Universitat Politècnica de València
Master in Artificial Intelligence, Pattern Recognition and Digital Imaging

2023-2024

MACHINE TRANSLATION

6. External Knowledge in Neural Machine Translation

Francisco Casacuberta

fcn@prhlt.upv.es

October 24, 2023

Index

Index

Why external linguistic knowledge?

- YES?
  - There are many linguistic knowledge available.
  - The bilingual training data can be better exploited.
- NOT?
  - Many linguistic knowledge is hard to formalize.
  - The generation of new linguistic knowledge requires great human effort.

- Other knowledge apart from linguistics? i.e. from statistical models?
- Can linguistic knowledge be extracted from a neural model?

External knowledge in NMT

- Formulation:

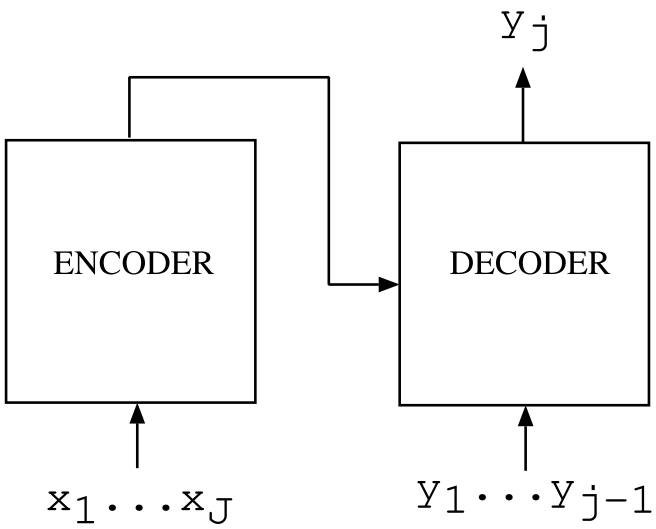
$$\begin{aligned}\hat{y}_1^{\hat{I}} &= \underset{I, y_1^I}{\text{argmax}}\, p(y_1^I \mid x_1^J, K(x_1^J)) \\ &= \underset{I, y_1^I}{\text{argmax}} \prod_{i=1}^{I} p(y_i \mid y_1^{i-1}, x_1^J, K(x_1^J))\end{aligned}$$

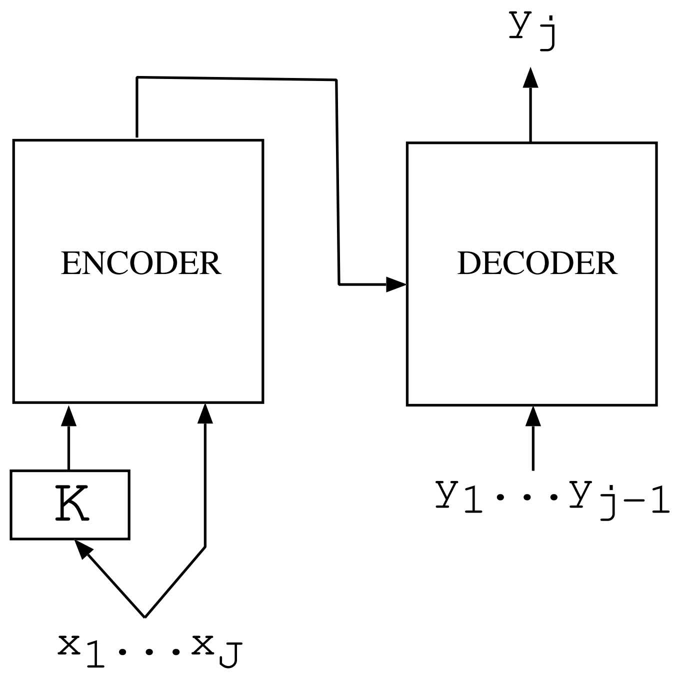
  $K(x_1^J)$ represents external information from the source sentence x_1^J and tfrom the generated target prefix y_1^{i-1}.

- Knowledge sources:
  - Statistical dictionaries from statistical translation models.
  - Syntax-aware of the source sentence.

External knowledge in NMT

Usual approach: A multimodal-like approach.





Index

Statistical dictionaries and NMT [Chen TASLP 2022]

- Given a prior statistical lexicon or dictionary: $l(\mathrm{y} \mid \mathrm{x})$, for x $\in \Sigma_S$ and y $\in \Sigma_T$, and a source sentence x_1^J.

- For each source word x_j, $1 \leq j \leq J$ a vector is built $[\mathrm{y}_j^1, \ldots, \mathrm{y}_j^L]$ with the top L target sentences according to $l(\mathrm{y} \mid \mathrm{x})$: $l(\mathrm{y}_j^1 \mid \mathrm{x}_j) \geq \cdots \geq l(\mathrm{y}_j^L \mid \mathrm{x}_j)$. The J vectors form a matrix $\mathbf{M}$ where the row j is $[\mathrm{y}_j^1, \ldots, \mathrm{y}_j^L]$.

- Two attention mechanism are used: One to take into account the source sentence as in the standard transformer, and the second one to take into account projections of $\mathbf{M}$ as key and values and a projection of the source sentence as the query.

- On tasks WMT14 En-De and WMT17 Zh-En small but significant improvements were achieved with respect to a baseline Transformer.

Statistical dictionaries and NMT

- Formulation:

$$\hat{y}_1^{\hat{I}} = \operatorname*{argmax}_{I, y_1^I} \prod_{i=1}^{I} p(y_i \mid y_1^{i-1}, x_1^J, \mathbf{M}(x_1^J))$$

In this case, $\mathbf{M}(x_1^J)$ represents a matrix M with the L large translations of each source word.

