USI/SUPSI
Istituto
Dalle Molle
di studi
sull'intelligenza
artificiale
IDSIA

Non-Autoregressive Electron Redistribution Modeling for Reaction Prediction

Proceedings of the 38th International Conference on Machine Learning

Hangrui Bi, Hengyi Wang, Chence Shi, Connor Coley, Jian Tang, Hongyu Guo

About the authors

- *Dr. Jian Tang:*
- Assistant Professor at Mila – Quebec AI institute. Leader of the team developing TorchDrug.

- *Dr. Hongyu Guo:*
- Researcher at National Research Council Canada, one of the authors of “A graph to graphs framework for retrosynthesis prediction, 2020”.

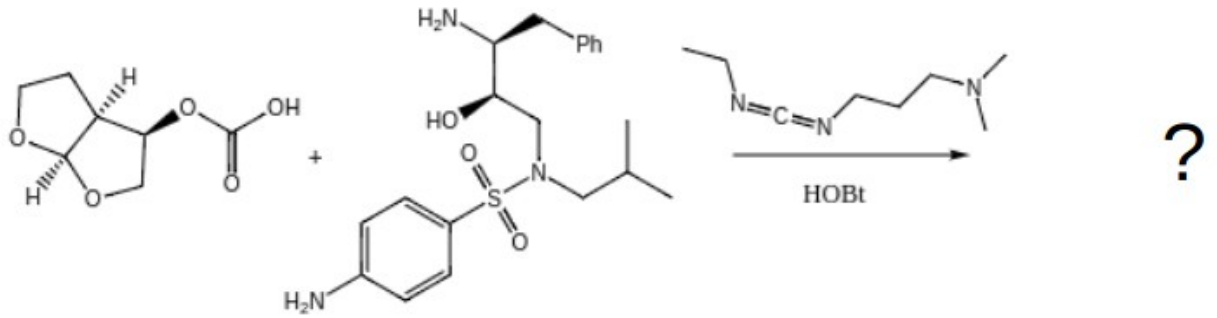
- *Dr. Connor Coley:*
- Assistant professor at MIT, head of the Coley group.

Task: Reaction prediction

USI/SUPSI
Istituto Dalle Molle di studi sull'intelligenza artificiale
IDSIA

A fundamental problem in computational chemistry. First formulated by Corey & Wipke in 1969.

Predict the products of an organic reaction given the reactants and reagents.



Brief history of the field

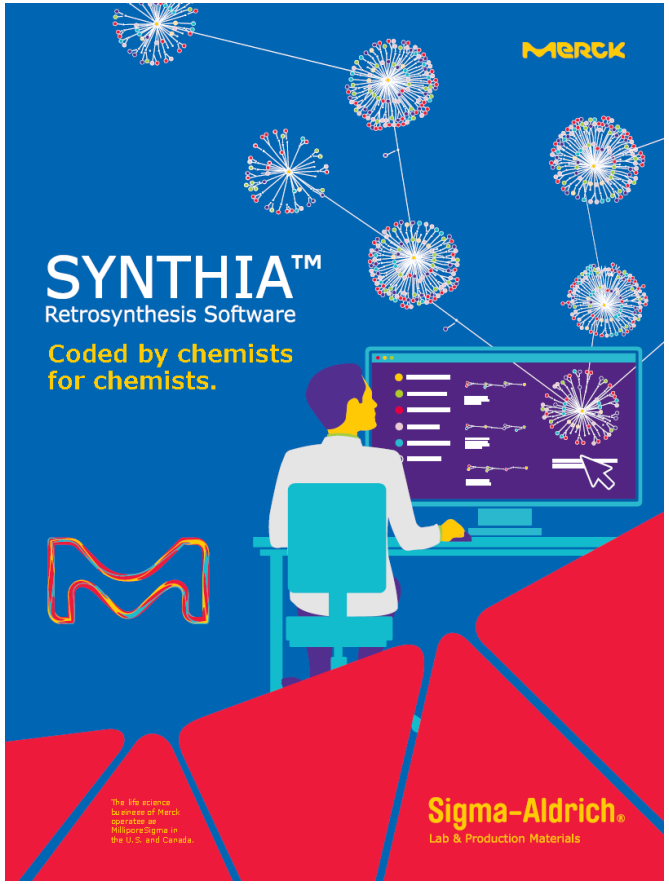
Istituto Dalle Molle di studi sull'intelligenza artificiale
USI/SUPSI
IDSIA

Template-based methods

Idea: *map reactions to predefined reaction templates.*

Developing since LHASA (1969).
Commercially succeful.

