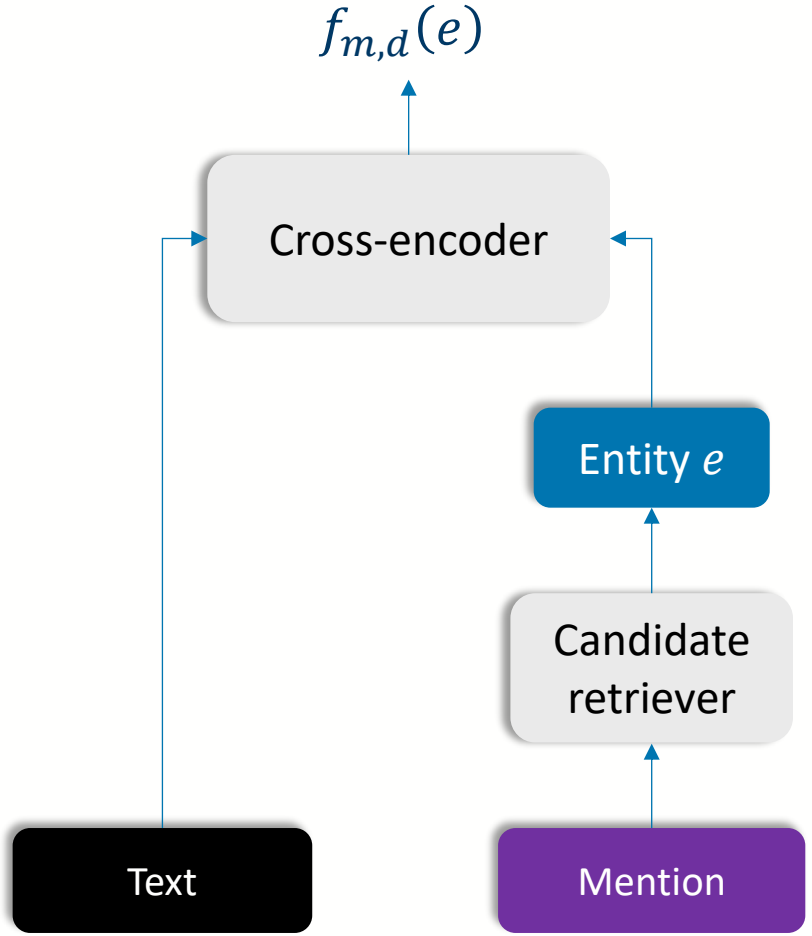
Common architecture for entity linking



- Two stages
- Candidate generator
  - Inspects all entities
  - Get the top k more relevant entities $k \ll |\mathcal{E}|$
  - Computationally efficient
  - Maximize recall@k
  - Ex: n-grams entity linkers
- Reranker
  - Inspects the top candidates from first phase
  - Precise ranker
  - Maximize accuracy
  - Computationally expensive
  - Ex: cross-encoder

Cross-encoder reranker

- Transformer-based model (encoder-only model)
  - BERT
  - BiomedBERT
  - Longformer
  - ModernBERT
- Inputs:
  - Text containing a mention
  - A candidate entity
- Output:
  - $f_{m,d}(e)$: the score for the entity
- Rank entities by descending score



How does the cross-encoder work?

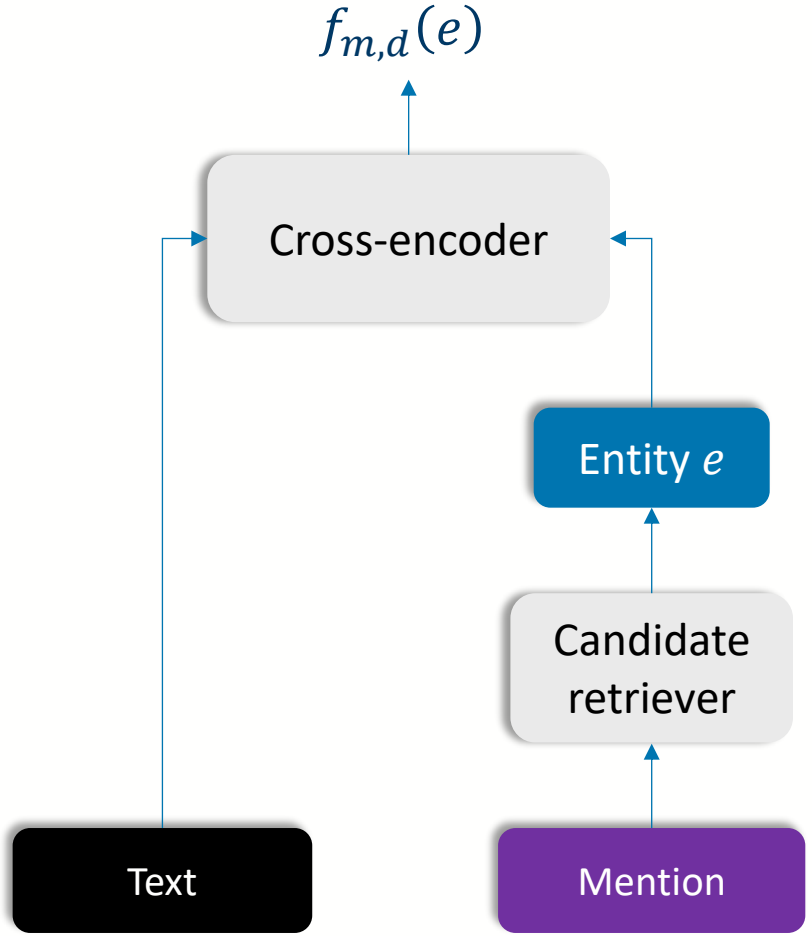
- It receives a sentence following a template

  Text [SEP] Mention [MASK] Entity name

- The mention is contained in the text
- The text here provides additional context

- The [MASK] token can take two values:
  - 1 – if the entity corresponds to the mention
  - 0 – otherwise

- Therefore, the score of the entity is the probability of the [MASK] token taking value 1