Statistical dictionaries and NMT

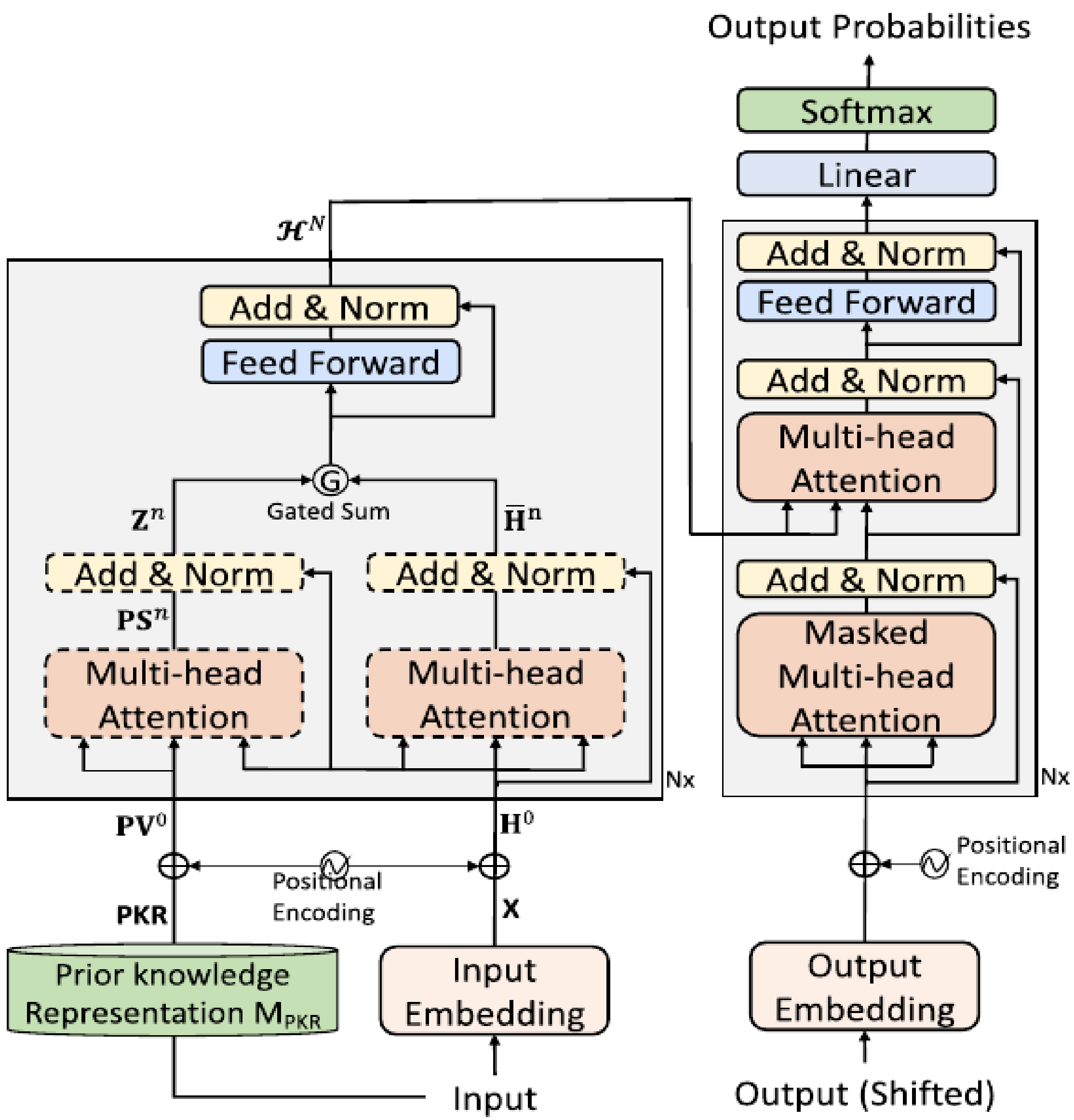
- Encoder implementation:

  $LN \equiv$ layer normalization, $\mathbf{F} \equiv$ feed-forward network and $\mathbf{A} \equiv$ attention mechanism.

$$
\begin{aligned}
\mathbf{U}^{e,l} &= LN(\mathbf{A}(\mathbf{W}_Q^{e,l}\mathbf{H}^{e,l-1}, \mathbf{W}_K^{e,l}\mathbf{H}^{e,l-1}, \mathbf{W}_V^{e,l}\mathbf{H}^{e,l-1}) + \mathbf{H}^{e,l-1}) \\
\mathbf{P}^{l} &= LN(\mathbf{A}(\mathbf{W}_Q^{es,l}\mathbf{H}^{e,l-1}, \mathbf{W}_K\mathbf{M}, \mathbf{W}_V\mathbf{M}) + \mathbf{H}^{e,l-1}) \\
g^{l} &= \mathbf{f}_S(\mathbf{W}_U^{l}\mathbf{U}^{e,l} + \mathbf{W}_G^{l}\mathbf{P}^{l}) \\
\overline{\mathbf{U}^{e,l}} &= \mathbf{U}^{e,l} + g^{l}\mathbf{P}^{l} \\
\mathbf{H}^{e,l} &= LN(\mathbf{F}(\overline{\mathbf{U}^{e,l}}) + \overline{\mathbf{U}^{e,l}})
\end{aligned}
$$

- Decoder implementation: similar to the decoder in the standard transformer.

