Peeling Back the Layers: An In-Depth Evaluation of Encoder Architectures in Neural News Recommenders

Andreea Iana[1], Goran Glavaš[2], Heiko Paulheim[1]

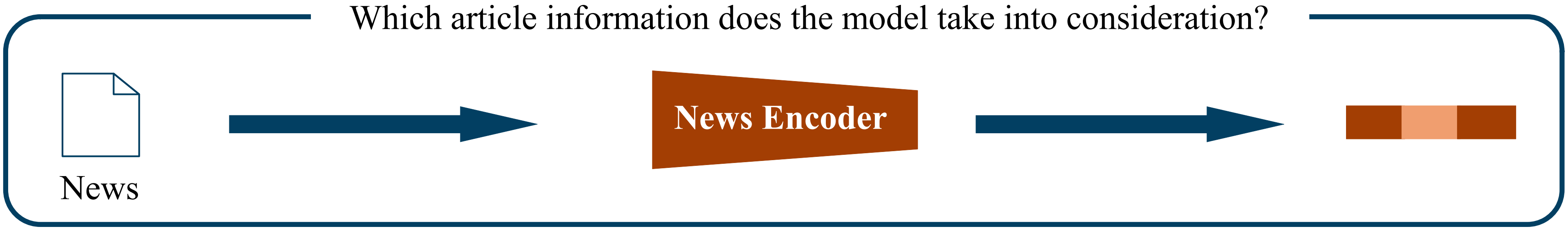
[1]Data and Web Science Group, University of Mannheim, Germany

[2]Center for Artificial Intelligence and Data Science, University of Würzburg, Germany

UNIVERSITÄT MANNHEIM

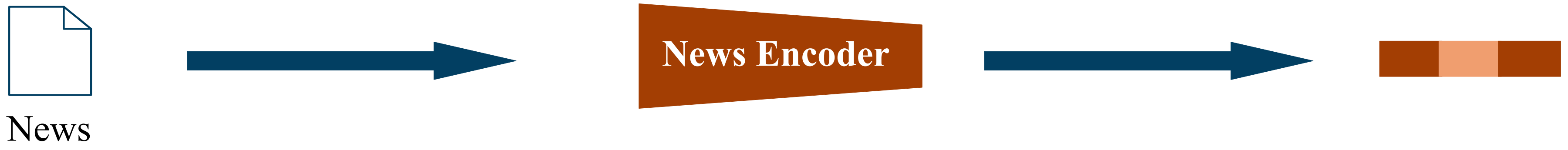
Julius-Maximilians-
UNIVERSITÄT
WÜRZBURG

Content-based Neural News Recommenders

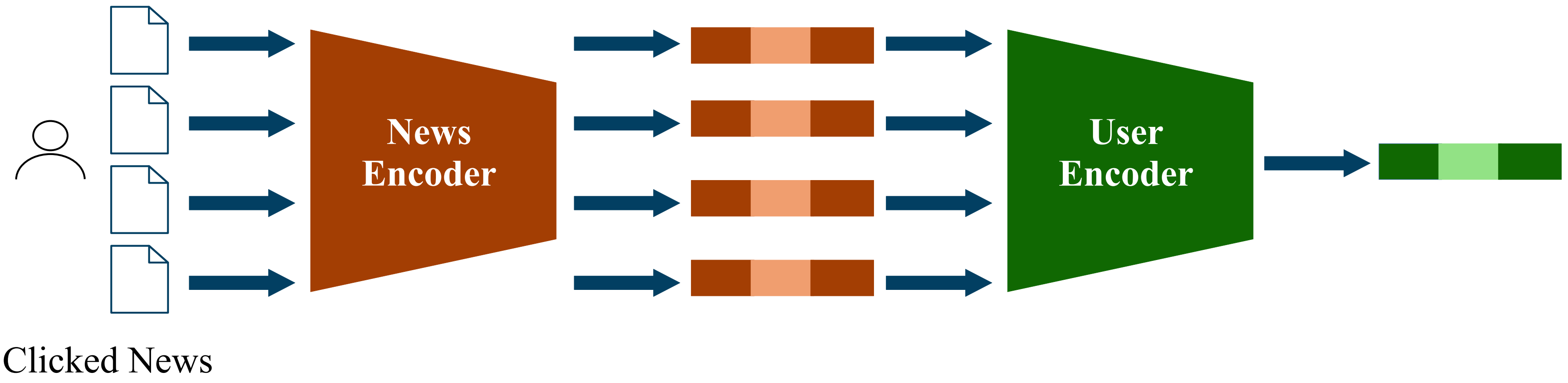


Content-based Neural News Recommenders

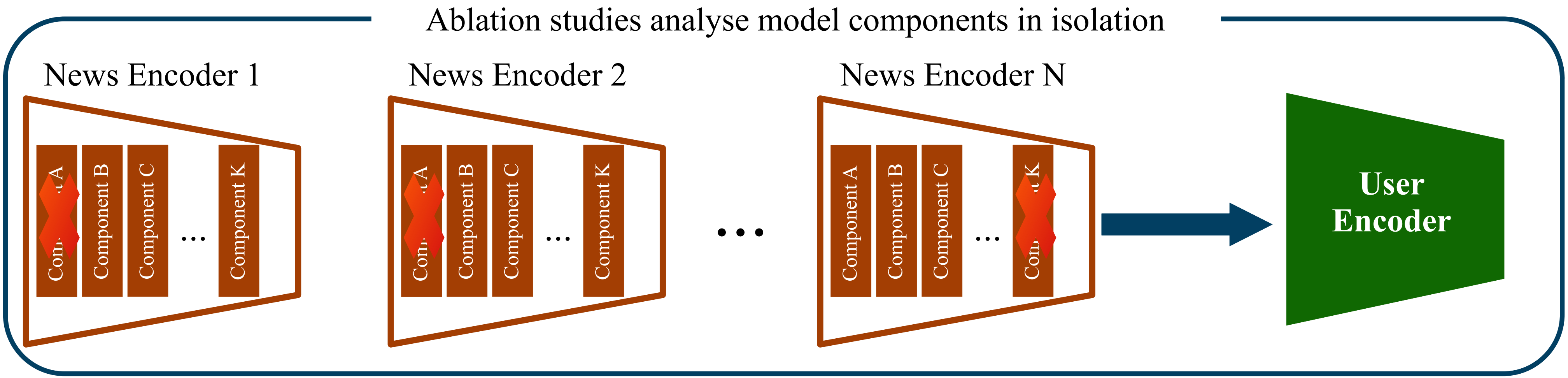
Which article information does the model take into consideration?



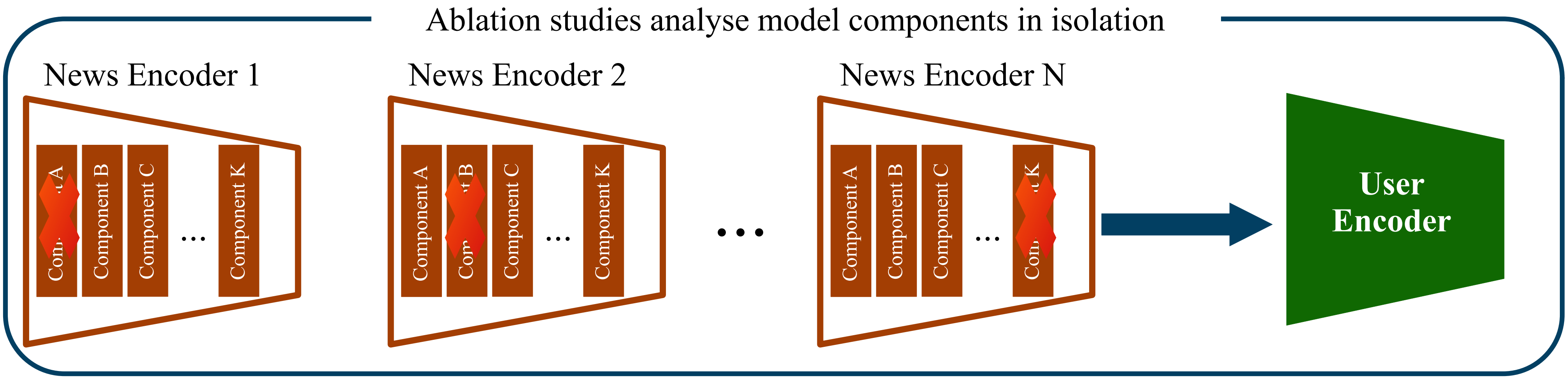
How to model users' preferences based on their histories?



Status Quo


Ablation studies analyse model components in isolation
News Encoder 1
News Encoder 2
News Encoder N
Component B
Component C
Component K
Component A
User Encoder

Status Quo

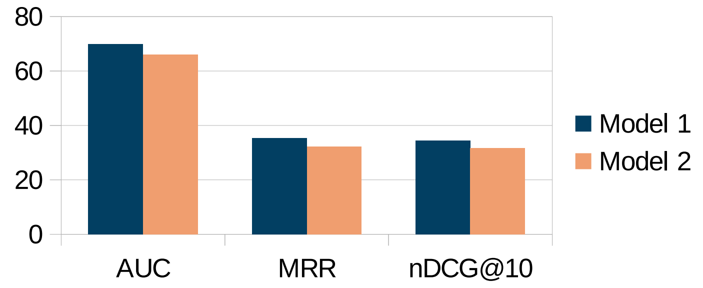


Emerging evidence of **similar performance** despite **varying model complexities**!

Status Quo

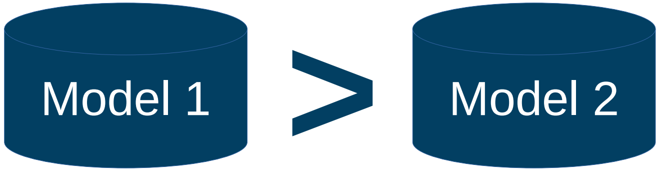
Progress in model design measured wr.t. recommendation performance

Model 1 > Model 2 when

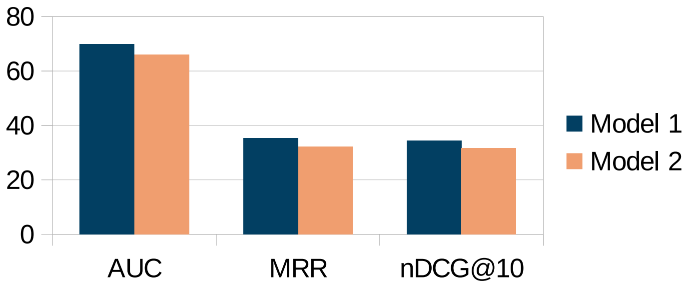


Status Quo

Progress in model design measured wr.t. recommendation performance



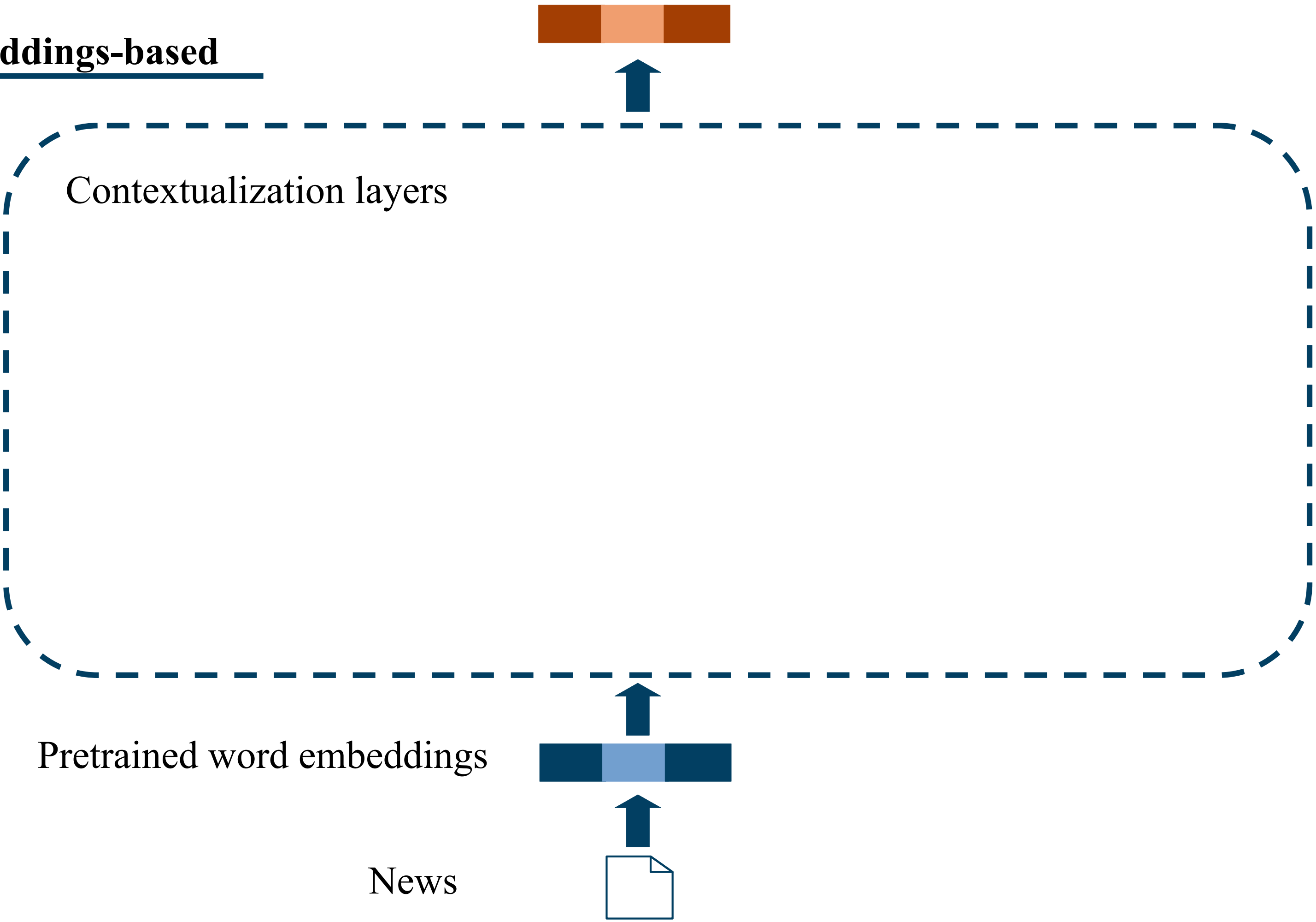
when



One-sided (performance-based) understanding of encoders' behavior can lead to **suboptimal model selection**!