University of Glasgow

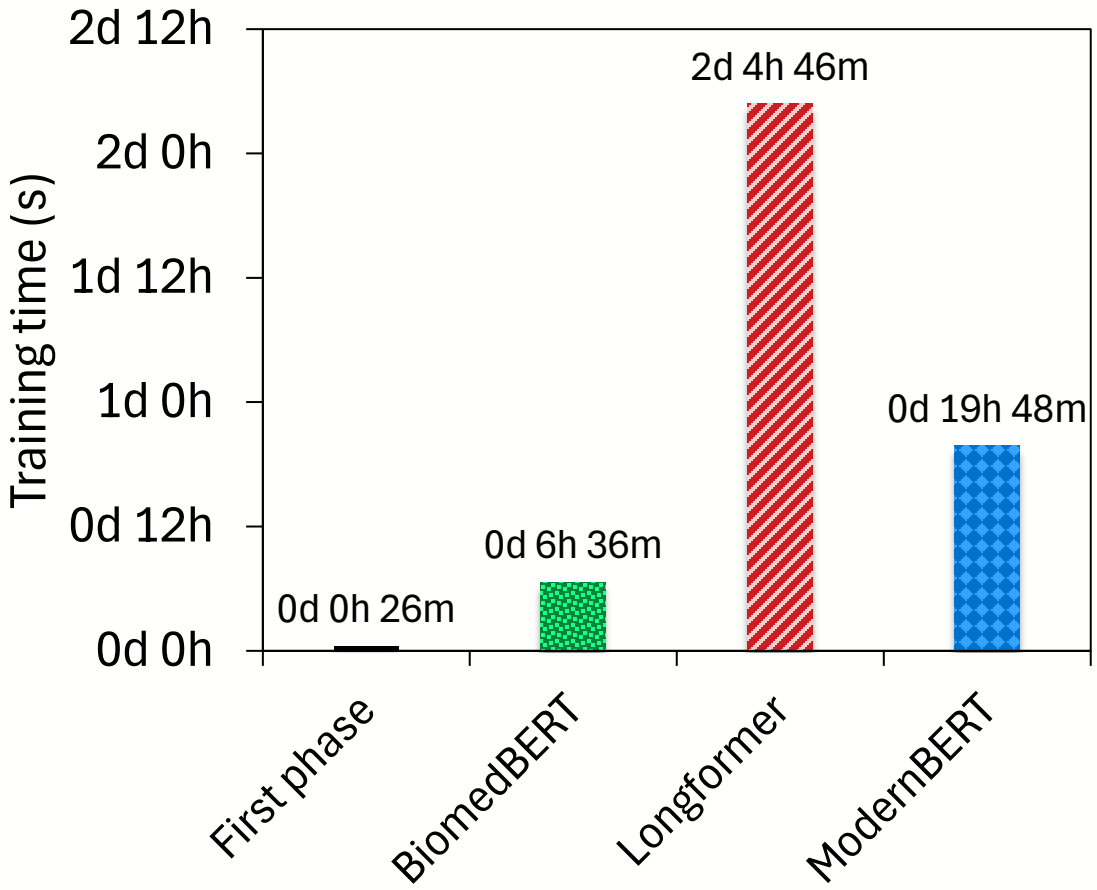
Problem with cross-encoders

Training and inference with cross-encoders is very slow

Example on MedMentions dataset

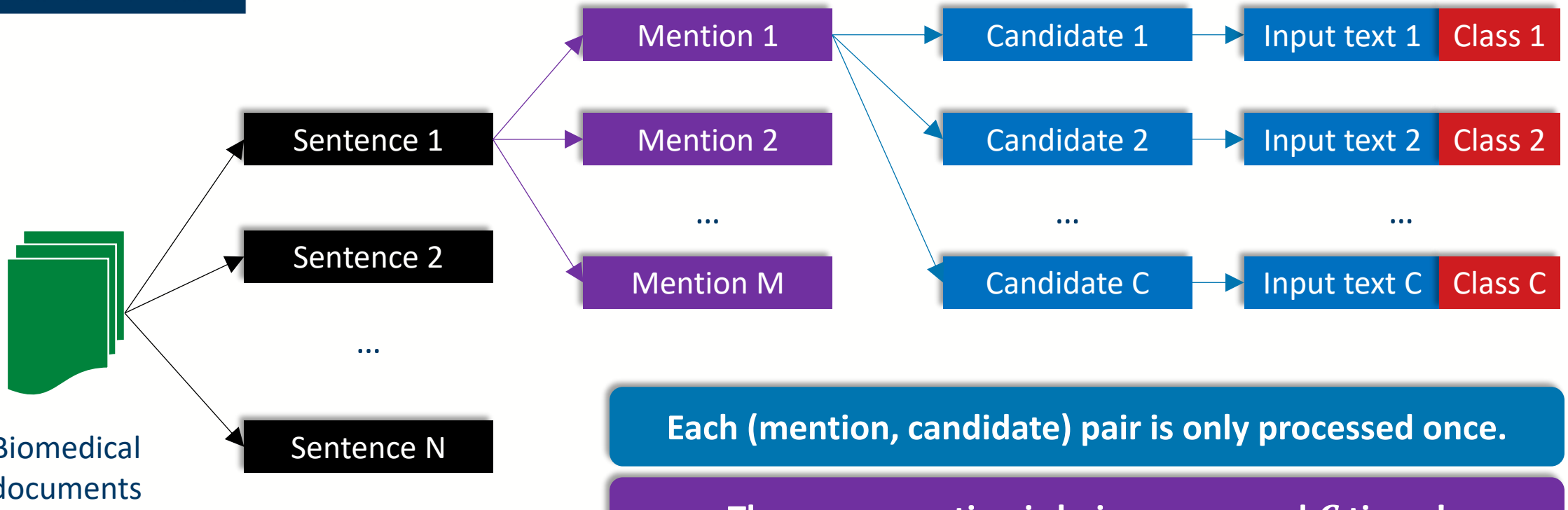
- Training a first-phase n-grams model takes half an hour
- Fastest cross-encoder reranker takes > 6 hours to train
- That's, at least, 12 times more!
- Similar observations can be observed on inference time.

Can we accelerate cross-encoder rerankers without harming effectiveness?



2. Accelerating cross-encoders

Let's review the cross-encoder working



Each (mention, candidate) pair is only processed once.

The same mention is being processed C times!

The same sentence text is being processed $C \times M$ times!

And the document has N different sentences!

University of Glasgow

Accelerating cross-encoders

Each (mention, candidate) pair is only processed once.