Statistical dictionaries and NMT [Chen TASLP 2022]



Index

Syntax in NMT [Zhang NAACL 2019]

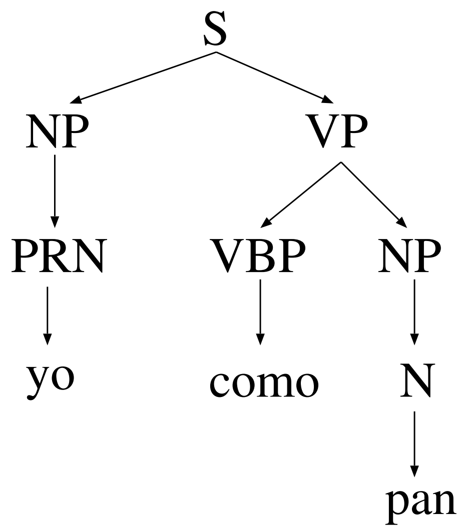
- Syntactic trees could offer long-distance relations in sentences
- Formulation:

$$\begin{aligned}\hat{y}_1^{\hat{I}} &= \operatorname*{argmax}_{I,y_1^I} p(y_1^I \mid x_1^J, tr(x_1^J)) \\ &= \operatorname*{argmax}_{I,y_1^I} \prod_{i=1}^{I} p(y_i \mid y_1^{i-1}, x_1^J, tr(x_1^J))\end{aligned}$$

- Approaches:
  - Tree-structured recurrent neural network (Tree-RNN) [Yang EMNLP 2017]
  - Tree-Linearization: traverse a constituent tree [Li ACL 2017].

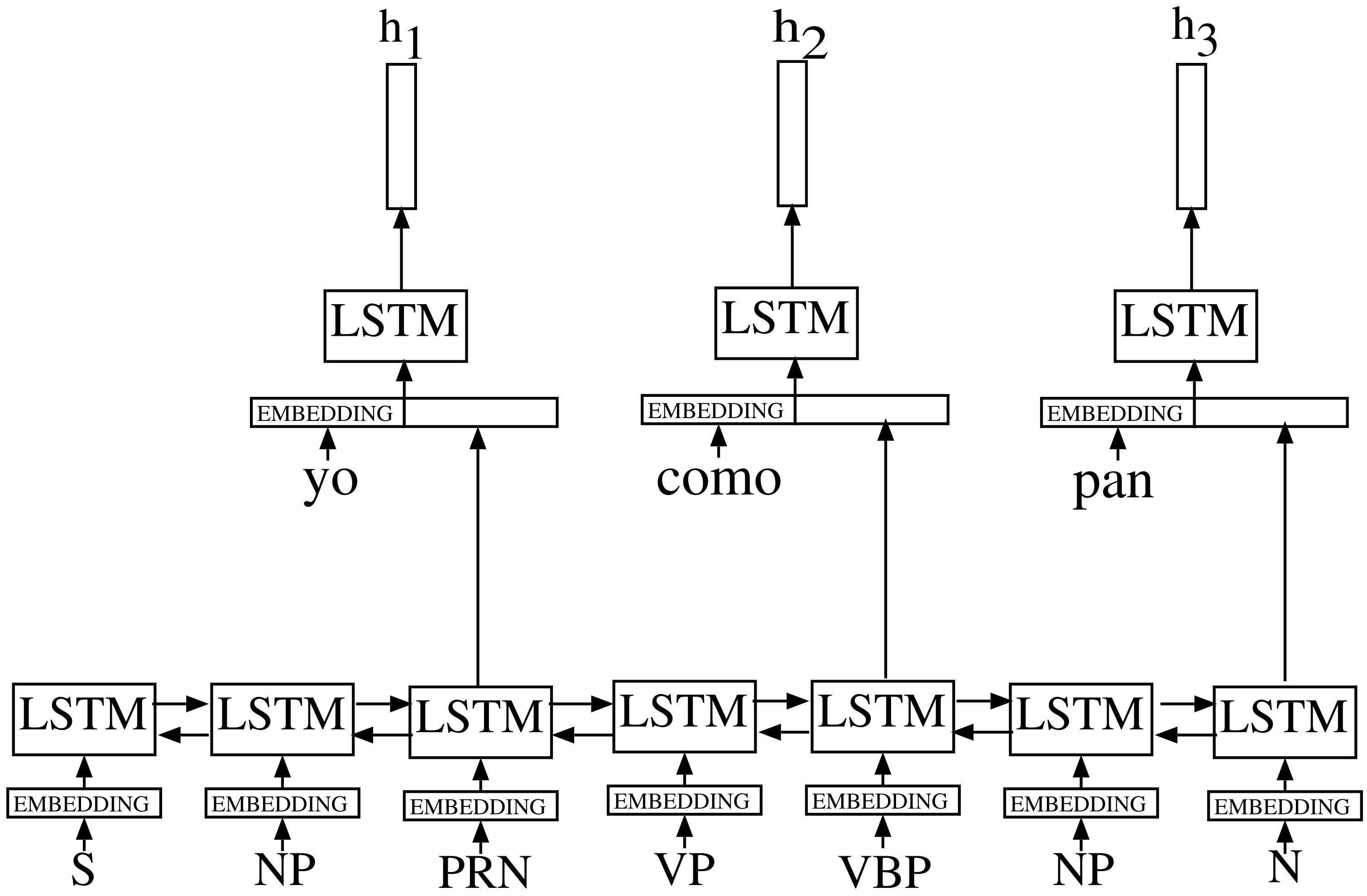
Tree-Linearization [Li ACL 2017]

- Sentence: "yo como pan"
- Syntax:

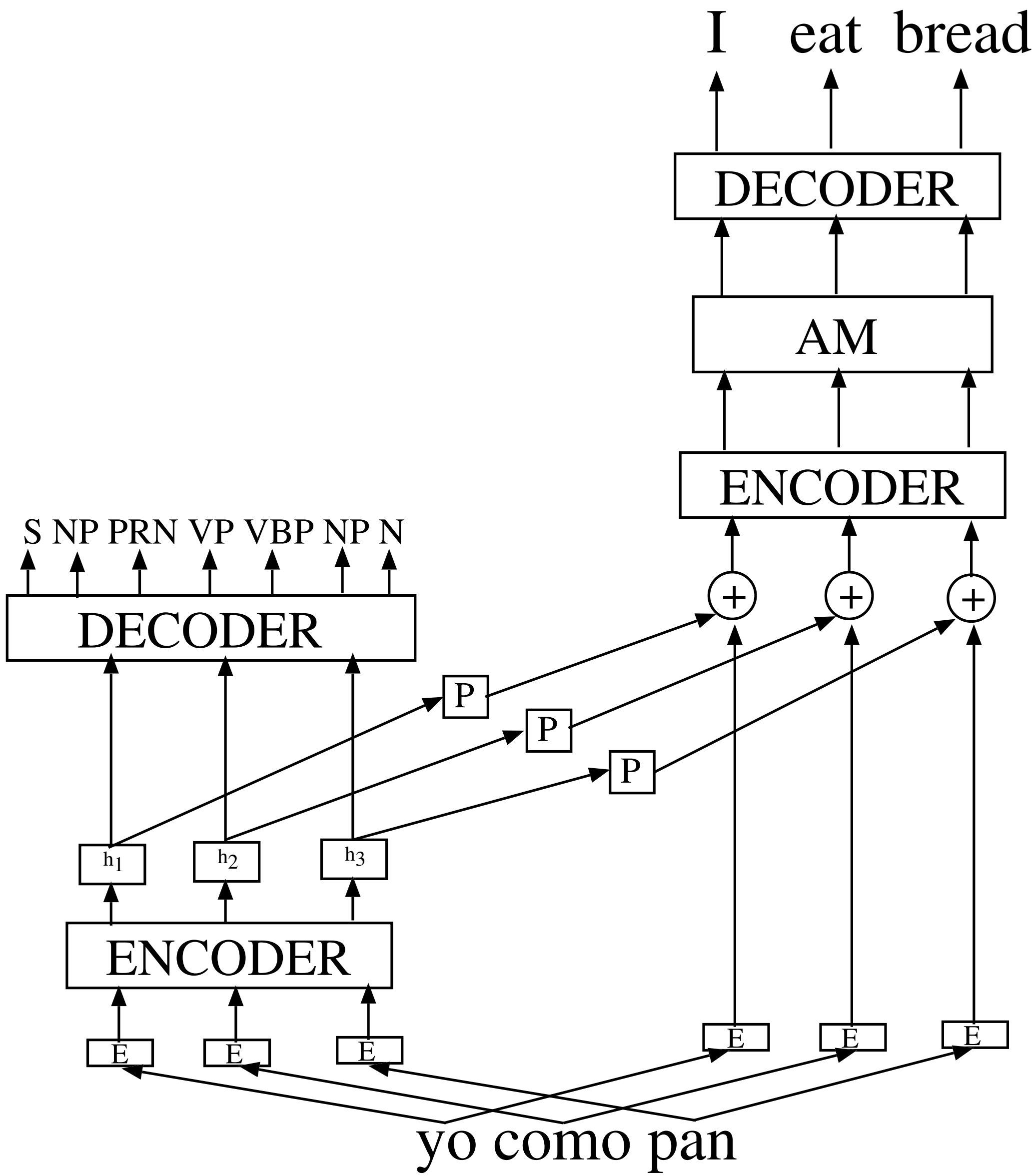


- Linearization: "S NP PRN VP VBP NP N"

Tree-Linearization and LSTMs: Hierarchical encoder [Li ACL 2017]



Syntax-Aware word representations [Zhang ACL 2019]



Factored Transformer [Armengol MT 2021]

- Factored machine translation: the use of word features alongside words themselves to improve translation quality. [Armengol MT 2021]

  Example:

yo	como	pan
PRONOUN	VERB	NOUN

- Using subwors units (i.e. BPE), the word label is repeated for each subword of the word.

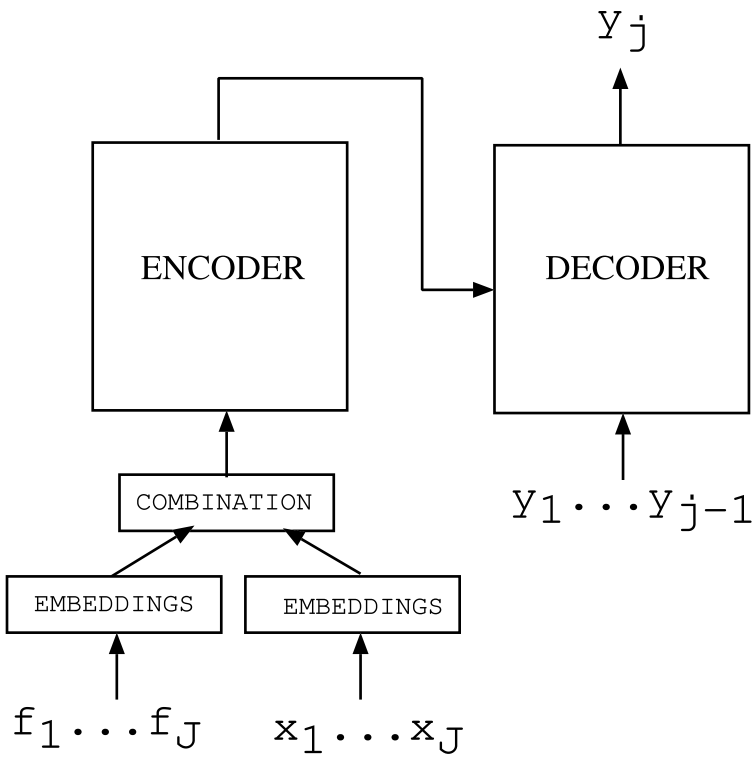
- Previous factored models: Phrase-based statistical translation models [Koehn ACl 2007] and reurrent neural networks [Garcia-Martinez arXiv 2016].