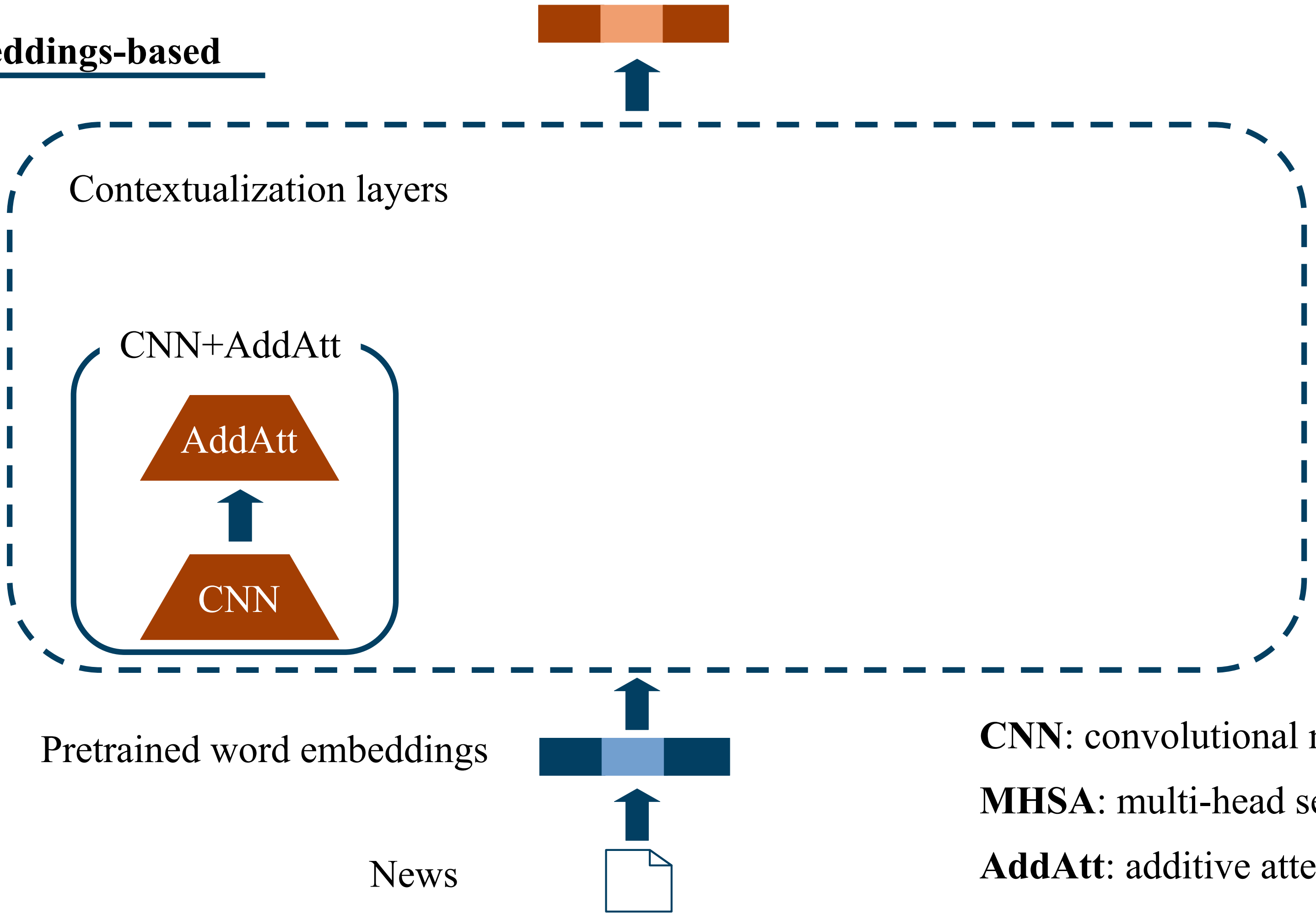
Text Encoder Architectures

(1) Word embeddings-based



Text Encoder Architectures

(1) Word embeddings-based



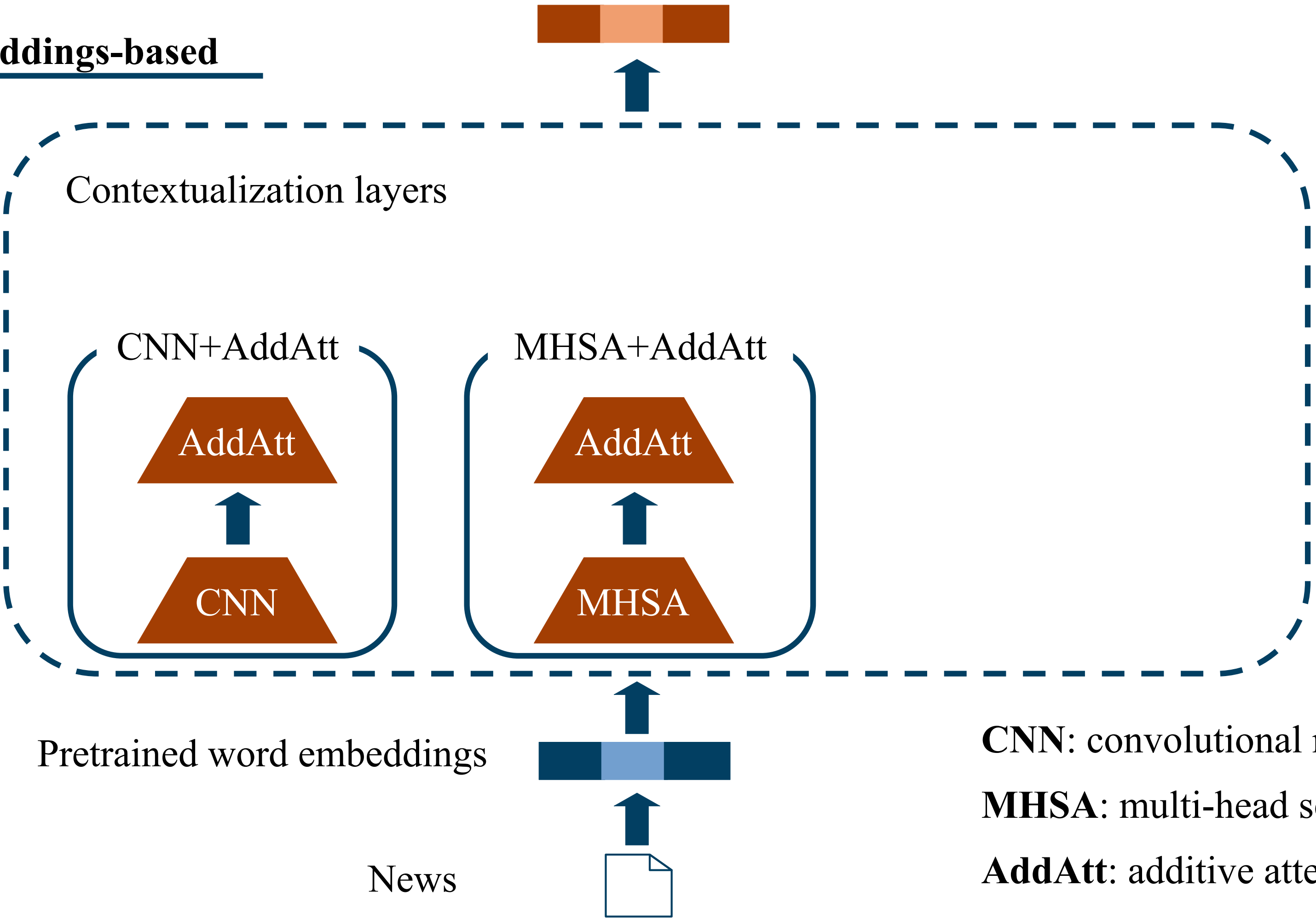
CNN: convolutional neural network

MHSA: multi-head self-attention

AddAtt: additive attention

Text Encoder Architectures

(1) Word embeddings-based



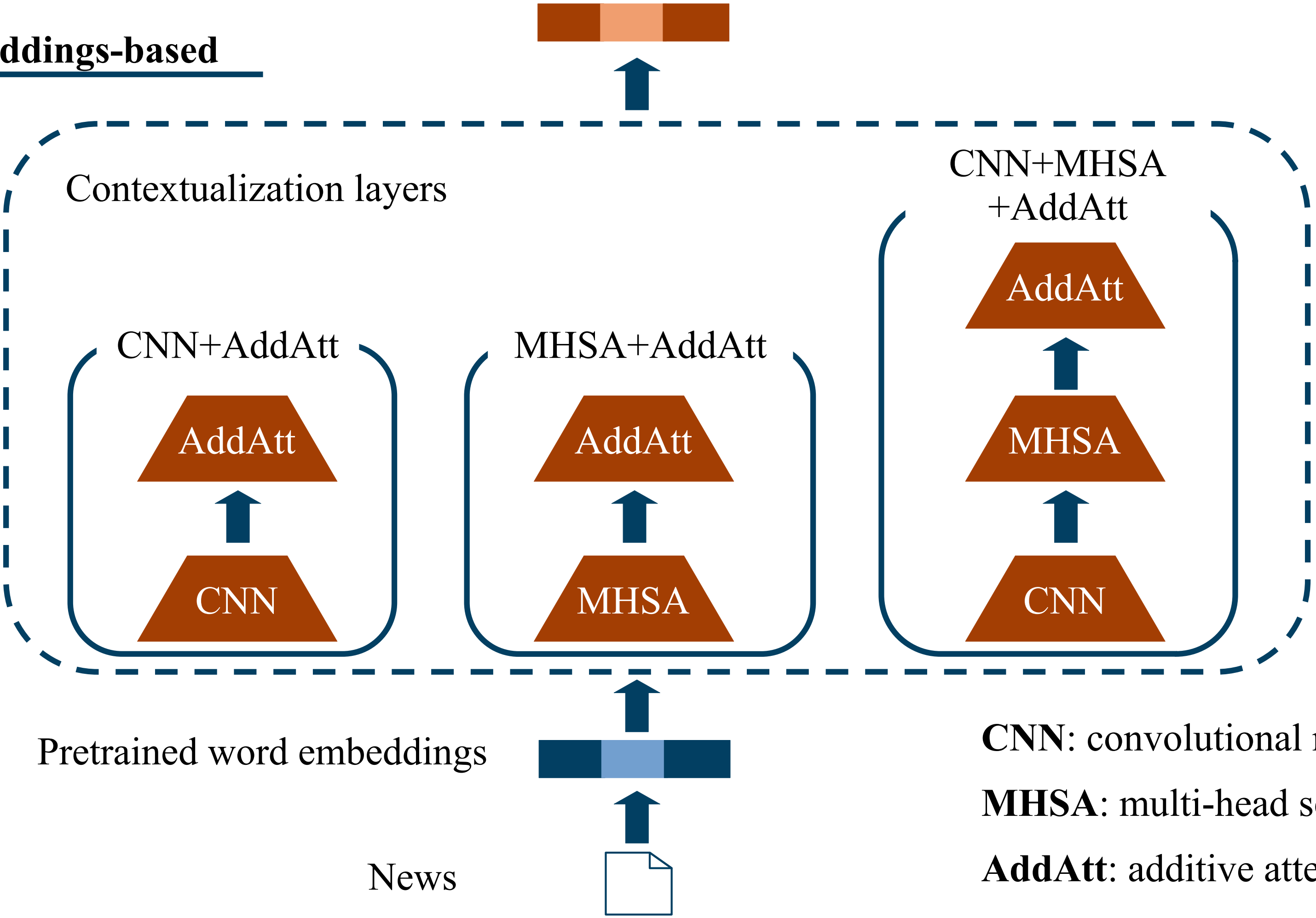
CNN: convolutional neural network

MHSA: multi-head self-attention

AddAtt: additive attention

Text Encoder Architectures

(1) Word embeddings-based



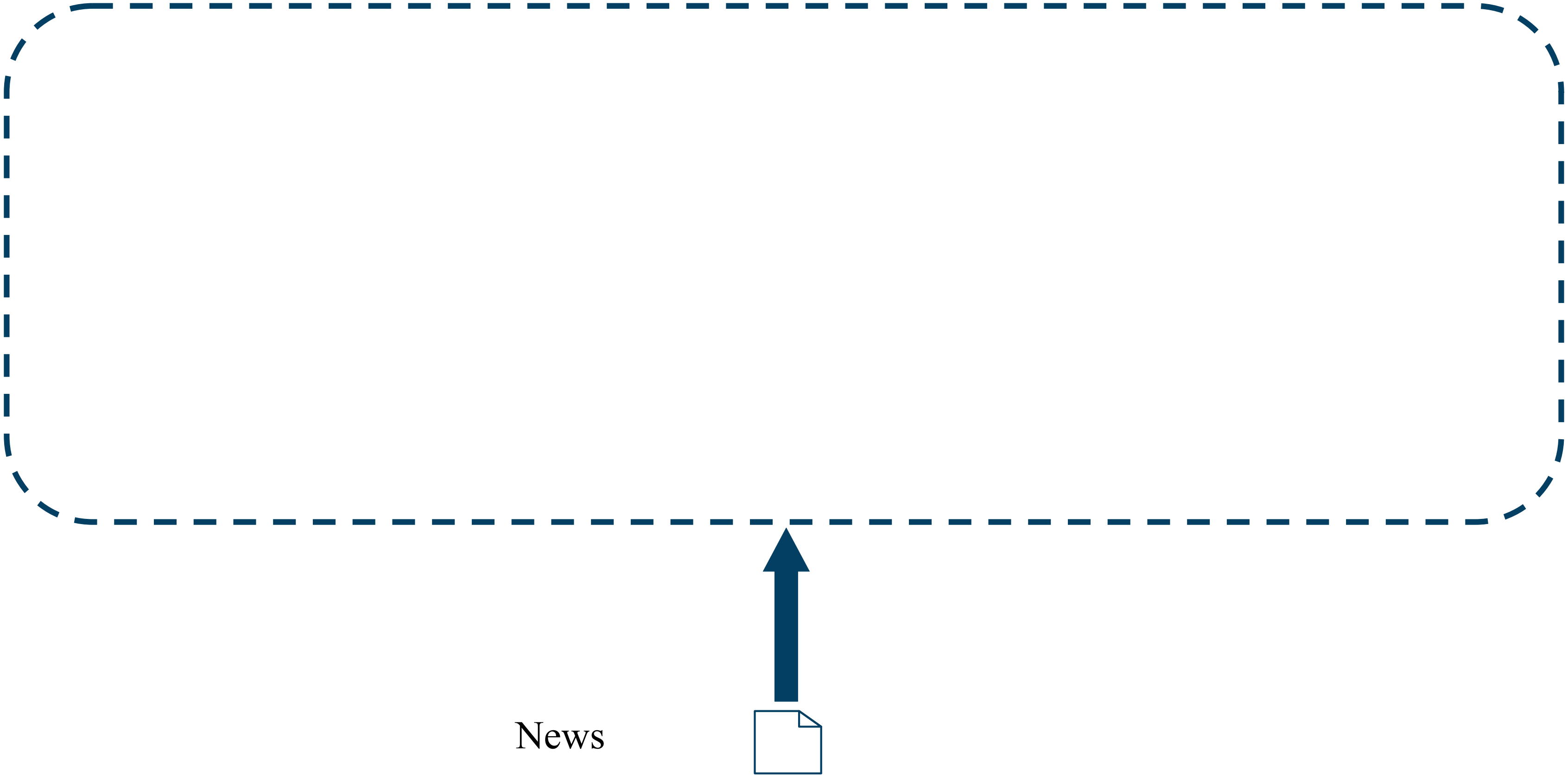
CNN: convolutional neural network