Dominant approach in reaction prediction before 2017



Synthia, formerly Chematica – a commercial tool for retrosynthesys

https://www.sigmaaldrich.com/CH/de/services/software-and-digital-platforms/synthia-retrosynthesis-software

Brief history of the field

USI/SUPSI Istituto Dalle Molle di studi sull'intelligenza artificiale IDSIA

Template-based methods

Idea: *map reactions to predefined reaction templates.*

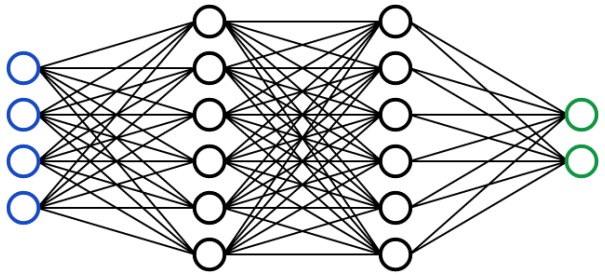
Developing since LHASA (1969). Commercially succeful.

Dominant approach in reaction prediction before 2017

Template-free methods

Let a model infer reaction rules themselves based on the training data.

Currently a common approach in reaction prediction.



Template-based methods

Istituto Dalle Molle di studi sull'intelligenza artificiale
USI/SUPSI
IDSIA

Neural Networks for the Prediction of Organic Chemistry Reactions

Jennifer N. Wei,[†] David Duvenaud,[‡] and Alán Aspuru-Guzik*[,†]

Neural-Symbolic Machine Learning for Retrosynthesis and Reaction Prediction

Marwin H. S. Segler[a] and Mark P. Waller*[a, b]

Prediction of Organic Reaction Outcomes Using Machine Learning

Connor W. Coley,[†] Regina Barzilay,[‡] Tommi S. Jaakkola,[‡] William H. Green,*[,†] and Klavs F. Jensen*[,†]

[†]Department of Chemical Engineering and [‡]Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology, 77 Massachusetts Avenue, Cambridge, Massachusetts 02139, United States

- **Cons:**
- No generalizabillity outside of the templates domain
- It's not straightforward to construct a good set of templates

Training data and benchmarks

USI/SUPSI
Istituto Dalle Molle di studi sull'intelligenza artificiale
IDSIA

The only open chemical reaction dataset – *chemical reactions from US patents*.

Gathered by Daniel Lowe in 2012 and presented in his doctoral thesis.

Size: 1-2 million reactions, pretty noisy.

There are different filtered subsets of it prepared by different authors.

reactions in	train	valid	test	total
USPTO_MIT set[23]	409 035	30 000	40 000	479 035
-No stereochemical information				
USPTO_LEF[25]	*	*	29 360	349 898
-Nonpublic subset of USPTO_MIT, without e.g. multistep reactions				
USPTO_STEREO[28]	902 581	50 131	50 258	1 002 970

From Schwaller et al. 2019

https://figshare.com/articles/dataset/Chemical_reactions_from_US_patents_1976-Sep2016_/5104873

Template-free methods

Istituto Dalle Molle di studi sull'intelligenza artificiale
USI/SUPSI
IDSIA

Molecule Edit Graph Attention Network: Modeling Chemical Reactions as Sequences of Graph Edits

Mikołaj Sacha,[†] Mikołaj Błaż,[†] Piotr Byrski,[†] Paweł Dąbrowski-Tumański,[†,‡] Mikołaj Chromiński,[¶] Rafał Loska,[§] Paweł Włodarczyk-Pruszyński,[†] and Stanisław Jastrzębski[*,†,∥,⊥]

ACS central science

Research Article

Cite This: *ACS Cent. Sci.* 2019, 5, 1572−1583

http://pubs.acs.org/journal/acscii

Molecular Transformer: A Model for Uncertainty-Calibrated Chemical Reaction Prediction