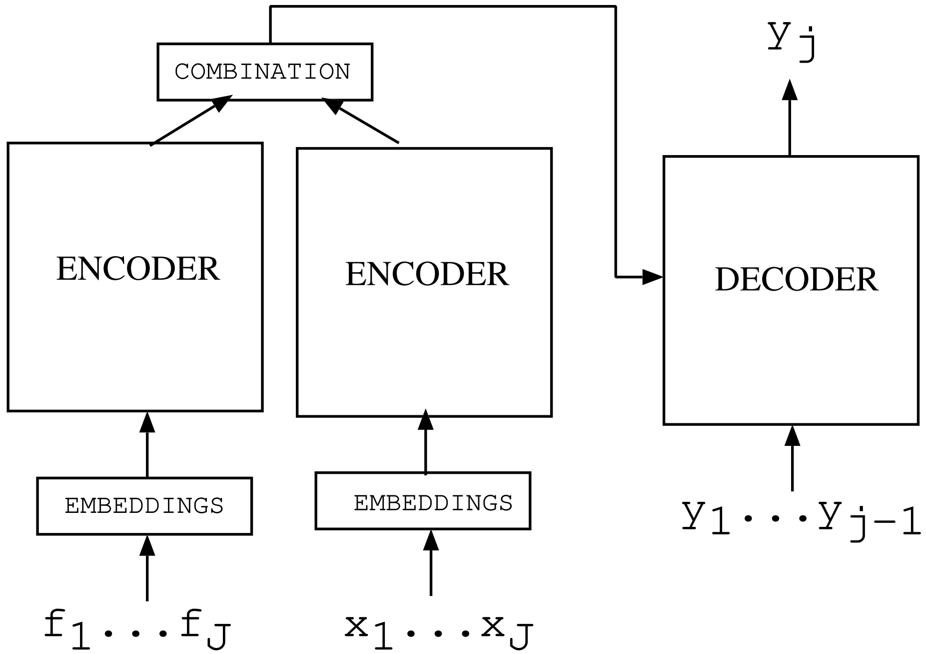
- Factored Transformer: regular Transformer [Vaswani arXiv 2017] with multiple encoders (a multimodal like approach), one for each factor plus the regular encoder.

Factored Transformer [Armengol MT 2021]

- Architectures:
  - In previous approaches (with LSTMs): 1-Encoder model.
    - Word and factor embedding are combined as the input of the encoder.
  - New approach: N-Encoder model.
    - One encoder for each factor plus the encoder for the source (sub)words.
    - States of the last layer of each encoder is combined to feed the cross-attention mechanism.

Factored Transformer





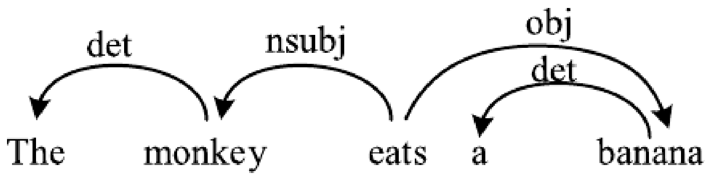
Factored Transformer [Armengol MT 2021]

- Combination strategies: For a sources sentence x_1^J, given N factor sequences F_j^n for $1 \leq j \leq J$ and $1 \leq n \leq N$ which outputs (embeddings for 1-Encoder or states for N-Encoder) $\mathbf{h}_j^n$
  - Concatenation: $\mathbf{h}_j = [\mathbf{h}_j^1; \ldots; \mathbf{h}_j^N]$ for $1 \leq j \leq J$.
  - Summation: $\mathbf{h}_j = \mathbf{h}_j^1 + \cdots + \mathbf{h}_j^N$ for $1 \leq j \leq J$

- On tasks IWSLT (Ge-En) and FLoRes (En-Nepali) , IWSLT14 (De-En) and IWSLT15 (En-Vi) small but significant improvements were achieved with respect to a baseline Transformer.

Syntax-graph guided self-attention [Gong KBS 2022]

- From a syntactic parsing of a source sentence x_1^J, a graph is built: the nodes represent the source words the edges represent their relationships.



- The graph is represented as a binay matrix M, where $M_{i,j}$ is set to 1 is there is a edge from token i to token j. $M_{i,i} \equiv 1$.

Syntax-graph guided self-attention [Gong KBS 2022]

- If subwords are used as units an new matrix is built M' such that if $M_{i,j} = 1$, then $M'_{i',j'} = 1$ for all subwords in position i' that correspond to word in position i and for all subwords in position j' that correspond to word in position j.