MHSA: multi-head self-attention

AddAtt: additive attention

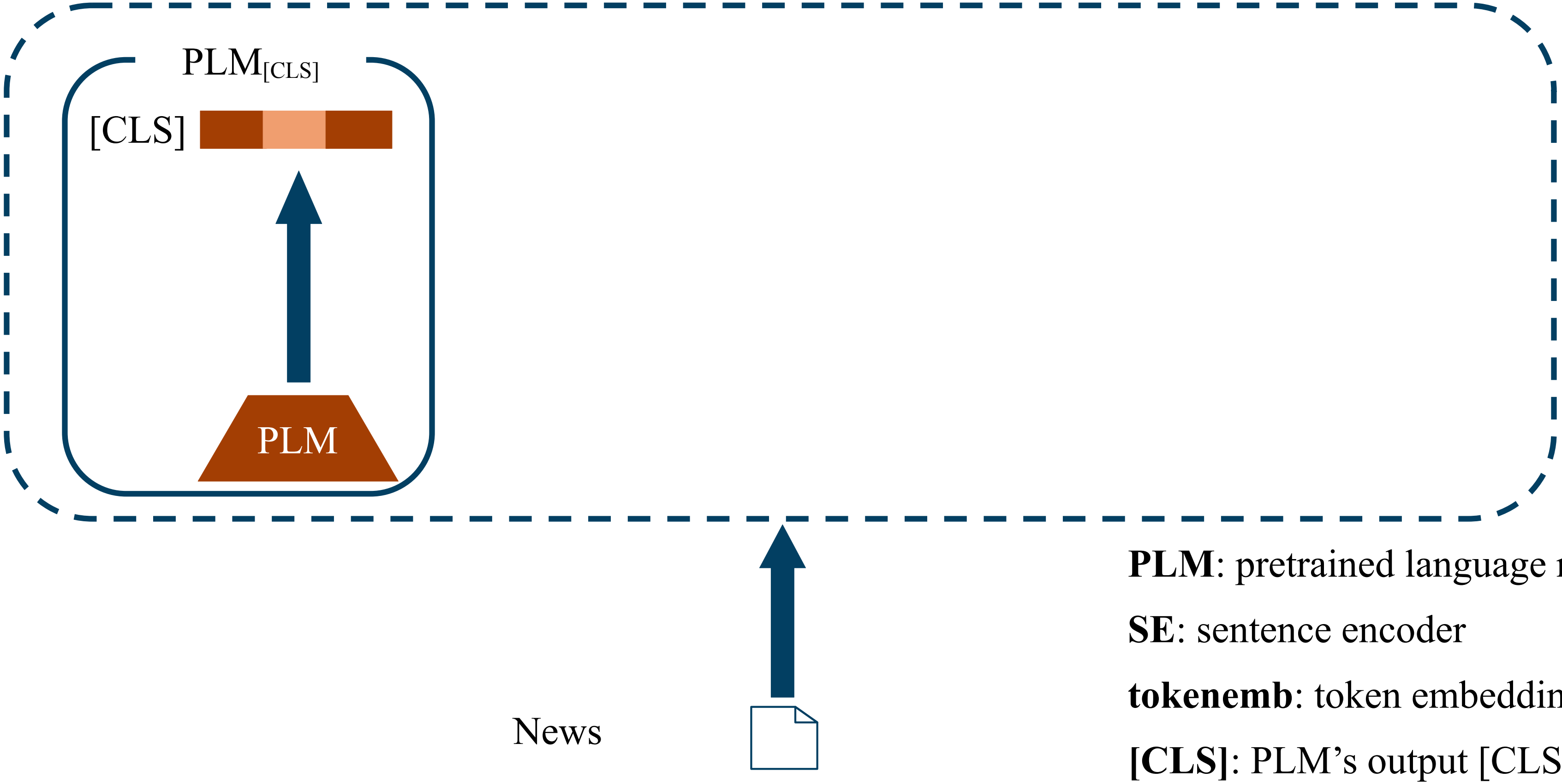
Text Encoder Architectures

(2) Language model-based



Text Encoder Architectures

(2) Language model-based



PLM: pretrained language model

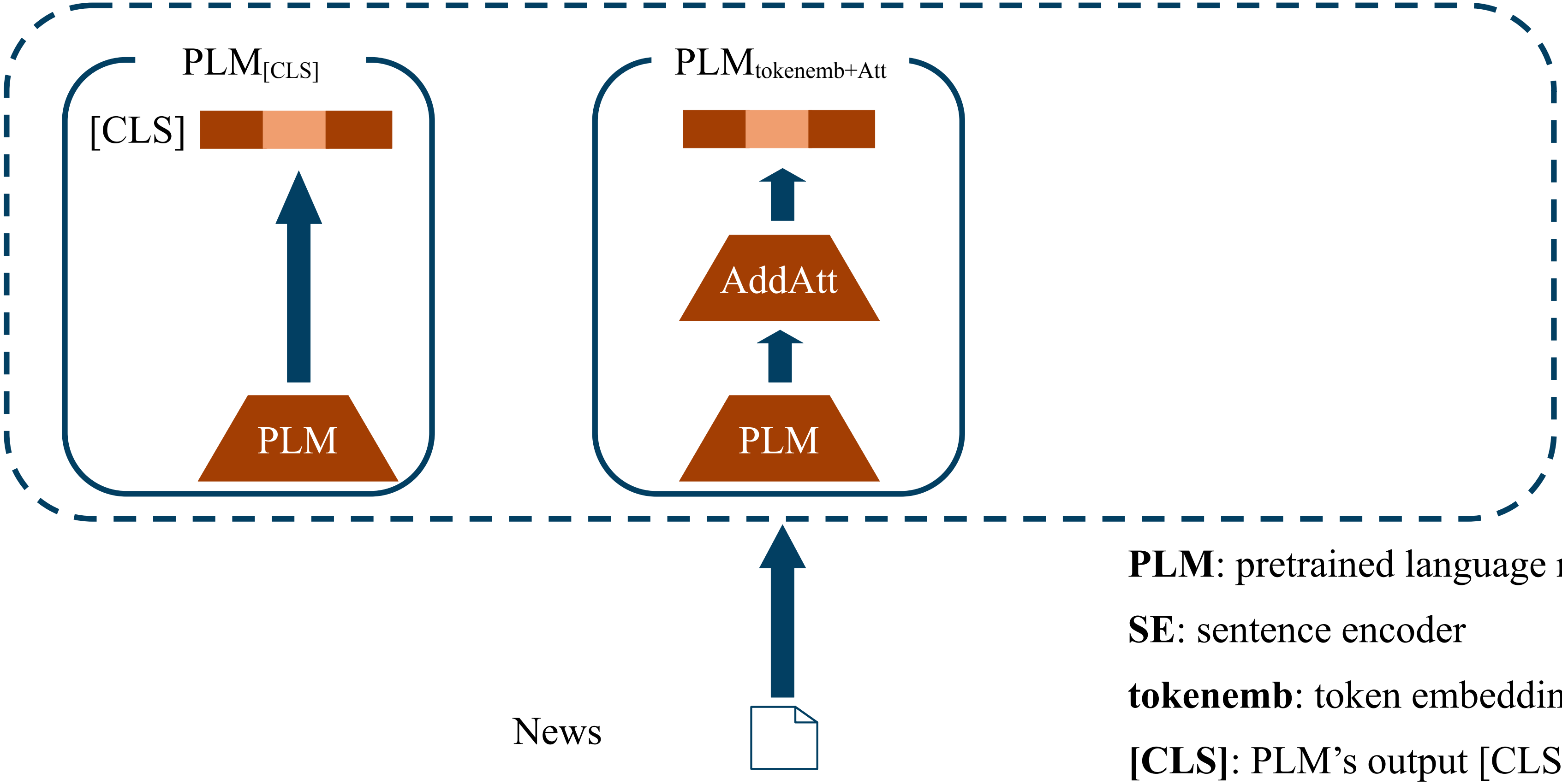
SE: sentence encoder

tokenemb: token embedding

[CLS]: PLM's output [CLS] token

Text Encoder Architectures

(2) Language model-based



PLM: pretrained language model

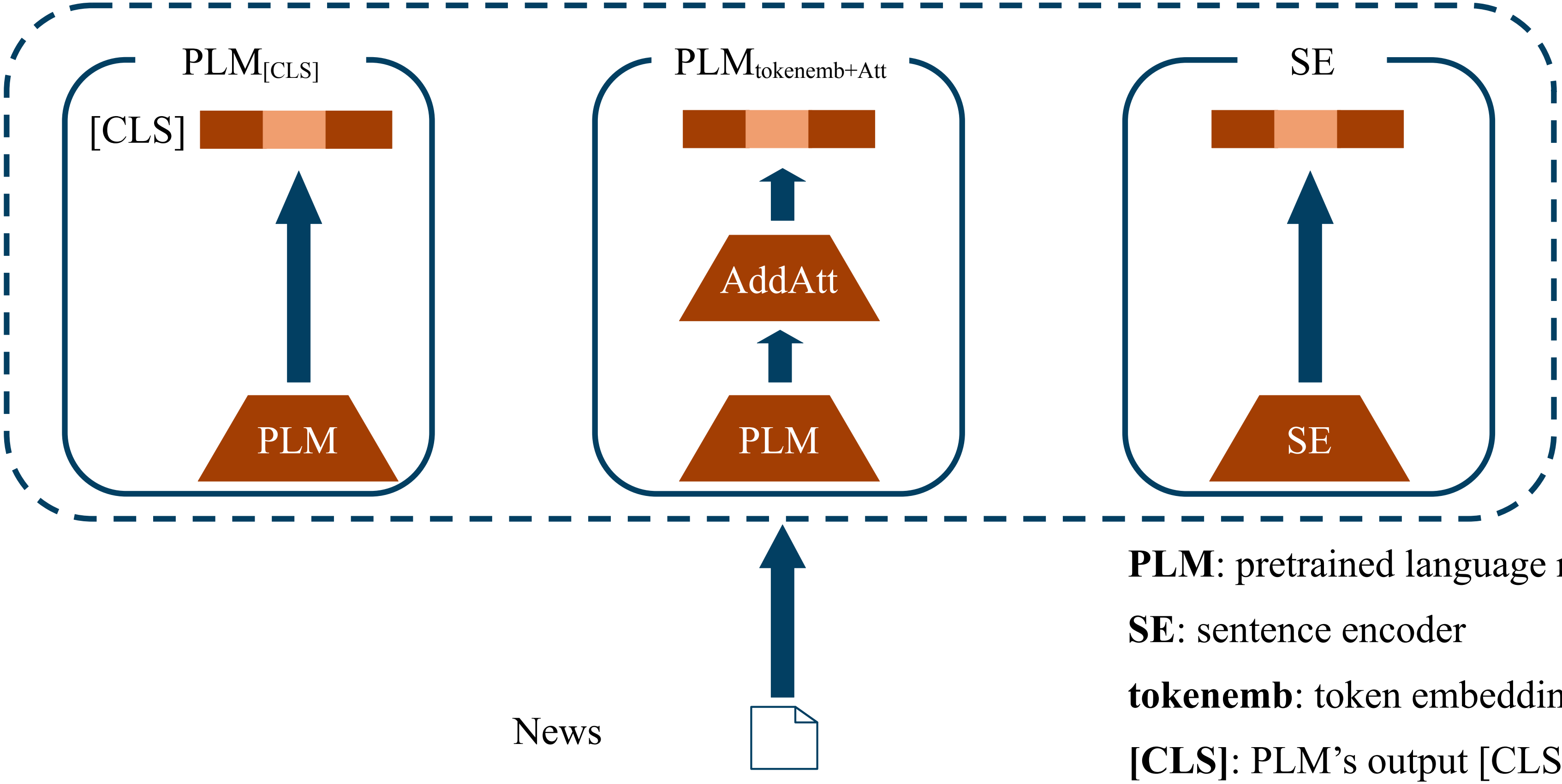
SE: sentence encoder

tokenemb: token embedding

[CLS]: PLM's output [CLS] token

Text Encoder Architectures

(2) Language model-based