Philippe Schwaller,[*,†,‡] Teodoro Laino,[†] Théophile Gaudin,[†] Peter Bolgar,[§] Christopher A. Hunter,[§] Costas Bekas,[†] and Alpha A. Lee[*,‡]

[†]IBM Research − Zurich, Rüschlikon 8803, Switzerland

[‡]Department of Physics, University of Cambridge, Cambridge CB3 0HE, United Kingdom

[§]Department of Chemistry, University of Cambridge, Cambridge CB2 1EW, United Kingdom

GRAPH TRANSFORMATION POLICY NETWORK FOR CHEMICAL REACTION PREDICTION

Kien Do, Truyen Tran and Svetha Venkatesh
Applied Artificial Intelligence Institute
Deakin University, Geelong, Australia
{*dkdo,truyen.tran,svetha.venkatesh*}*@deakin.edu.au*

Chemical Science

ROYAL SOCIETY OF CHEMISTRY

EDGE ARTICLE

View Article Online
View Journal | View Issue

Check for updates

Cite this: *Chem. Sci.*, 2019, **10**, 370

All publication charges for this article have been paid for by the Royal Society of Chemistry

A graph-convolutional neural network model for the prediction of chemical reactivity†

Connor W. Coley,[a] Wengong Jin,[b] Luke Rogers,[a] Timothy F. Jamison,[c] Tommi S. Jaakkola,[b] William H. Green,[a] Regina Barzilay*[b] and Klavs F. Jensen*[a]

Brief history of the field

Important concepts:

Reactants:

Molecules that contribute atoms to products.

Reagents:

Molecules that don’t change by the end of the reaction but are nesessary to make it possible, e.g. catalysts and solvents.

Atom-to-atom mapping:

Numeric labels of atoms preserved between both sides of a reaction. Can be incorporated into SMILES strings.

An example of an atom-mapped reaction SMILES string:
[Cl:25][CH2:26][Cl:27].[Na+:13].[Na+:14].[Na+:24].[O-:15][S:16]([O-:17])(=[S:18])=[O:19].[O-:20][C:21]([OH:22])=[O:23].[OH:1][CH2:2][CH2:3][CH2:4][NH:5][C:6]([O:7][C:8]([CH3:9])([CH3:10])[CH3:11])=[O:12]>>[O:1]=[CH:2][CH2:3][CH2:4][NH:5][C:6]([O:7][C:8]([CH3:9])([CH3:10])[CH3:11])=[O:12]

Performance comparison

- All of the recent template-free methods perform roughly the same on USPTO. Molecular transformer is mostly the best model.

Top-k accuracy on USPTO-MIT:

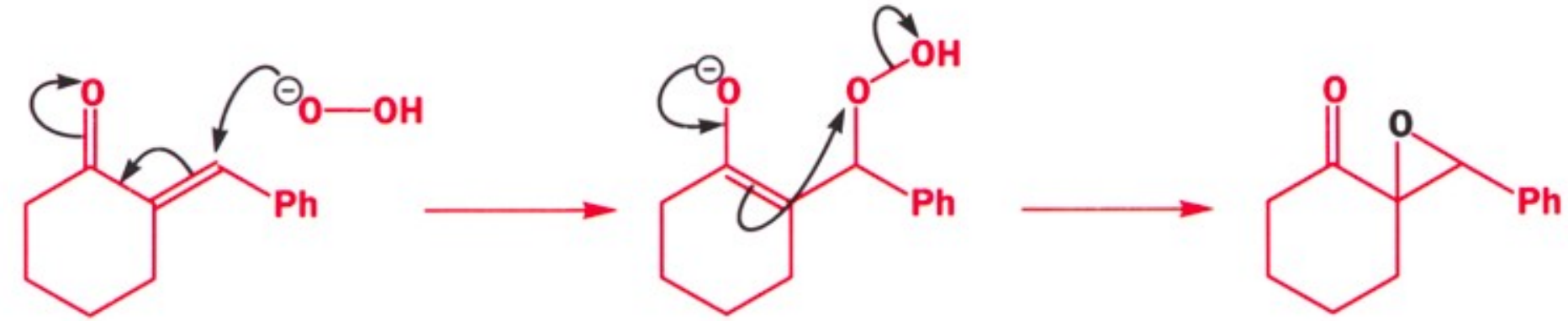
	Accuracies(%)					
Model Name(scheme)	Top-1	Top-2	Top-3	Top-5	parallel	end-to-end
WLDN† (combinatorial)	79.6	-	87.7	89.2	✓	×
GTPN †(graph)	83.2	-	86.0	86.5	×	✓
Transformer-base † (sequence)	88.8	92.6	93.7	94.4	×	✓
MEGAN†(graph)	89.3	92.7	94.4	**95.6**	×	✓
Transformer-augmented †(sequence)	90.4	**93.7**	**94.6**	95.3	×	✓
Symbolic† (combinatorial)	90.4	93.2	94.1	95.0	✓	×