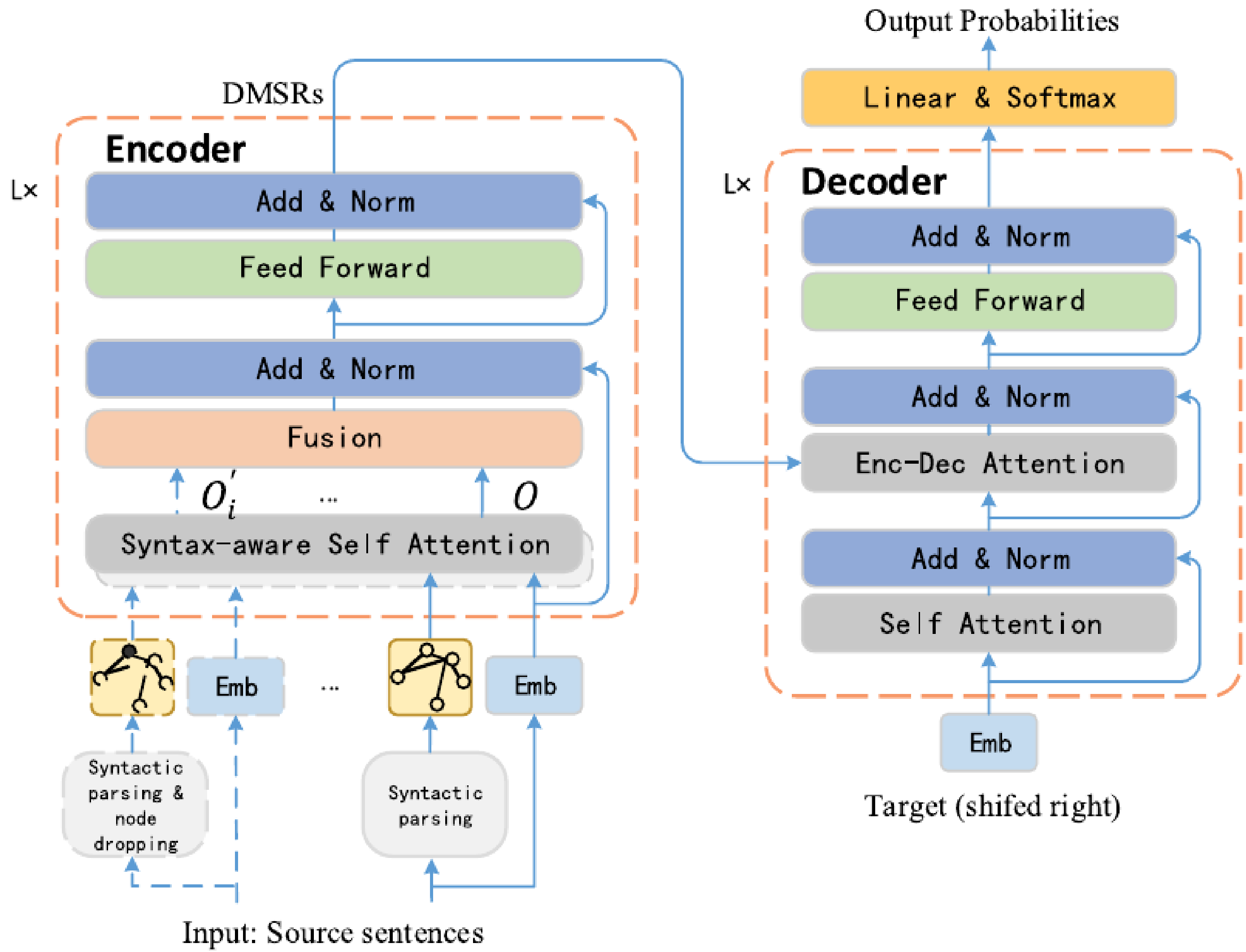
	The	monkey	eats	a	banana
The	1	1	0	0	0
monkey	1	1	1	0	0
eats	0	1	1	0	1
a	0	0	0	1	1
banana	0	0	1	1	1

(a) word-level.

	T@@	he	monkey	eats	a	ban@@	ana
T@@	1	1	1	0	0	0	0
he	1	1	1	0	0	0	0
monkey	1	1	1	1	0	0	0
eats	0	0	1	1	0	0	0
a	0	0	0	0	1	1	1
ban@@	0	0	0	0	1	1	1
ana	0	0	0	0	1	1	1

(b) sub-word-level.

Syntax-graph guided self-attention [Gong KBS 2022]



On tasks WMT18 (En-De), IWSLT14 (De-En) and IWSLT15 (En-Vi) small but significant improvements were achieved with respect to a baseline Transformer.

Index

Knowledge graphs [Zhao IJCAI 2020]

- Factual triplets as (subject entity, relation, object entity).

  Example: ("Pulp Fiction", "award received", "Palme d'Or")

- Knowledge graphs from factual triplets: entities $\equiv$ nodes; relations $\equiv$ edges.

- Nodes and edge embeddings: Given factual triplets (h, r, t), the goal is such that $\mathcal{E}(h) + \mathcal{E}(r) \approx \mathcal{E}(t)$.

- Approaches:
  - Generate training sentences from a knowledge graph.
  - Multitask learning: Additional enconder/decoder for training knowledge graphs.

Knowledge graphs in NMT [Zhao COLING 2020]

- Training data (Multitask learning):
  - A training set of bilingual sentences T
  - Training source KGs: $KG_s = \{(h_s, r_s, t_s)\}$
  - Training target KGs: $KG_t = \{(h_t, r_t, t_t)\}$

- Scenarios:
  - Only source KGs are available.
  - Only target KGs are available.
  - Both source and targt KGs are available.