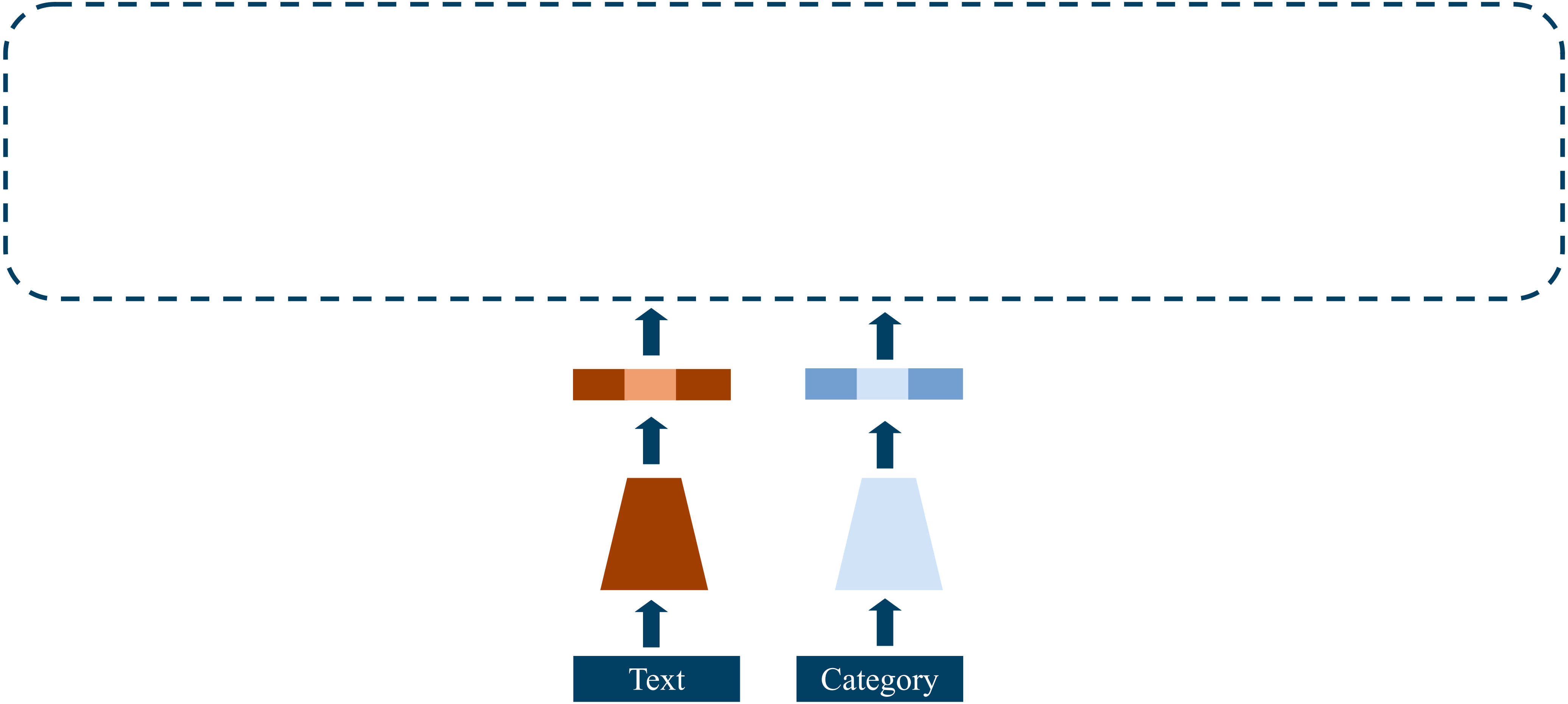
PLM: pretrained language model

SE: sentence encoder

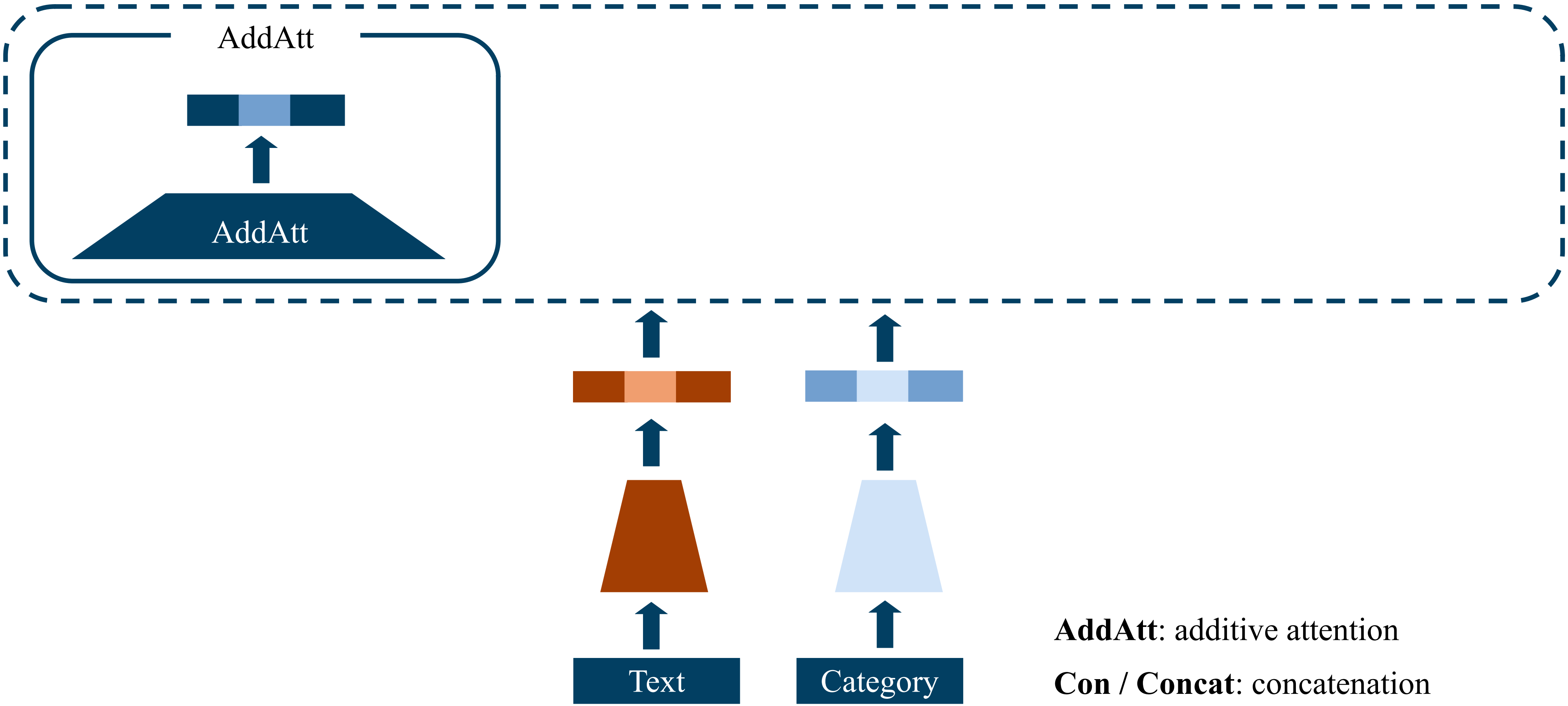
tokenemb: token embedding

[CLS]: PLM's output [CLS] token

Multi-feature Aggregation Strategies



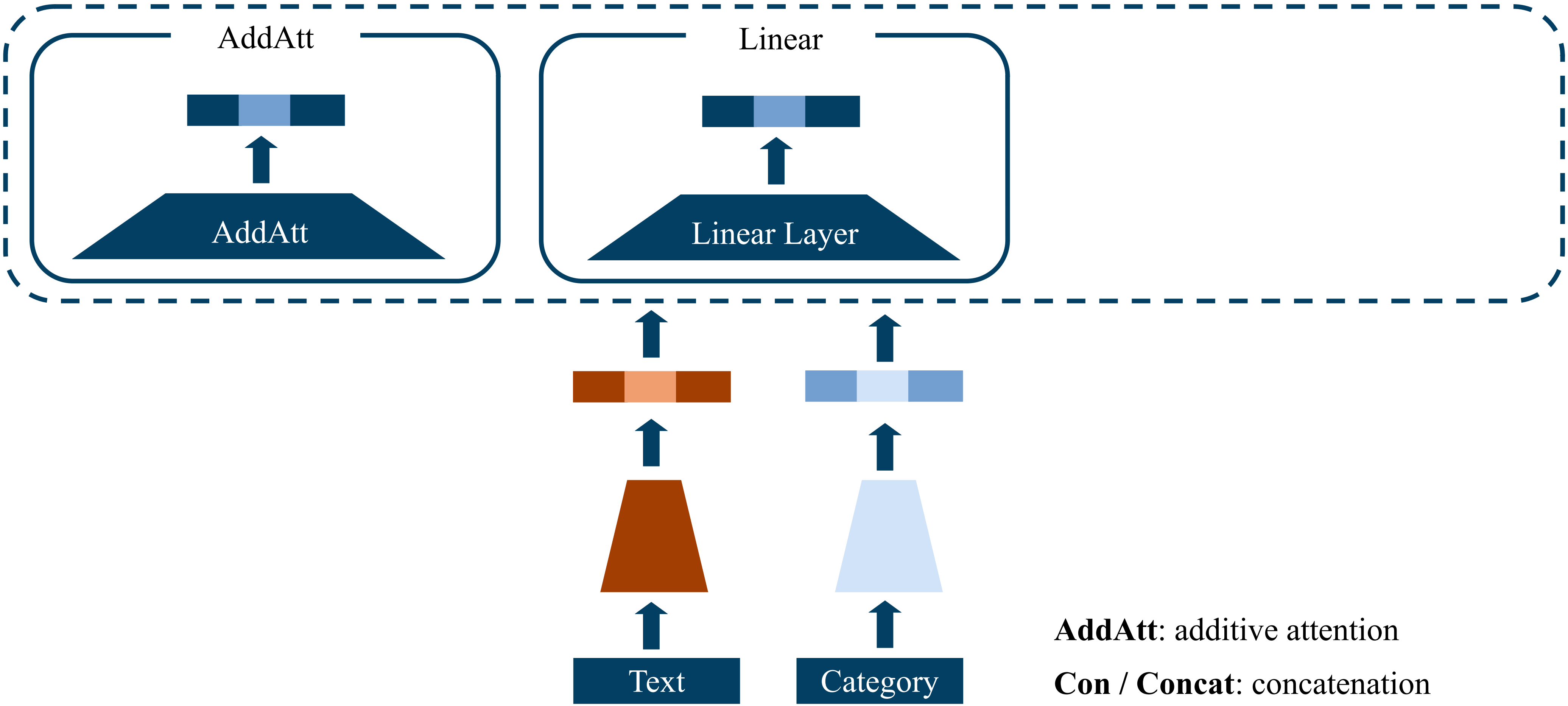
Multi-feature Aggregation Strategies



AddAtt: additive attention

Con / Concat: concatenation

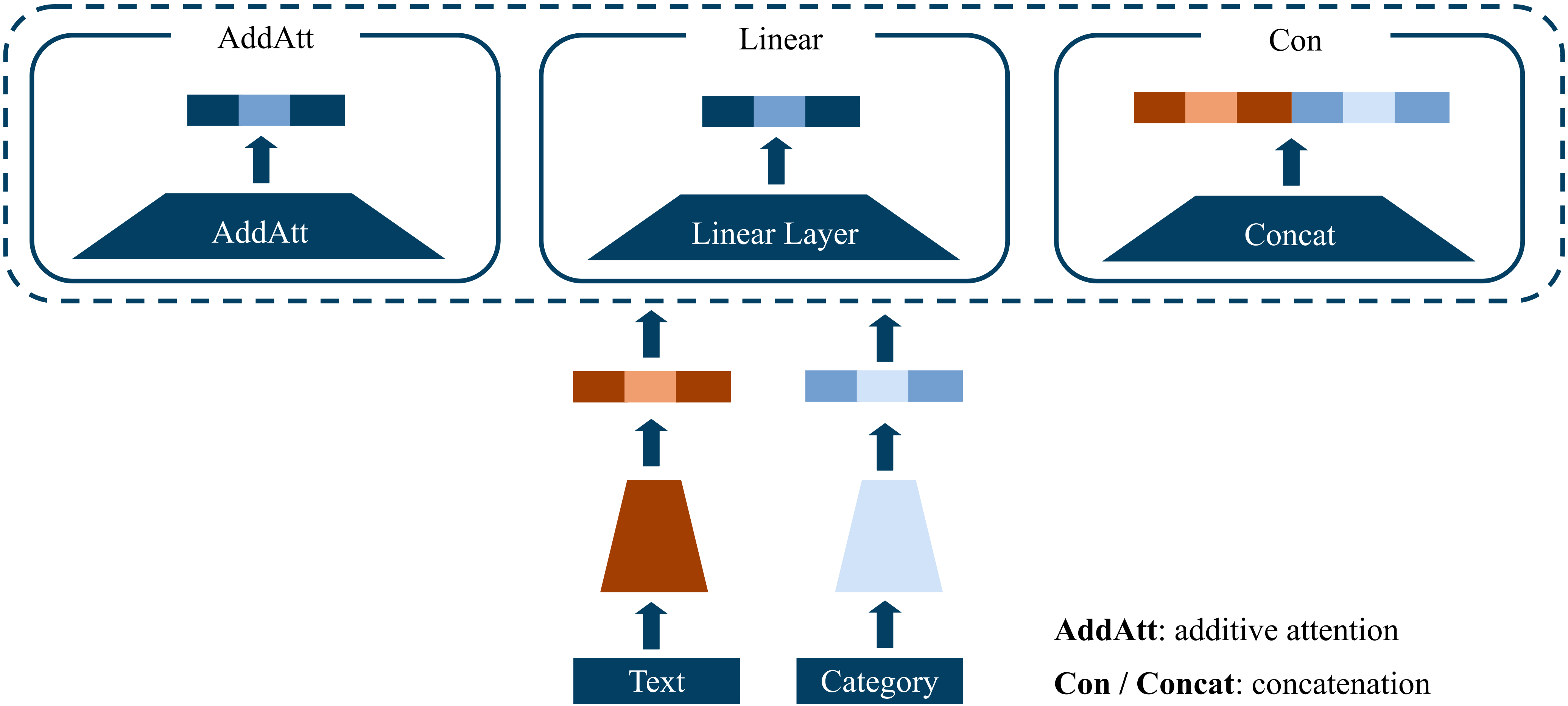
Multi-feature Aggregation Strategies



AddAtt: additive attention

Con / Concat: concatenation

Multi-feature Aggregation Strategies

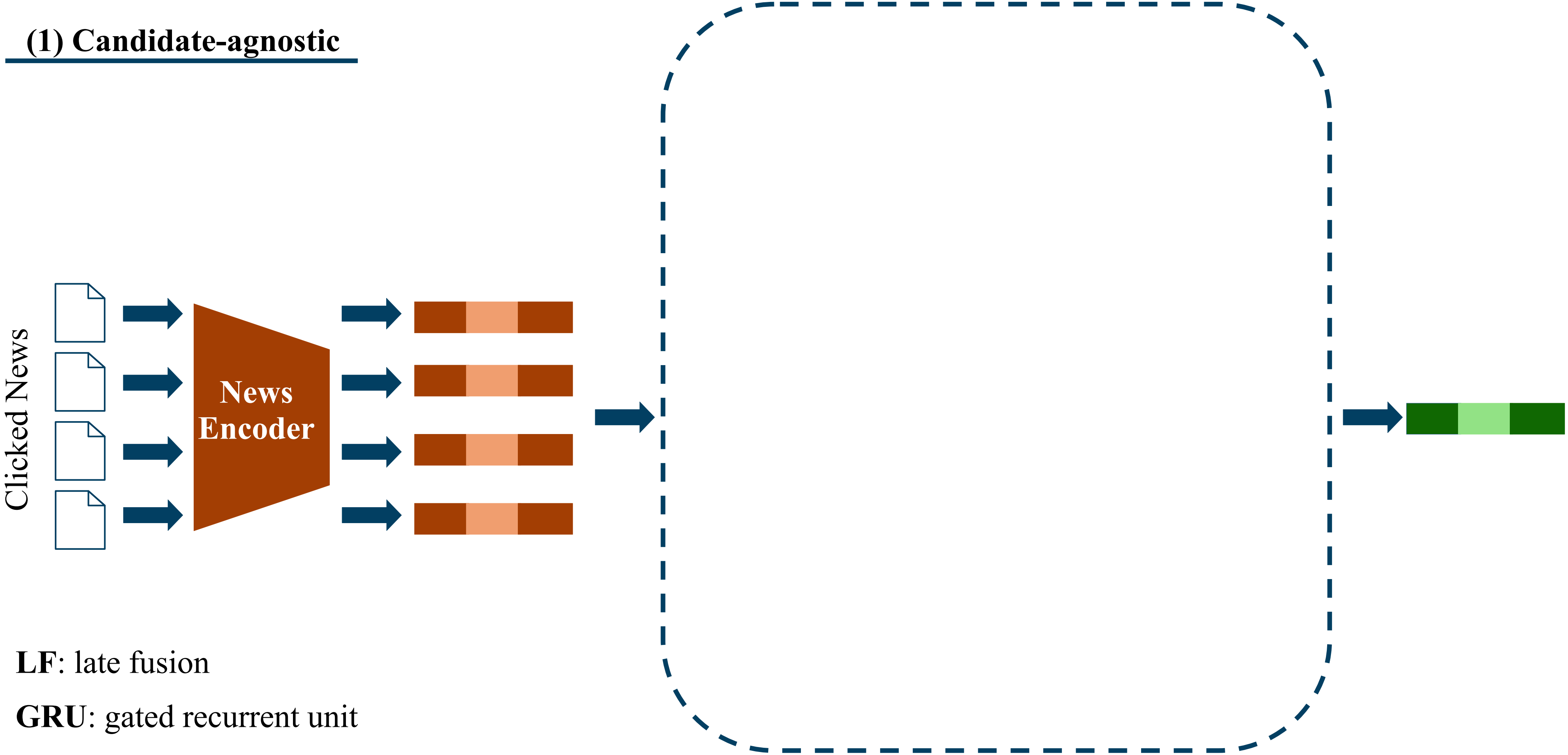


AddAtt: additive attention