The same mention is being processed C times!

The same sentence text is being processed $C \times M$ times!

And the document has N different sentences!

Both in training and inference

Idea: Can we accelerate training / inference by showing each text less times to the cross-encoder?

University of Glasgow

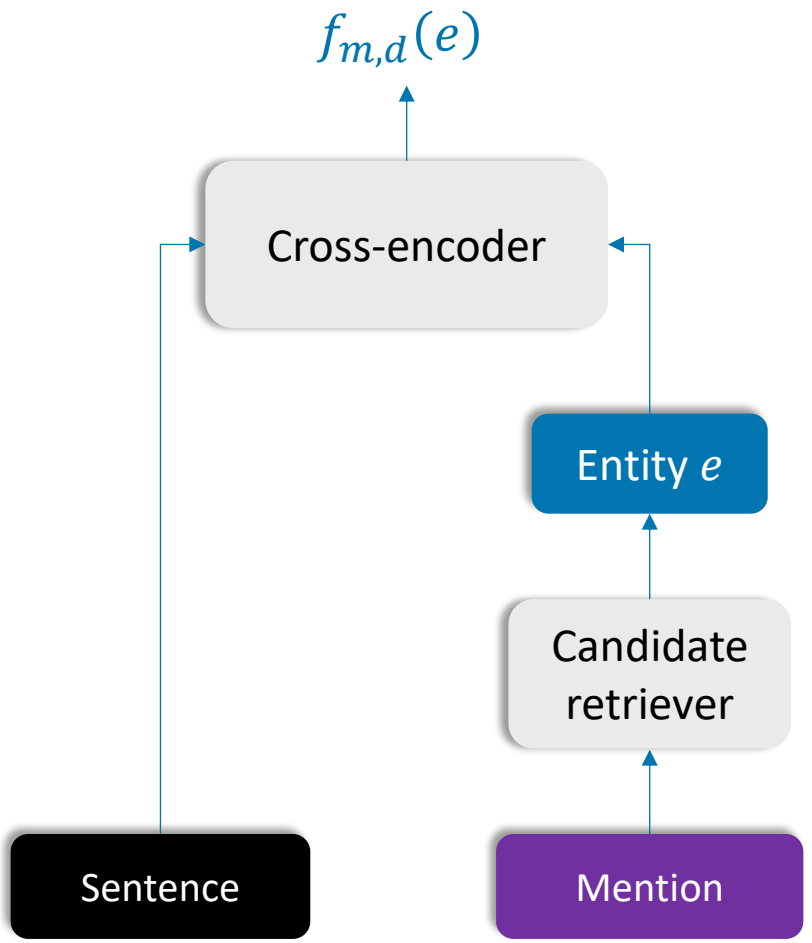
Solution 1: Parallel cross-encoder

Each (mention, candidate) pair is only processed once.

The same mention is being processed C times!

The same sentence text is being processed $C \times M$ times!

And the document has N different sentences!



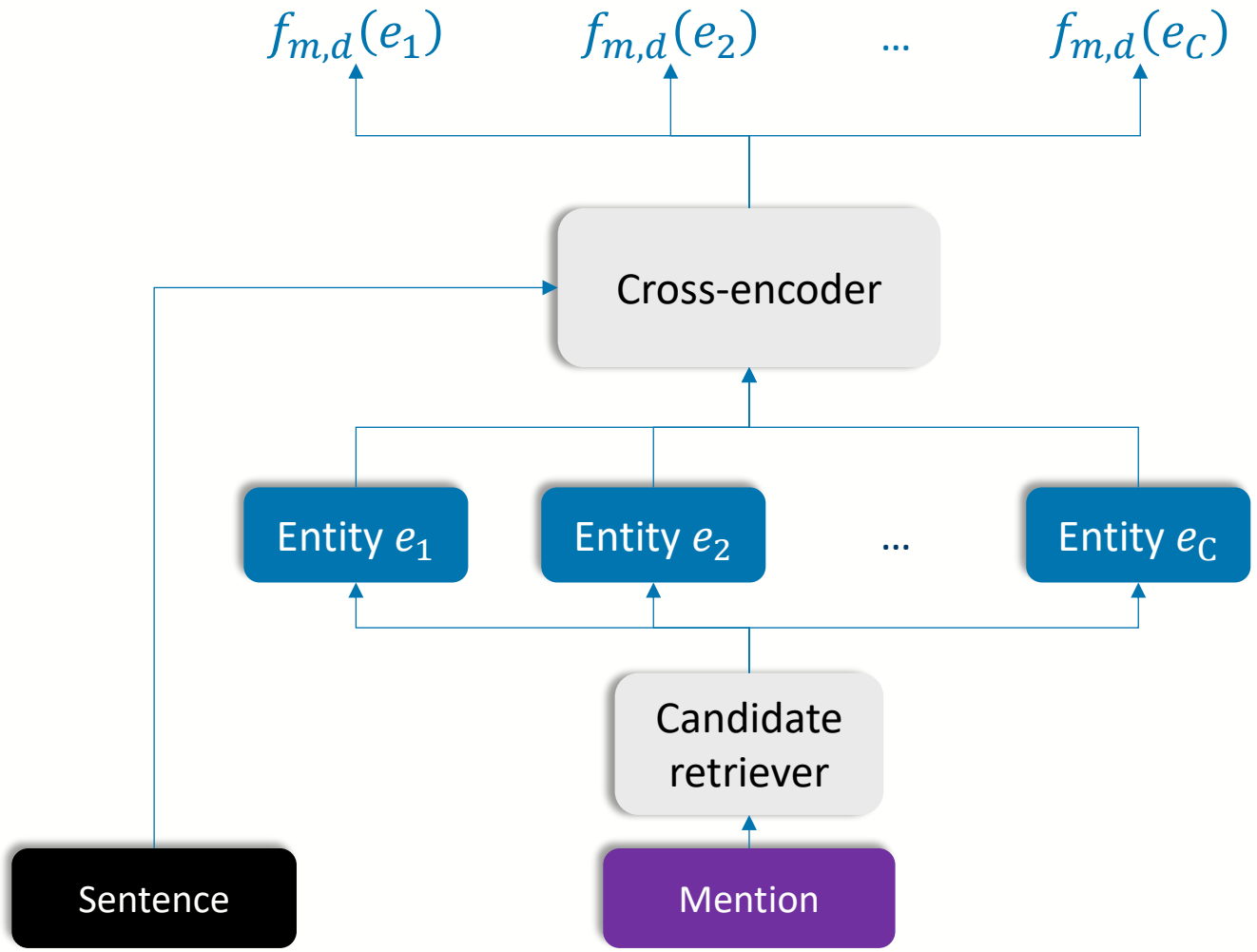
Solution 1: Parallel cross-encoder

Each (mention, candidate) pair is only processed once.