Knowledge graphs in NMT [Zhao COLING 2020]

- Training procedure: Given T and KG_s and/or KG_t
  - Perform BPE with T and the entities of KG_s and/or KG_t.
  - Perform multi-task learning:
    - Machine Translation Task: Train a NMT by using T.
    - Knowledge Reasoning Task: Train a NMT to “translate” $\{h_1, \ldots, h_m\}$ (the heads of factual triplets) into $\{t_1, \ldots, t_m\}$ (the tails of the factual triplets)

Knowledge graphs in NMT [Zhao COLING 2020]

- Training criteria:
  - Scenario 1: Only source KGs are available.

$$L(\boldsymbol{\theta})_{T,KG_s} = \sum_{(\mathsf{x},\mathsf{y})\in T} \log p(\mathsf{y} \mid \mathsf{x}; \boldsymbol{\theta}) + \sum_{(h_s,r_s,t_s)\in KG_s} \log p(t_s \mid h_s; \boldsymbol{\theta})$$

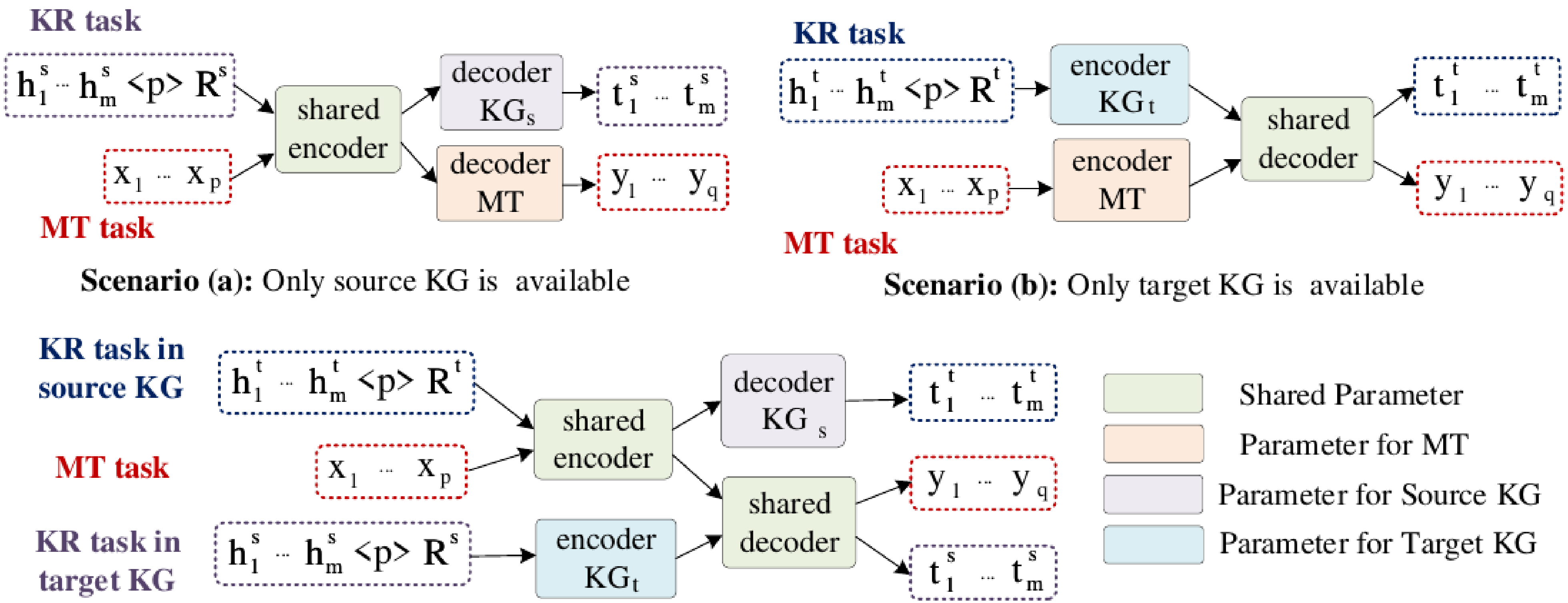
  - Scenario 2: Only target KGs are available.

$$L(\boldsymbol{\theta})_{T,KG_t} = \sum_{(\mathsf{x},\mathsf{y})\in T} \log p(\mathsf{y} \mid \mathsf{x}; \boldsymbol{\theta}) + \sum_{(h_t,r_t,t_t)\in KG_t} \log p(t_t \mid h_t; \boldsymbol{\theta})$$

  - Scenario 3: Both source and targt KGs are available.

$$L(\boldsymbol{\theta})_{T,KG_s,KG_t} = \sum_{(\mathsf{x},\mathsf{y})\in T} \log p(\mathsf{y} \mid \mathsf{x}; \boldsymbol{\theta}) + \sum_{(h_s,r_s,t_s)\in KG_s} \log p(t_s \mid h_s; \boldsymbol{\theta}) + \sum_{(h_t,r_t,t_t)\in KG_t} \log p(t_t \mid h_t; \boldsymbol{\theta})$$

Knowledge graphs in NMT [Zhao COLING 2020]



Index

Structural information in a set of word embeddings [Manning PNAS 2020]

- Is there structural information in a set of word embeddings from a neural language model?

- Structural information in a sentence: a undirected (or directed) tree T where nodes are the words in a sentence $\mathrm{x} = \mathrm{x}_1, \ldots, \mathrm{x}_J$
  - Distance between nodes x_i and x_j in T is set as the number of edges in the path from x_i to x_j: $d_T(\mathrm{x}_i, \mathrm{x}_j)$.