Electron flow modelling

USI/SUPSI
Istituto Dalle Molle di studi sull'intelligenza artificiale
IDSIA

Theoretically, all chemical reactions can be described by the stepwise rearrangement of electrons in molecules.

Why not try to model this mechanism with neural networks?



Picture from the organic chemistry textbook by Clayden, Greeves, Warren

Linear electron flow modelling

USI/SUPSI
Istituto Dalle Molle di studi sull'intelligenza artificiale
IDSIA

Consider reactions from USPTO with *linear electron flow*.

Sequentially predict how the reaction unfolds, adding or deleting one bond at a time.

A GENERATIVE MODEL FOR ELECTRON PATHS

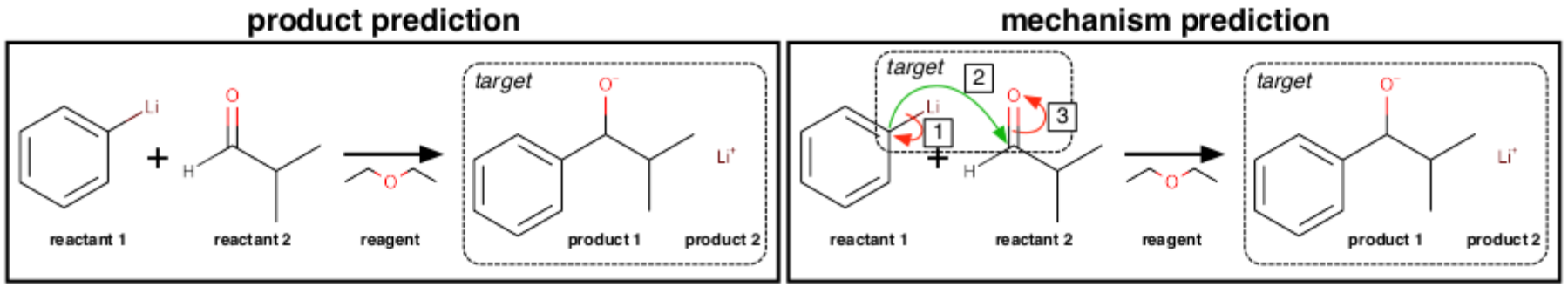
John Bradshaw
University of Cambridge
Max Planck Institute, Tübingen
jab255@cam.ac.uk