The same mention is being processed C times!

The same sentence text is being processed $C \times M$ times!

And the document has N different sentences!



University of Glasgow

Solution 1: Parallel cross-encoder

- While the cross encoder uses template for each candidate:

Text [SEP] Mention [MASK] Entity e name

Each (mention, candidate) pair is only processed once.

The same mention is being processed C times!

The same sentence text is being processed $C \times M$ times!

And the document has N different sentences!

University of Glasgow

Solution 1: Parallel cross-encoder

Each (mention, candidate) pair is only processed once.

The same mention is being processed C times!

The same sentence text is being processed $C \times M$ times!

And the document has N different sentences!

- The parallel cross-encoder receives a sentence using the following a template for each mention:

Text [SEP] Mention [MASK] Entity e_1 name
[SEP] Mention [MASK] Entity e_2 name
…
[SEP] Mention [MASK] Entity e_C name

- Therefore, the score of the entity e_i is the probability of its [MASK] token taking value 1

University of Glasgow

Solution 1: Parallel cross-encoder

Each (mention, candidate) pair is only processed once.

The same mention is being processed once.

The same sentence text is being processed $C \times M$ times!

And the document has N different sentences!

- The parallel cross-encoder receives a sentence using the following a template for each mention:

Text [SEP] Mention [MASK] Entity e_1 name
[SEP] Mention [MASK] Entity e_2 name
…
[SEP] Mention [MASK] Entity e_C name

- Therefore, the score of the entity e_i is the probability of its [MASK] token taking value 1

But, every sentence can have more than one mention!

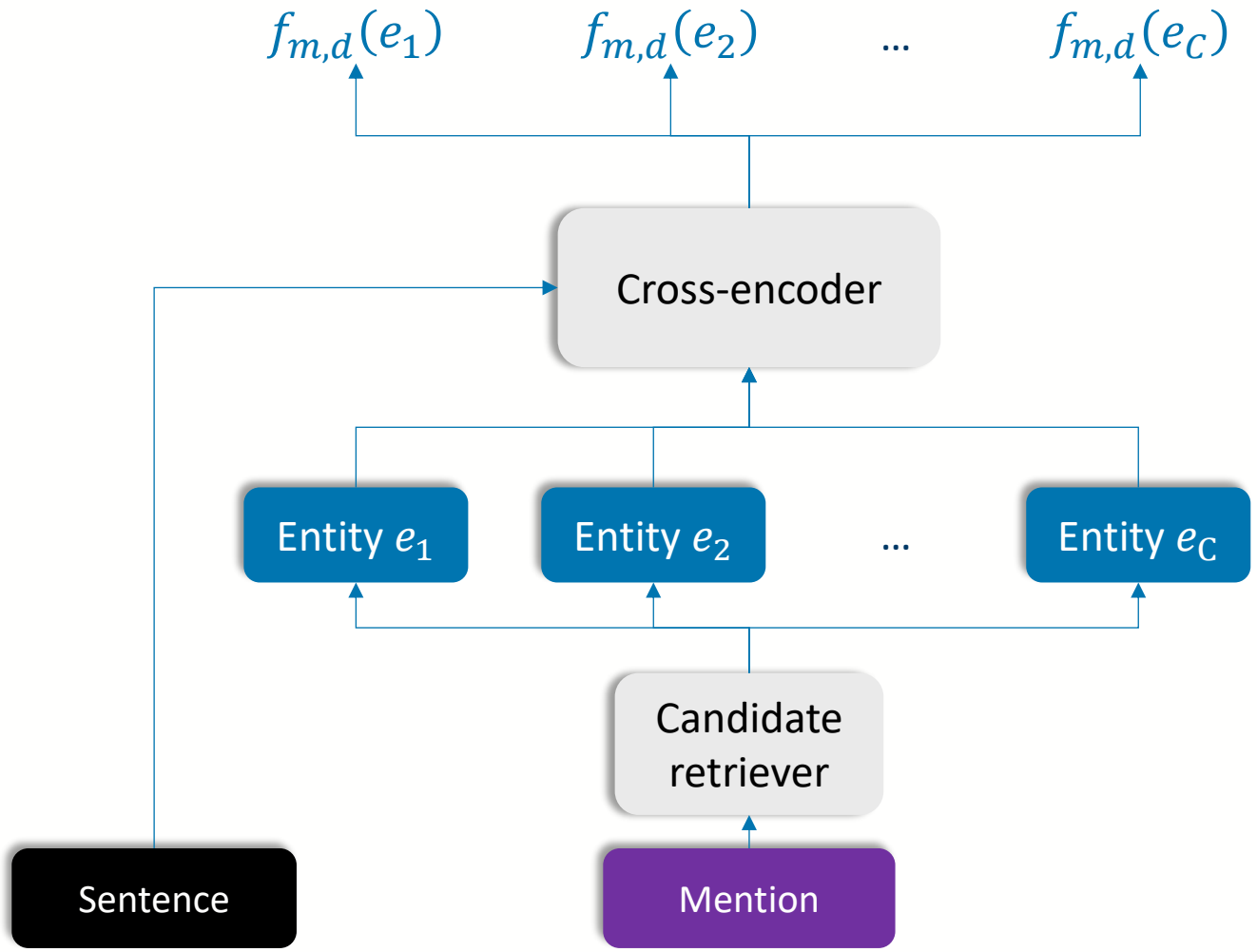
Solution 2: Multi cross-encoder

Each (mention, candidate) pair is only processed once.

The same mention is only processed once

The same sentence text is being processed $C \times M$ times!

And the document has N different sentences!