- Embeddings of words in a sentence from a language model (i.e. the pre-trained BERT).
  - Embedings of words x_i and x_j from a neural language model: $\mathbf{h}_i$ and $\mathbf{h}_j$, respectively.
  - Generalized distance between word embedding of x_i and word embedding of x_j in x: For a Matrix A such that $A = B^t B$.

$$d_A(\mathbf{h}_i, \mathbf{h}_j) \stackrel{\text{def}}{=} (\mathbf{h}_i - \mathbf{h}_j)^t A (\mathbf{h}_i - \mathbf{h}_j) = \|B(\mathbf{h}_i - \mathbf{h}_j)\|^2$$

Structural information in a set of word embeddings [Manning PNAS 2020]

- Given a set of L sentences $\{\mathsf{x}_1^l \ldots \mathsf{x}_{J_l}^l\}\big|_{l=1}^{L}$, each sentence labelled by a tree structure T_l, compute:

$$\underset{B}{\text{argmin}} \sum_{l=1}^{L} \frac{1}{J_l} \sum_{i,j} \left| d_{T_l}(\mathsf{x}_i^l, \mathsf{x}_j^l) - \|B(\mathbf{h}_i^l - \mathbf{h}_j^l)\|^2 \right|$$

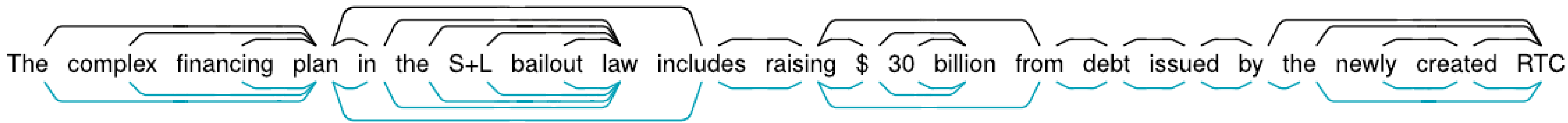
where $\mathbf{h}_i^l$ and $\mathbf{h}_j^l$ are the contextual word embeddings of x_i^L and x_j^L, respectivelly, obtained from a neural language model (i.e. BERT)

- That is, the distance between two word embeddings should be as close as possible to the distance of the two words in the structusal tree.

Structural information in a set of word embeddings [Manning PNAS 2020]

- At inference phase, given a sentence $\mathrm{x}_1 \dots \mathrm{x}_J$, a matrix is computed where each element is $d_A(\mathbf{h}_i, \mathbf{h}_j)$ that defines a undirected tree T which nodes are the words x_j for $1 \leq j \leq J$ and the adges are $(\mathrm{x}_i, \mathrm{x}_j)$ such that $d_A(\mathbf{h}_i, \mathbf{h}_j) = 1$. From T, minimum-spanning trees is obtained.

Blue, below: structural probe tree on BERT; Black, above: Human-Annotated tree



Index

Bibliography (1)

- Jordi Armengol-Estapé, Marta R. Costa-jussà. Semantic and syntactic information for neural machine translation. Machine Translation. 2021.
- Kehai Chen, Rui Wang, Masao Utiyama,Eiichiro Sumita. Integrating Prior Translation Knowledge Into Neural Machine Translation. IEEE/ACM Transactions on Audio, Speech, and Language Processing-2022.
- Mercedes García-Martínez, Loïc Barrault and Fethi Bougares. Factored Neural Machine Translation. arXiv 1609.04621. 2016.
- Longchao Gong, Yan Li, Junjun Guo , Zhengtao Yu, Shengxiang Gao. Enhancing low-resource neural machine translation with syntax-graph guided self-attention. Knowledge-Based Systems. 2022.
- Philipp Koehn et al. Moses: Open Source Toolkit for Statistical Machine Translation. Proceedings of the ACL. 2007.
- Fedor Moiseev, Zhe Dong, Enrique Alfonseca, Martin Jaggi. SKILL: Structured Knowledge Infusion for Large Language Models. Proceedings of the NAACl 2022.
- Junhui Li, Deyi Xiong, Zhaopeng Tu, Muhua Zhu, Min Zhang, Guodong Zhou. Modeling Source Syntax for Neural Machine Translation. Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, 2017.

Bibliography (2)

- Christopher D. Manning, Kevin Clark, John Hewitt, Urvashi Khandelwal, Omer Levy. Emergent linguistic structure in artificial neural networks trained by self-supervision. Proceedings of the National Academy of Sciences (PNAS). 2020.
- Baosong Yang, Derek F. Wong, Tong Xiao, Lidia S. Chao, Jingbo Zhu. Towards Bidirectional Hierarchical Representations for Attention-Based Neural Machine Translation. Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, 2017.
- A, Vaswani et al. Attention is all you needAdvances in neural information processing systems. 2017.
- Meishan Zhang, Zhenghua Li, Guohong Fu, Min Zhang. Syntax-Enhanced Neural Machine Translation with Syntax-Aware Word Representations. Proceedings of the North-American Association of Computational Linguistics, 2019.
- Yang Zhao, Jiajun Zhang, Yu Zhou and Chengqing Zong. Knowledge Graphs Enhanced Neural Machine Translation. Proceedings of Twenty-Ninth International Joint Conference on Artificial Intelligence. 2020.
- Yang Zhao, Lu Xiang, Junnan Zhu, Jiajun Zhang, Yu Zhou, Chengqing Zong. Knowledge Graph Enhanced Neural Machine Translation via Multi-task Learning on Sub-entity Granularity. Proceedings of the 28th International Conference on Computational Linguistics. 2020.