Con / Concat: concatenation

User Encoder Architectures

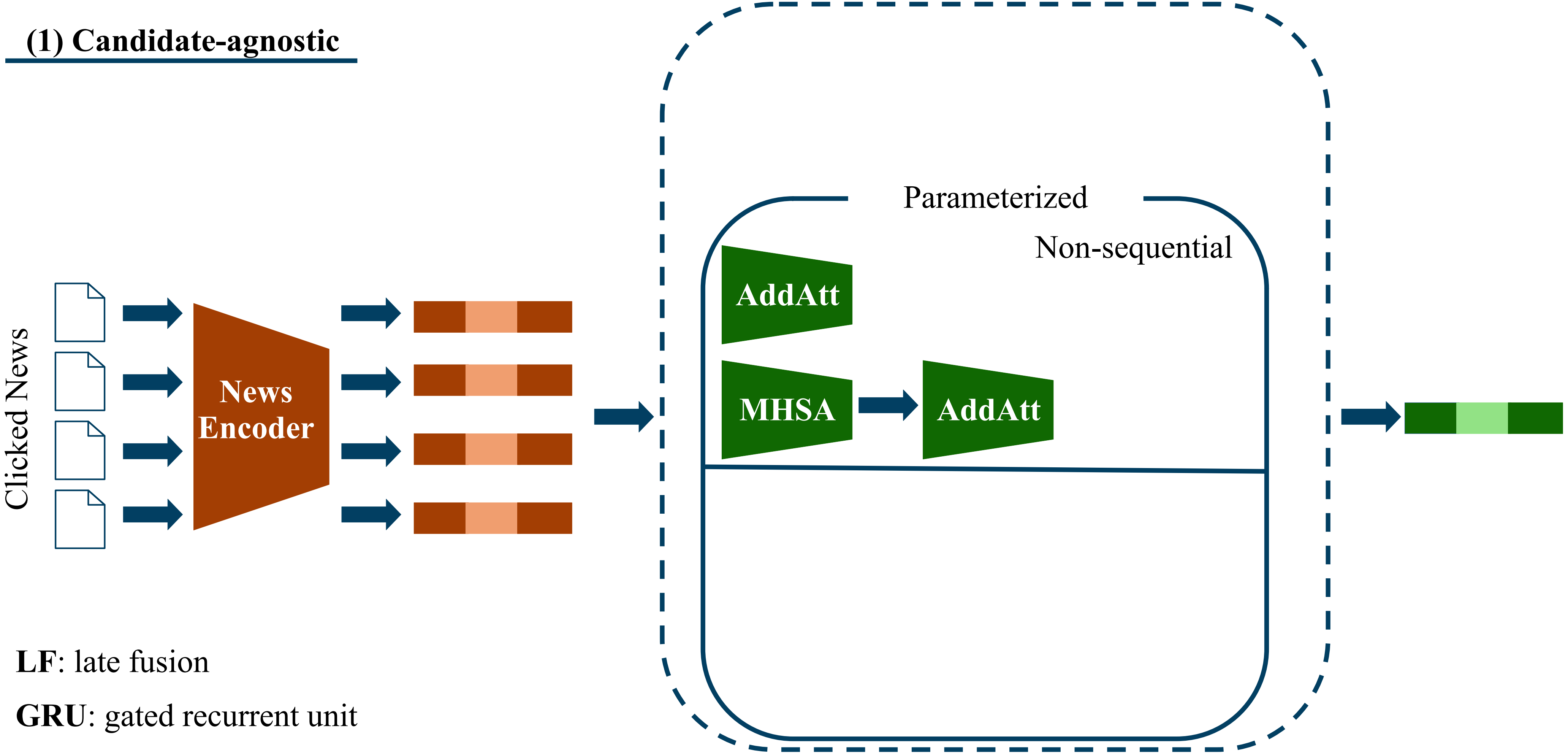
(1) Candidate-agnostic



LF: late fusion

GRU: gated recurrent unit

(1) Candidate-agnostic

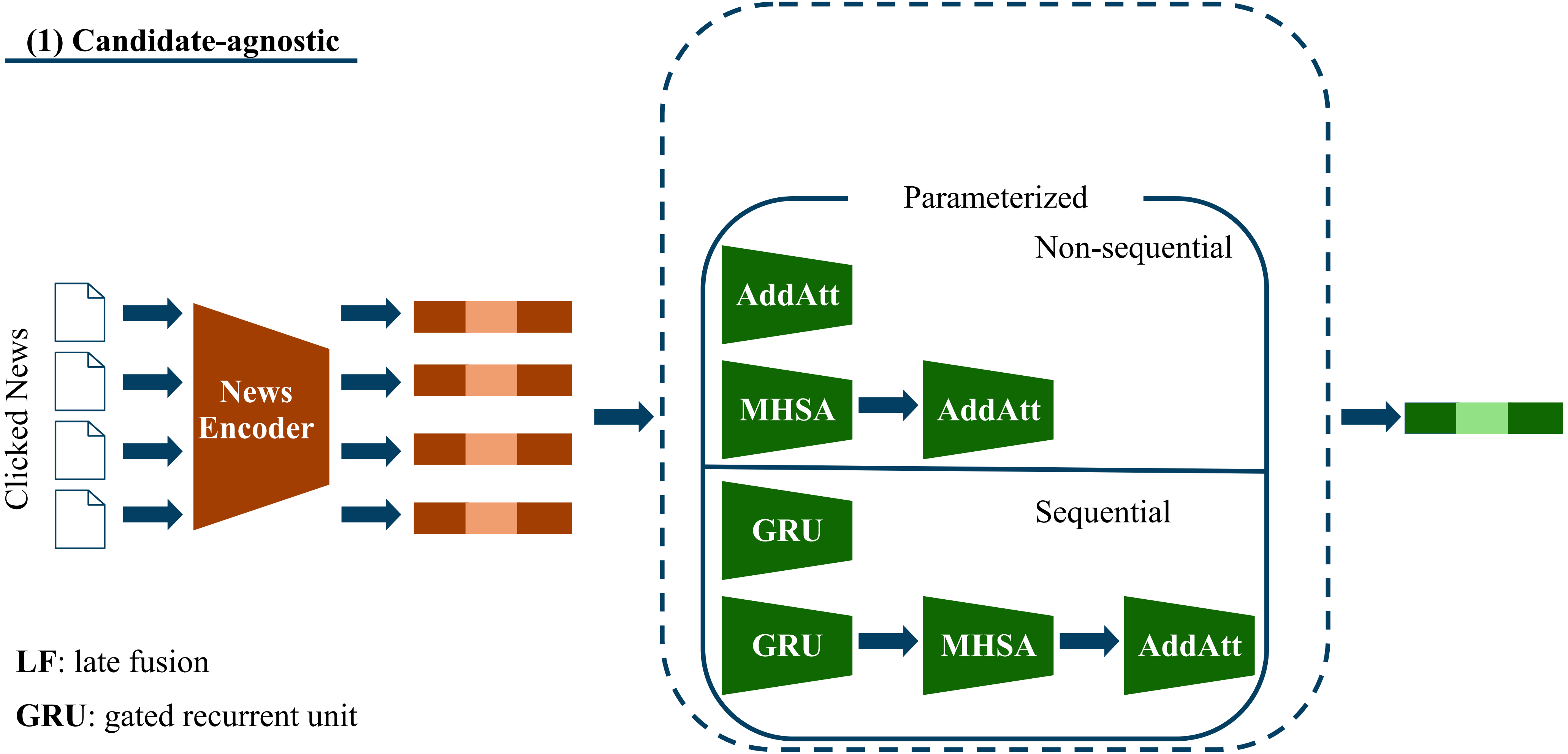


LF: late fusion

GRU: gated recurrent unit

User Encoder Architectures

(1) Candidate-agnostic

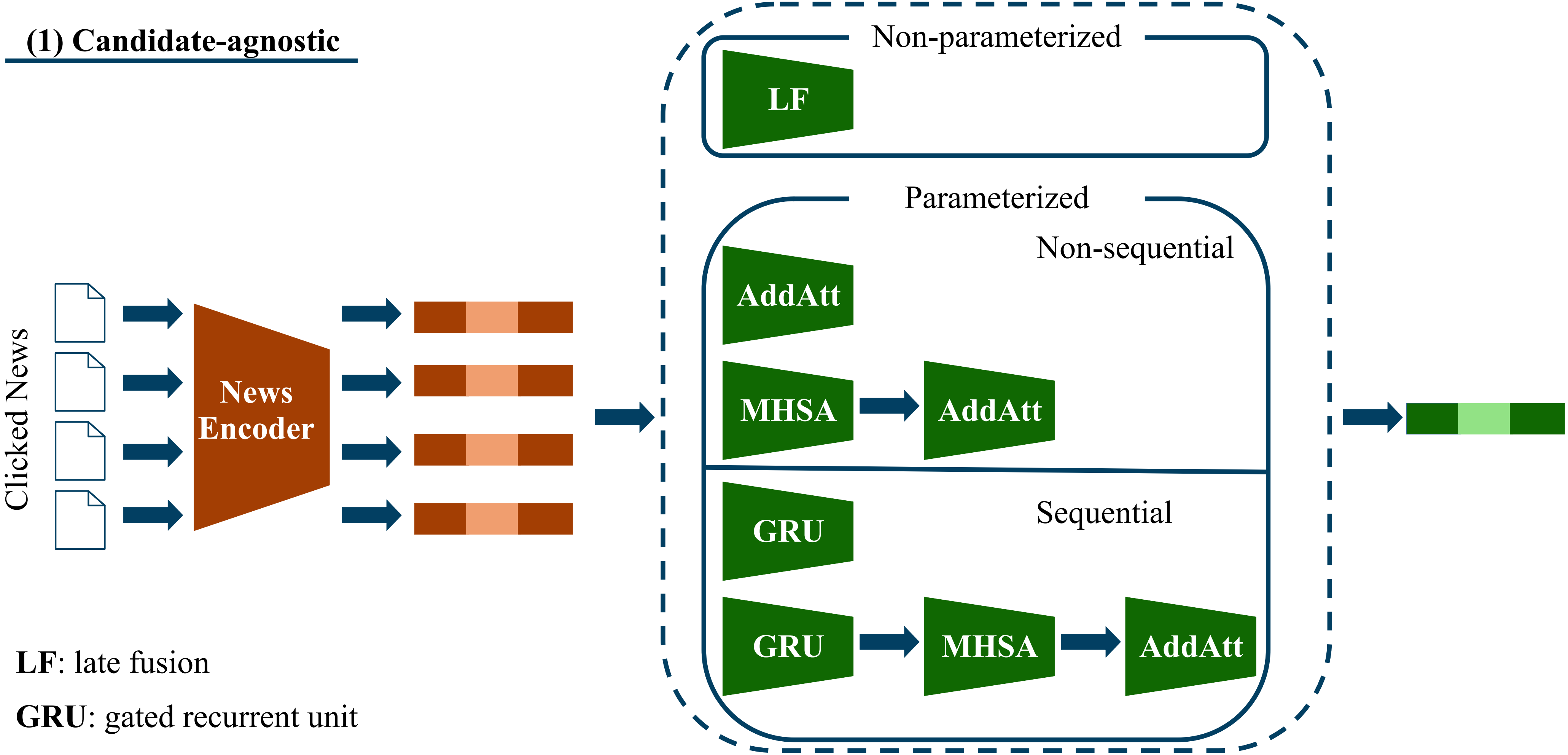


LF: late fusion

GRU: gated recurrent unit

User Encoder Architectures

(1) Candidate-agnostic

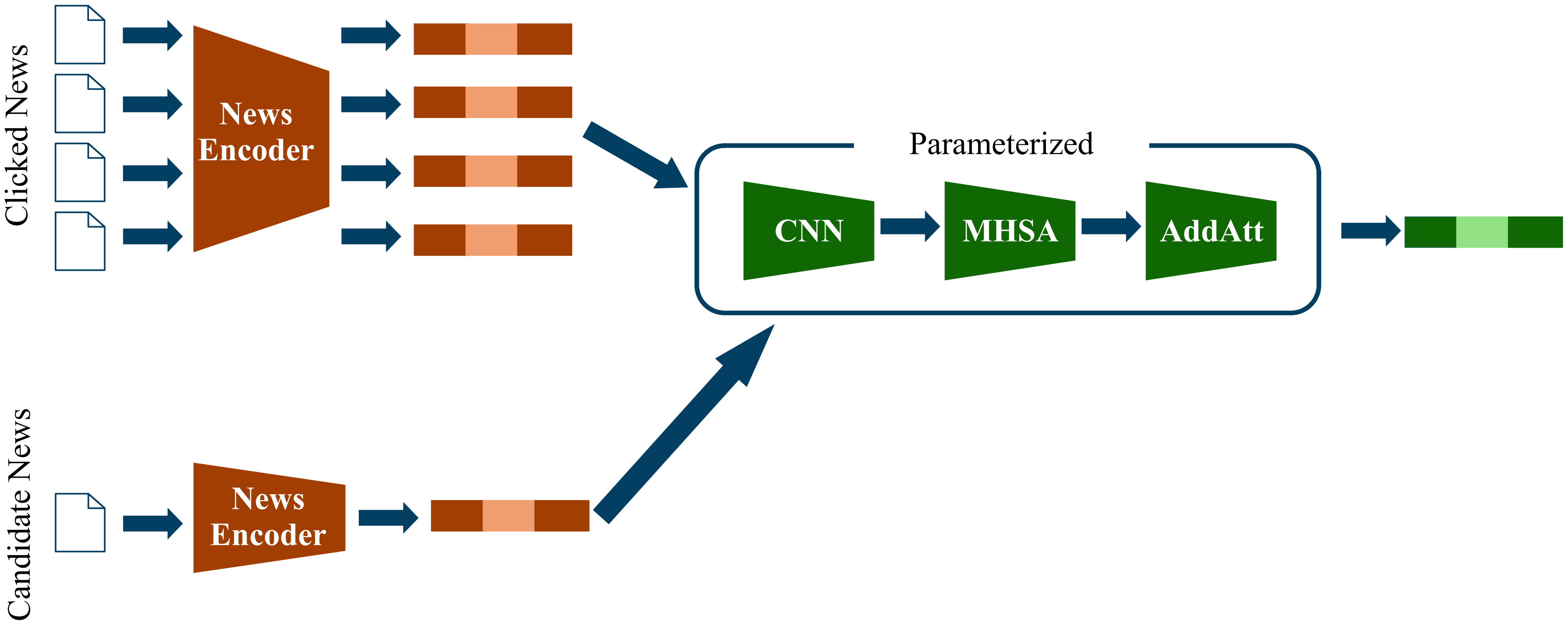


LF: late fusion

GRU: gated recurrent unit

User Encoder Architectures

(2) Candidate-aware



Similarity Evaluation