University of Glasgow

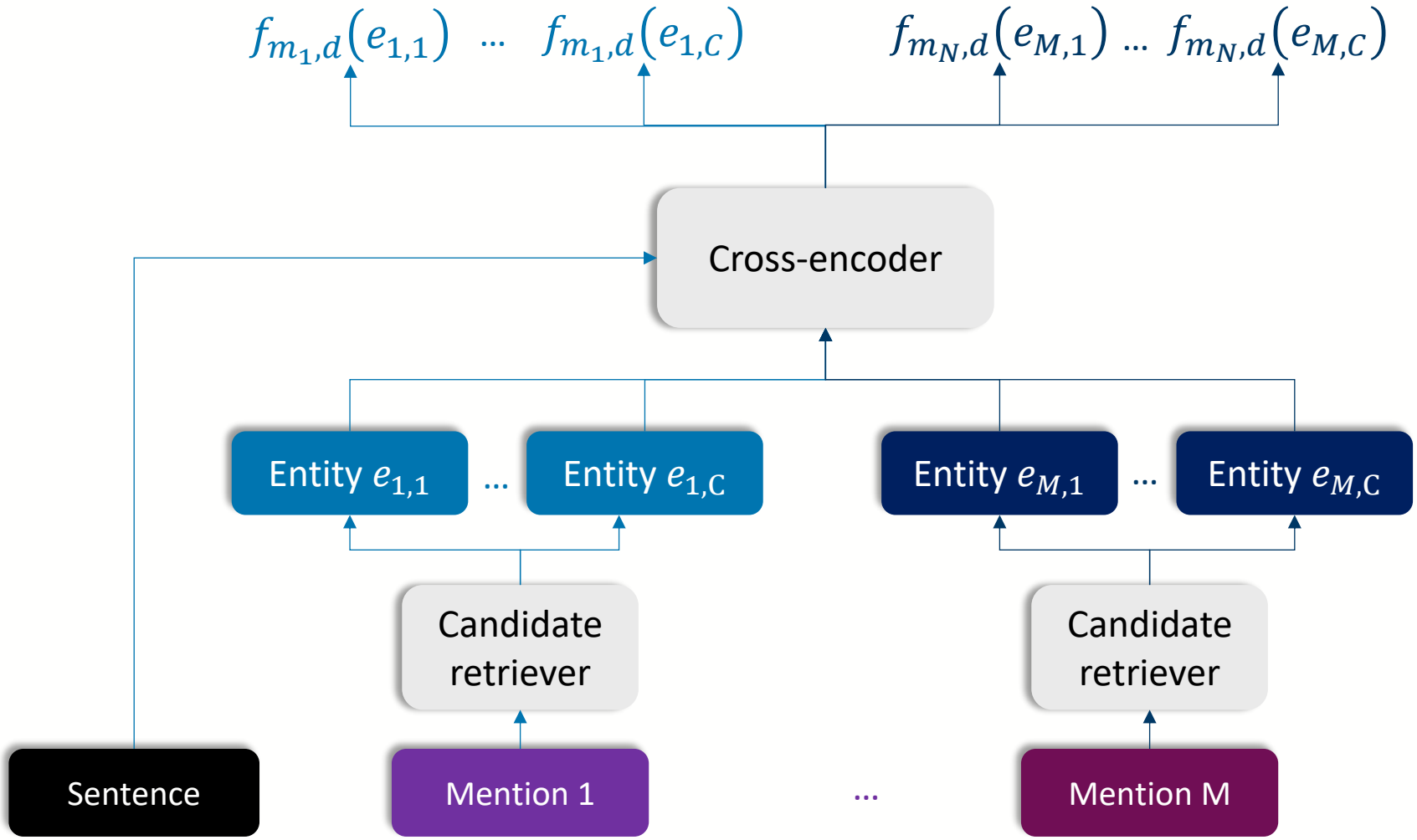
Solution 2: Multi cross-encoder

Each (mention, candidate) pair is only processed once.

The same mention is only processed once

The same sentence text is being processed $C \times M$ times!

And the document has N different sentences!



University of Glasgow

Solution 2: Multi cross-encoder

Each (mention, candidate) pair is only processed once.

The same mention is only processed once

The same sentence text is being processed $C \times M$ times!

And the document has N different sentences!

- Use a similar trick to the parallel cross-encoder
- The new template is:

Text [SEP] Mention 1 [MASK] Entity $e_{1,1}$ name
[SEP] Mention 1 [MASK] Entity $e_{1,2}$ name
…
[SEP] Mention 1 [MASK] Entity $e_{1,C}$ name
…
[SEP] Mention M [MASK] Entity $e_{M,1}$ name
[SEP] Mention M [MASK] Entity $e_{M,2}$ name
…
[SEP] Mention M [MASK] Entity $e_{M,C}$ name

- And, again, the score for each entity and mention is the probability of the [MASK] token being one

Solution 3: Document cross-encoder

Each (mention, candidate) pair is only processed once.

The same mention is only processed once

The same sentence text is only processed once

And the document has N different sentences!

- The previous trick can be further applied
- Instead of processing one sentence, we can process multiple at the same time.
- How? Concatenating the templates for a sentence using a [SEP] token
- We call this document cross-encoder
- **Note:** if each document is divided in passages, we can have an intermediate cross-encoder. We denote this as passage cross-encoder

Solution 3: Document cross-encoder

Each (mention, candidate) pair is only processed once.

The same mention is only processed once

The same sentence text is only processed once

The document is only processed once

- The previous trick can be further applied
- Instead of processing one sentence, we can process multiple at the same time.
- How? Concatenating the templates for a sentence using a [SEP] token
- We call this document cross-encoder
- **Note:** if each document is divided in passages, we can have an intermediate cross-encoder. We denote this as passage cross-encoder

3. Experiments and evaluation

University of Glasgow

Research questions

Research question 1

How does the parallelism of the cross-encoder affect the effectiveness of the model?

Research question 2

How does the parallelism of the cross-encoder affect the training and inference speeds?

University of Glasgow

Experimental setup

- We test our models on four biomedical datasets:
  - **MedMentions:** PubMed abstracts annotated with entities in UMLS 2017AA
  - **NCBI Disease:** PubMed abstract annotated with disease mentions of entities in the MEDIC ontology
  - **NLM Chem:** Full-text PubMed Central articles, with annotated mentions of chemical entities in MeSH 2021
  - **BC5CDR:** PubMed abstracts with chemical and disease annotations. Linked with MeSH 2015.

University of Glasgow

Algorithms

- **First stage candidate retrieval:** n-grams TF-IDF
  - 3-grams for MedMentions, 2-grams for the other datasets
  - Compute 5 candidates for each mention

- **Second stage:**
  - Baseline: base cross-encoder
  - Parallel cross-encoder
  - Multi cross-encoder
  - Document cross-encoder

University of Glasgow

Cross-encoder configurations

- **Backbone LMs:** We use models with different context-window size
  - BiomedBERT: 512
  - Longformer: 4096
  - ModernBERT: 8192

- **Early stopping:** if F1 is not improved on the validation set after three epochs
- **Learning rate:** all cross-encoders use the same one (1e-6)
- **Batch size:** depends on backbone model (fit on a single 4090)
- **Loss function:** cross-entropy loss

- **Hardware:** 2 CPU, 16 GB RAM, 1 Nvidia RTX 4090 GPU

University of Glasgow

Metrics

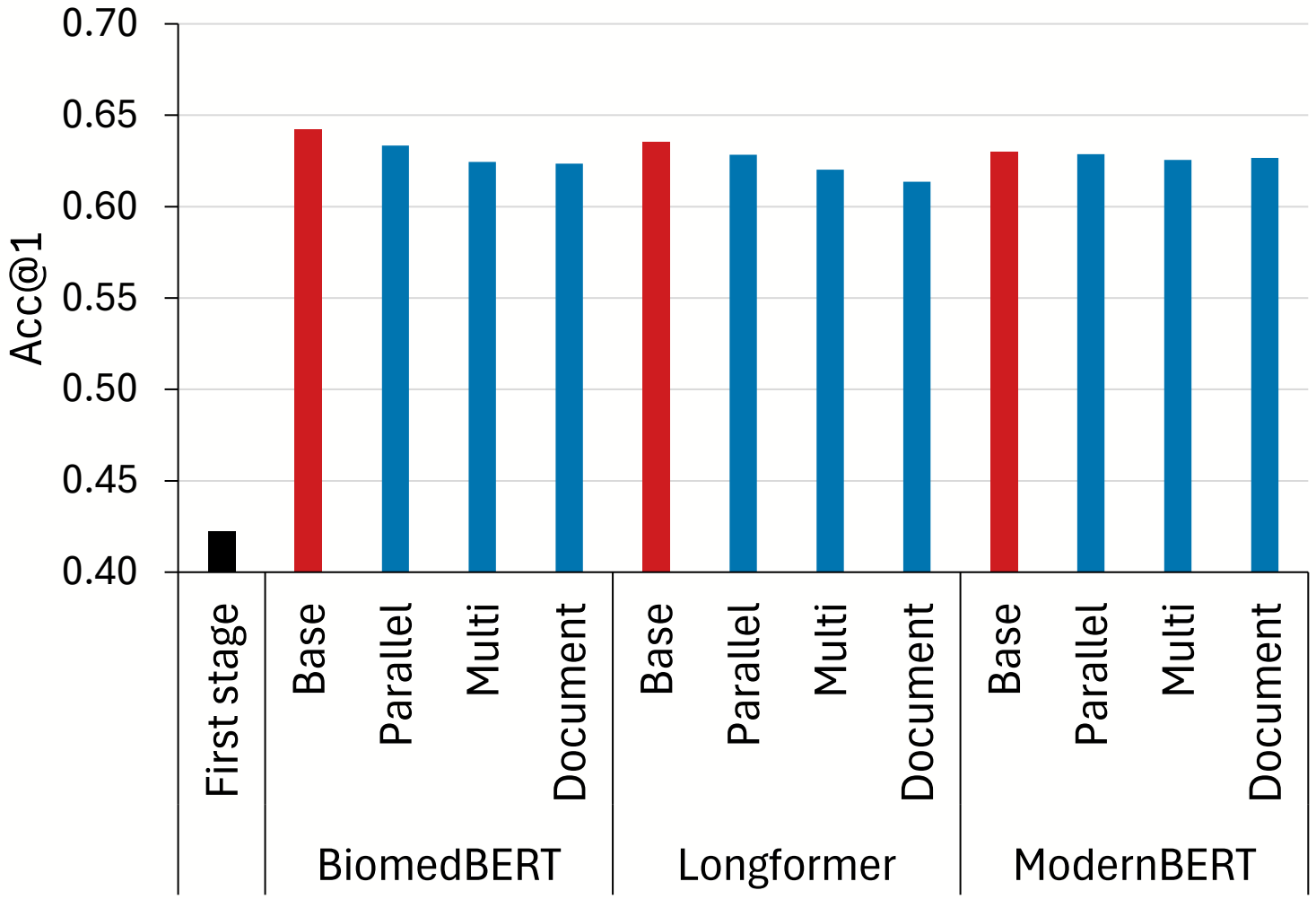
- **Acc@1:** is the top-ranked entity correct?

- **Training speed:**
  - How many training examples (mention, candidate) pairs can we process per second?
  - Ensures fair comparison, as different models might run for different epochs.

- **Inference speed:**
  - How many inference examples (mention, candidate) pairs can we process per second?

University of Glasgow

RQ1: Effectiveness (MedMentions)



Cross-encoders improve effectiveness of the first stage model

Adding more information reduces Acc@1 on MedMentions

But difference is small (between 0.54% and 3.42% loss)