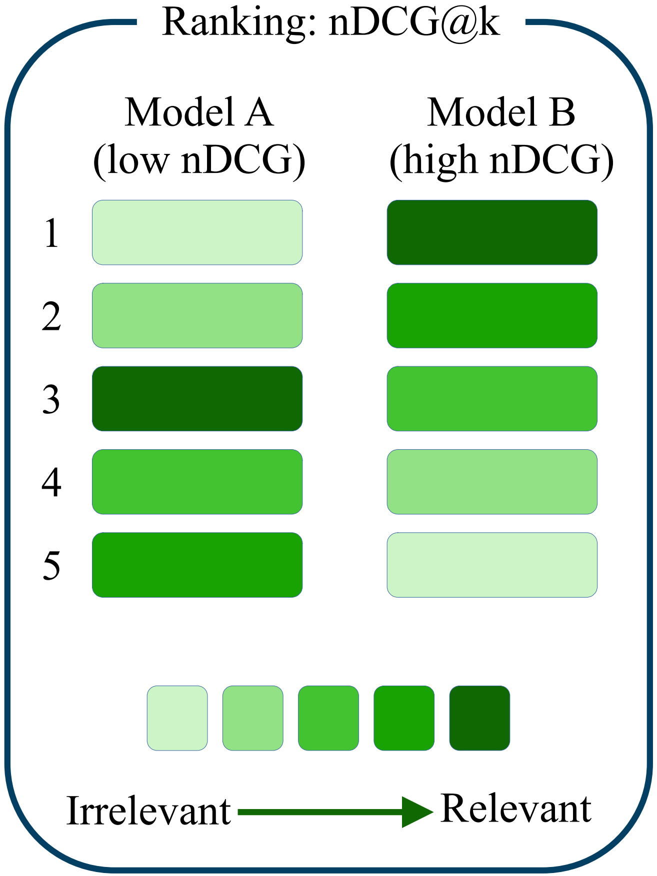
Recommendation Performance

Similarity of Recommendations

Representational Similarity

Similarity Evaluation

Recommendation Performance



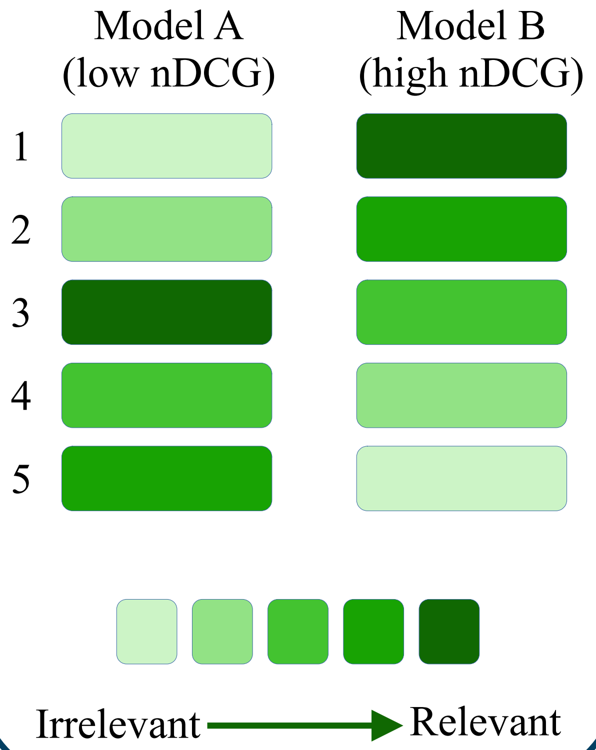
Similarity of Recommendations

Representational Similarity

Similarity Evaluation

Recommendation Performance

Ranking: nDCG@k

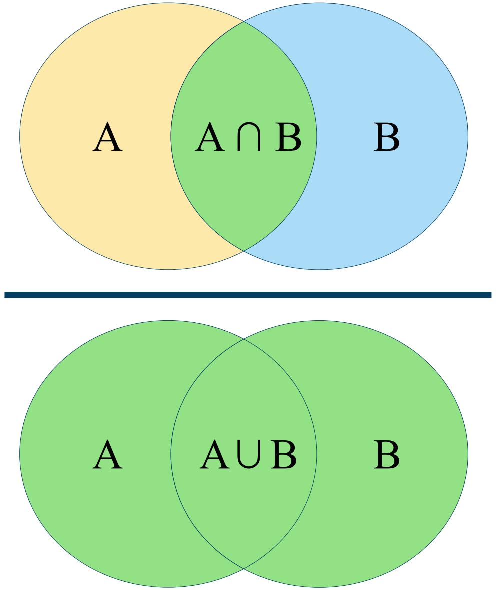


Similarity of Recommendations

Jaccard similarity

A: model A's recommendation list

B: model B's recommendation list

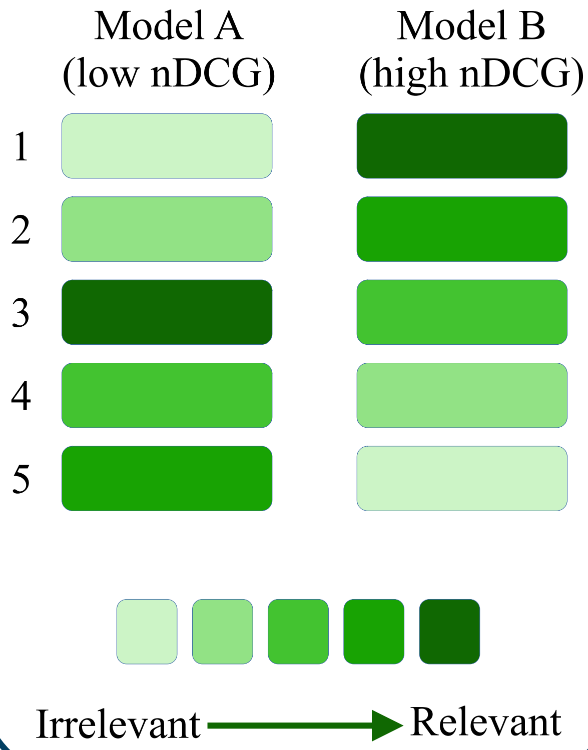


Representational Similarity

Similarity Evaluation

Recommendation Performance

Ranking: nDCG@k

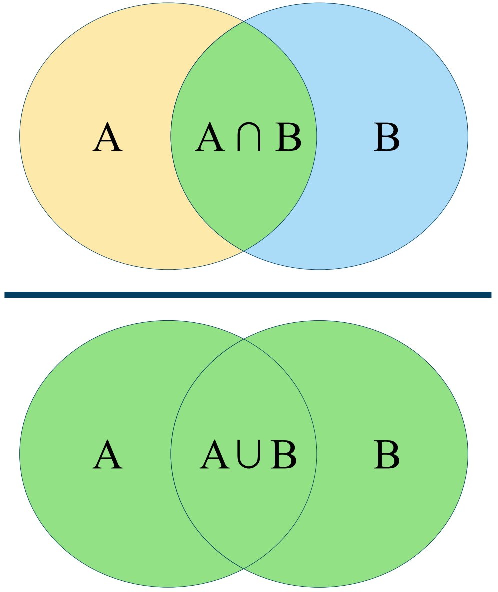


Similarity of Recommendations

Jaccard similarity

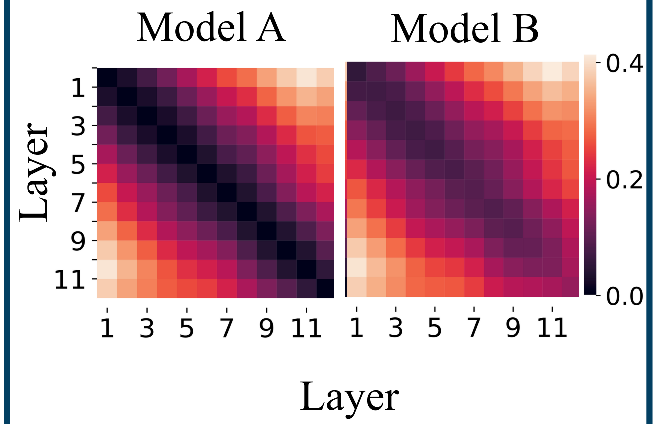
A: model A's recommendation list

B: model B's recommendation list



Representational Similarity

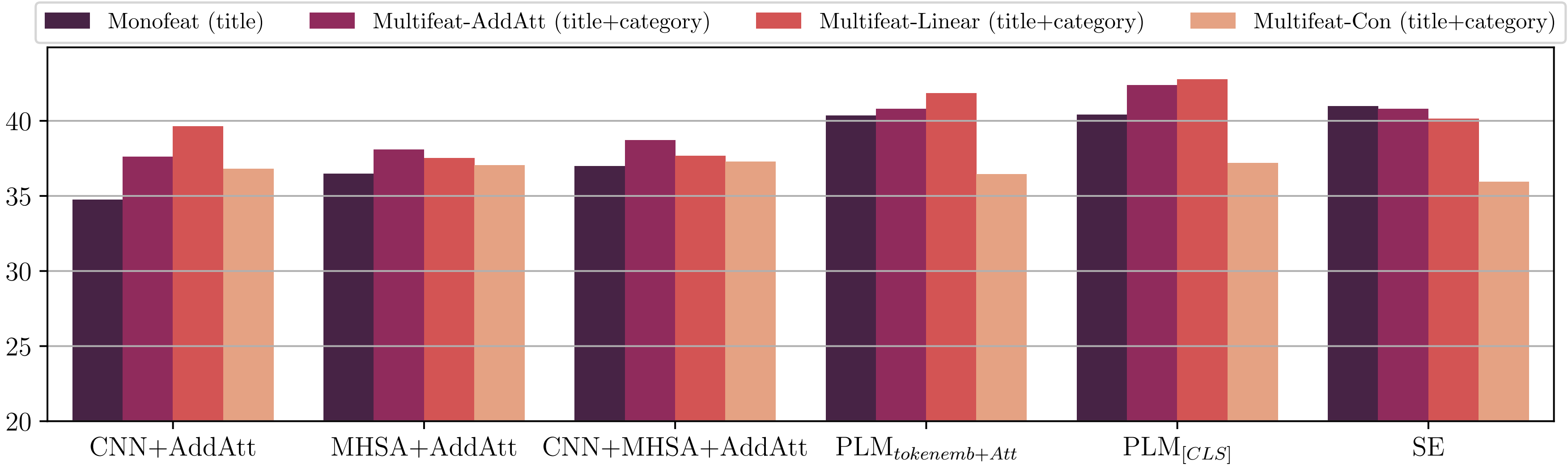
Central Kernel Alignment



Results: News Encoders

Recommendation Performance

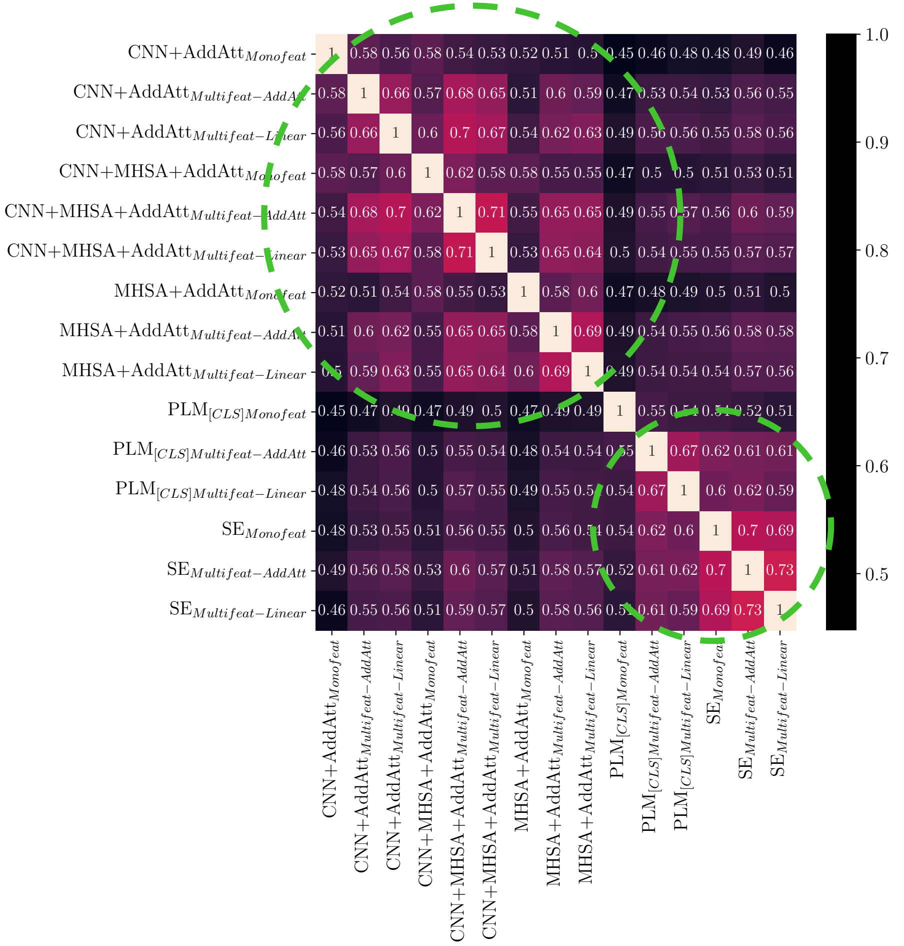
Setup: fixed user encoder = LF



- Similar ranking performance for models using the same family of news encoders
- Categorical information is beneficial only for less contextualized / semantically informed text encoders