University of Glasgow

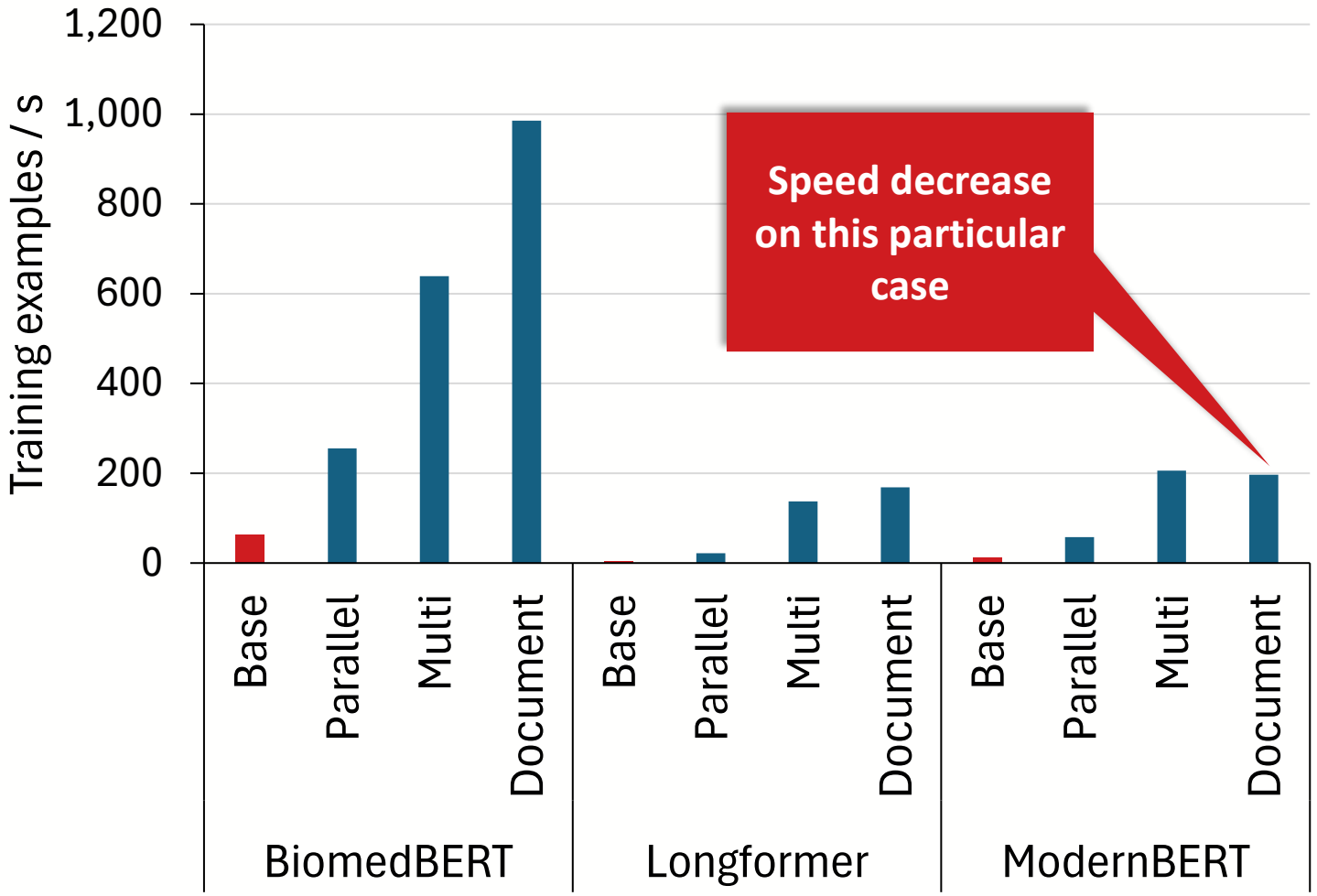
RQ1: Effectiveness

Adding more (mention, entity) pairs to the cross-encoder has limited impact on accuracy.

All the proposed cross-encoders are reasonable entity linking rerankers

Different datasets can react differently to the parallelism of the cross-encoders.

RQ2: Training speed (MedMentions)



Parallel cross-encoders accelerate the training between 3.12 and 3.9 times

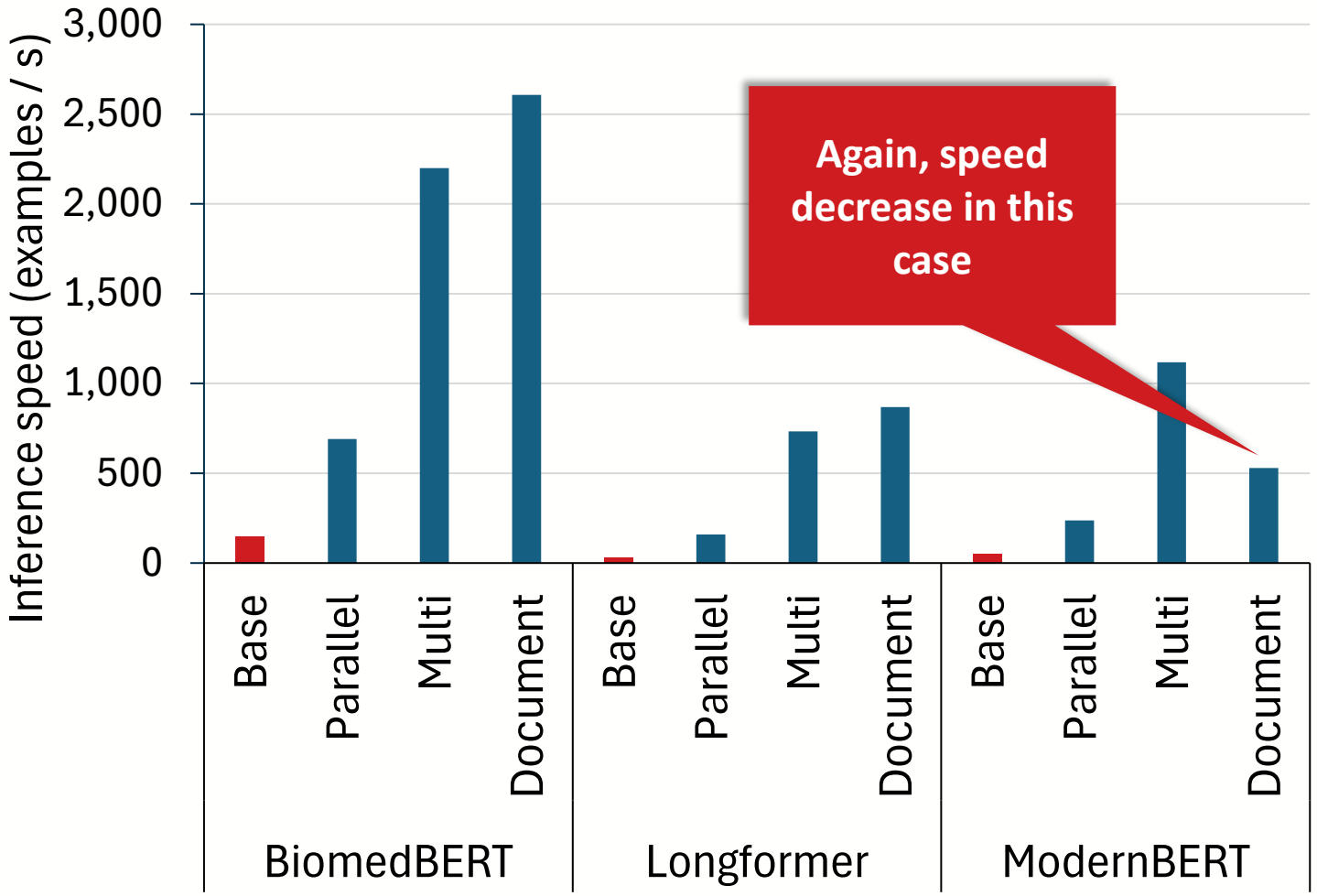
Multi cross-encoders accelerate the training between 9.3 and 29.93 times