Results: News Encoders

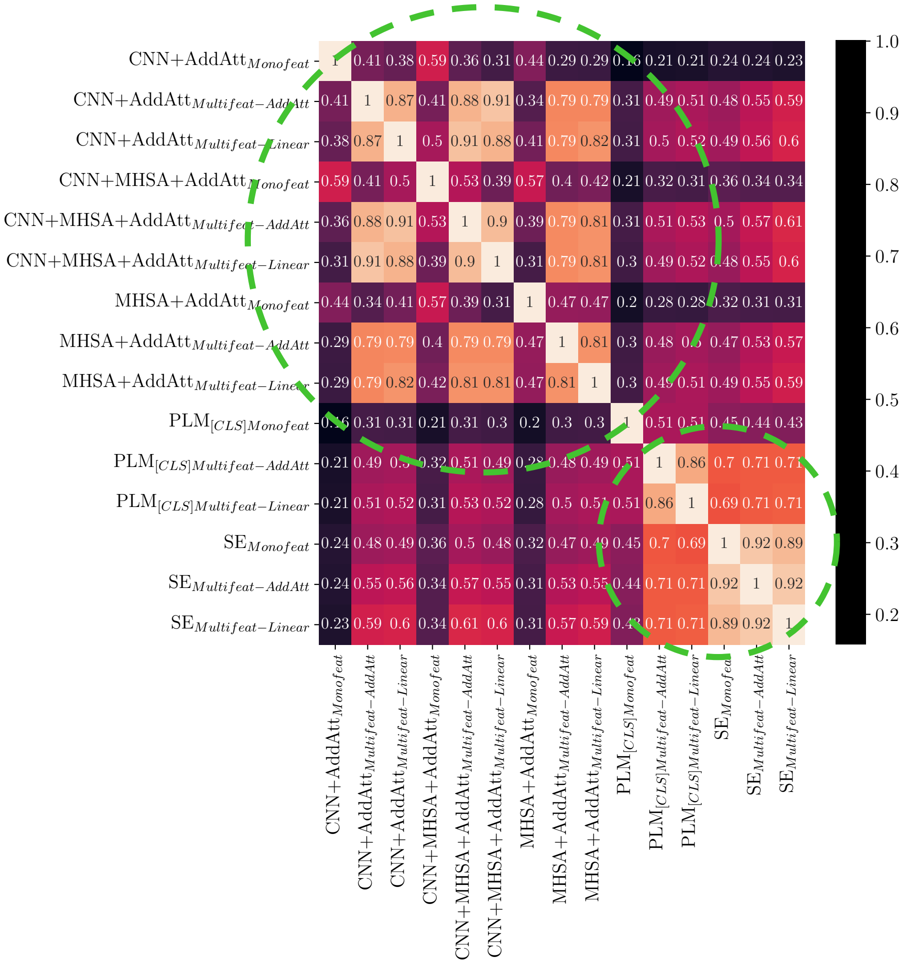
Similarity of Recommendations



- Intra-family news encoders clusters
- Same recommended news in over 70% of the time, regardless of architectural design & complexity

Results: News Encoders

Representational Similarity

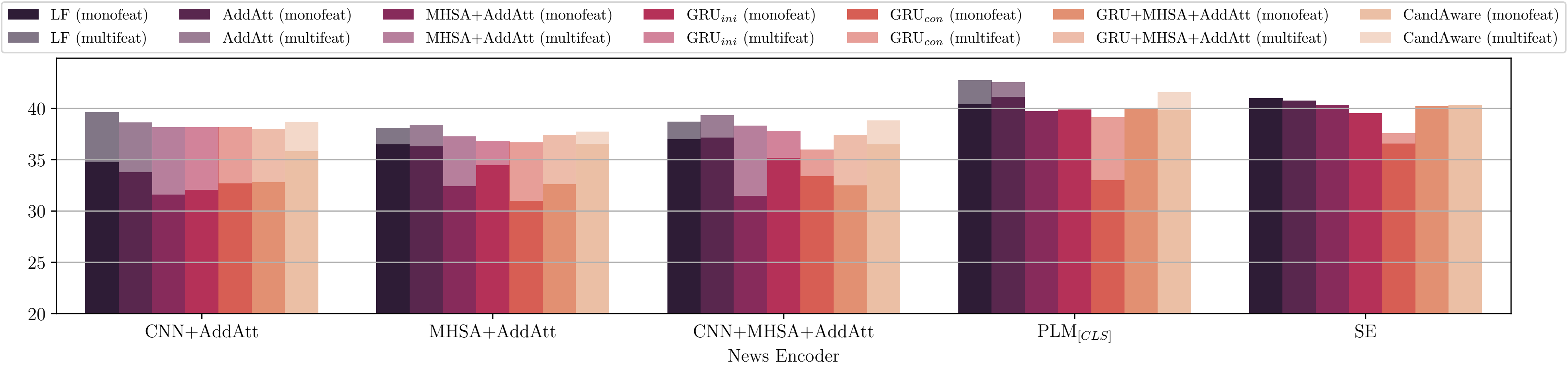


- Similar embeddings of intra-family news encoders
- Similar retrieved items for small *k*, even for models with low representational similarity scores

Results: User Encoders

Recommendation Performance

Setup: fixed news encoder for different user encoders

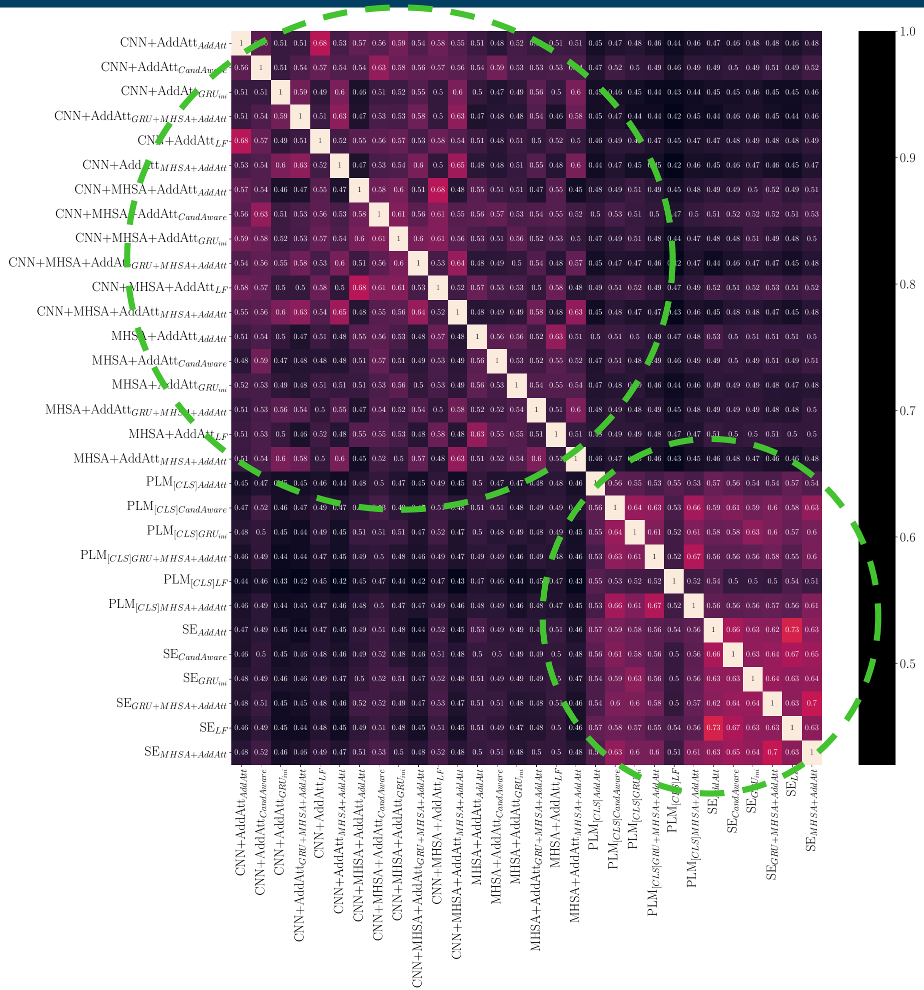


- Simpler encoders (e.g., LF, AddAtt) outperform more complex ones
- Multi-feature inputs close the gap
  - in between inter-family user encoders for the same base news encoder
  - across intra-family user encoders for different underlying news encoders

Results: User Encoders

Similarity of Recommendations

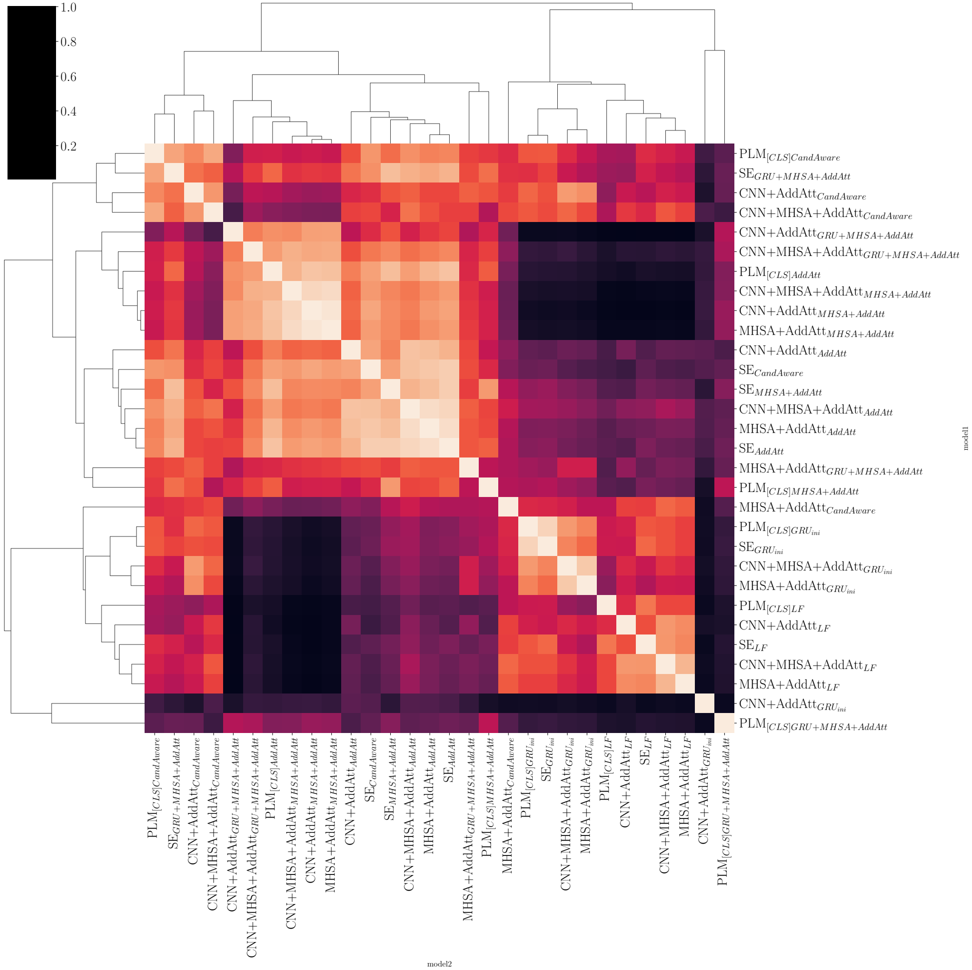
- Models clustered based on the underlying news encoder family, regardless of the user encoder
- Large overlap of recommended news for intra-family user encoders



Results: User Encoders

Representational Similarity

- Architecturally comparable families of user encoders dictate the similarity of embeddings, regardless of the news encoder
- Differences in representational similarity not directly correlated with more dissimilar recommendations



Main Takeaways

1

Semantic Richness is Key

- **News encoding** should focus more on using or adapting **semantically informed, contextualized language models**

Main Takeaways

1 **Semantic Richness is Key**

2 **User Encoders Can be Considerably Simplified**

- **Simpler architectures** are more lightweight, equally effective user encoder alternatives
- **User modeling** should focus also on (i) **collecting** richer, more accurate **user feedback** & (ii) news **consumption motivations**

Main Takeaways

1 Semantic Richness is Key

2 User Encoders Can be Considerably Simplified

3 More Rigorous Evaluation is Needed for Better Model Selection

- Ablations & evaluations should go **beyond perfomance-based evaluation** & consider the **broader architectural context**

Conclusion

1 Semantic Richness is Key

2 User Encoders Can be Considerably Simplified

3 More Rigorous Evaluation is Needed for Better Model Selection

NewsRecLib

Contact