Document cross-encoders accelerate the training between 14.88 and 36.97 times

Similar patterns are observed on other datasets

University of Glasgow

RQ2: Inference speed (MedMentions)



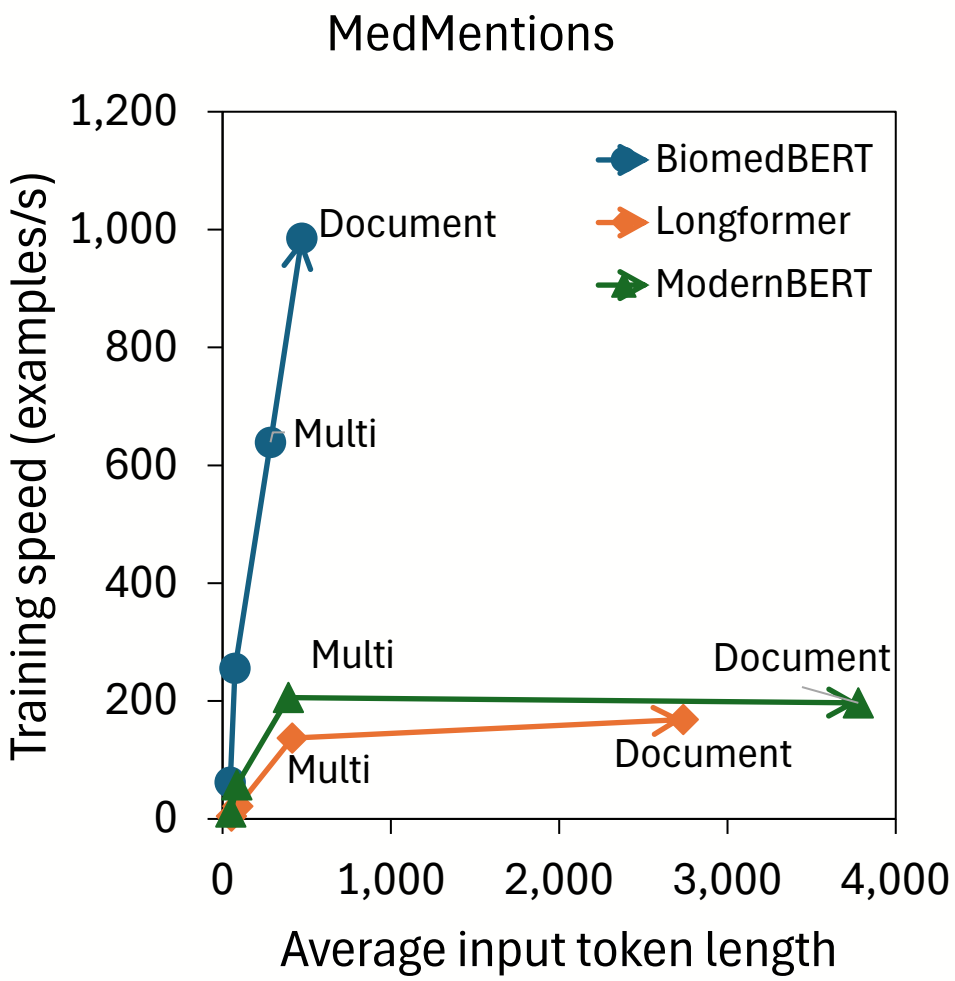
Parallel cross-encoders accelerate the inference between 3.75 and 4 times

Multi cross-encoders accelerate the inference between 15 and 22.18 times

Document cross-encoders accelerate the inference between 9.83 and 26.47 times

Similar patterns are observed on other datasets

RQ2: Limitations on speed improvements



ModernBERT document cross-encoder works slower than the ModernBERT multi cross-encoder. WHY?

- ModernBERT has a longer context window (8192 vs. 4096 of Longformer)
- Therefore, input strings for ModernBERT can be longer than 4096 characters.
- When this happens, training speed diminishes.
- Very lengthy input strings can hinder the efficiency of the transformer.
- Although it is still much faster than a base cross-encoder.

University of Glasgow

RQ2: Efficiency

Adding more (mention, entity) pairs to the cross-encoder greatly increases training speed.

Adding more (mention, entity) pairs to the cross-encoder greatly increases inference speed.

Very lengthy input sentences can hinder the efficiency of the models.

Conclusions

University of Glasgow

Conclusions

- **We can accelerate cross-encoders by allowing them to classify multiple (mention, entity) pairs at once**
  - As we add more information, training / inference speeds improve
  - Training speed: between 2.68 and 36.97 times faster
  - Inference speed: between 3.8 and 26.47 times faster

- **Adding more information produces small effects on performance**
  - Usually, parallel cross-encoders achieve slightly better performance
  - Document cross-encoders worsen base performance
  - Differences in a -3.42% to 2.76% differences

- **We can have a major training/inference speed improvement at a small accuracy cost!**

This research was funded by the U.S. National Cancer Institute (NCI), with grant number U24CA275783.

Questions?

Dr. Javier Sanz-Cruzado
AI4BioMed Group, University of Glasgow

javier.sanz-cruzadopuig@glasgow.ac.uk
JavierSanzCruza
Javiersanzcruza.bsky.social
https://www.linkedin.com/in/javier-sanz-cruzado-puig/