Matt J. Kusner
University of Oxford
Alan Turing Institute
mkusner@turing.ac.uk

Brooks Paige
Alan Turing Institute
University of Cambridge
bpaige@turing.ac.uk

Marwin H. S. Segler
BenevolentAI
marwin.segler@benevolent.ai

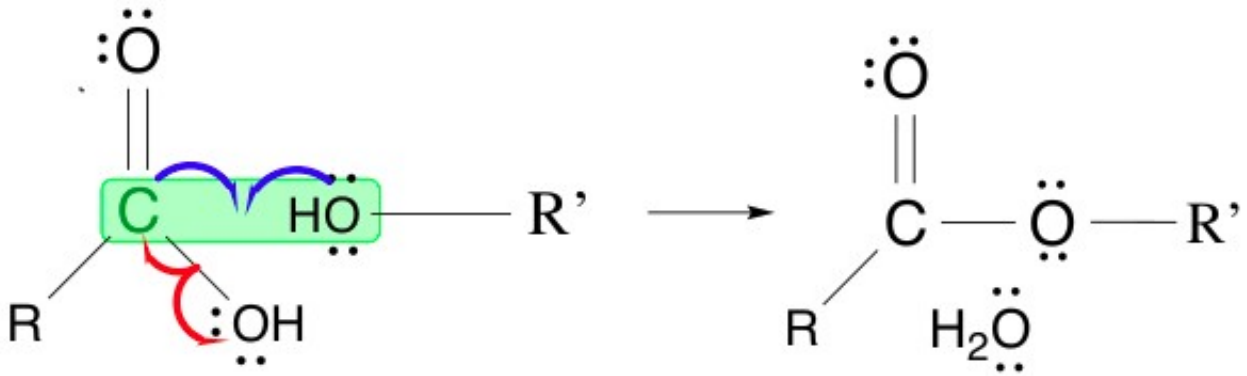
José Miguel Hernández-Lobato
University of Cambridge
Microsoft Research Cambridge
Alan Turing Institute
jmh233@cam.ac.uk



New idea

USI/SUPSI
Istituto Dalle Molle di studi sull'intelligenza artificiale
IDSIA

- Why not model the redistribution of electrons in one-shot?
- What is we consider individual electrons instead of labeled bonds?

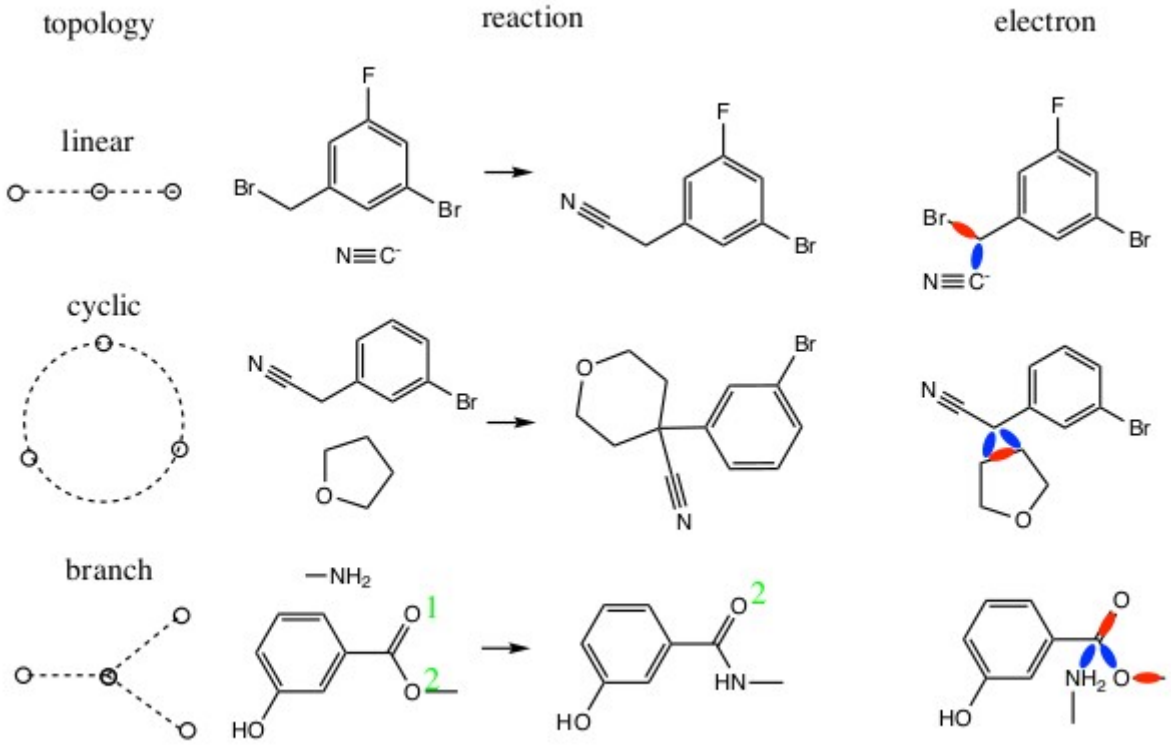


A half-arrow means a single electron instead of a pair.

We can get the products instantly without unfolding a recurrent mechanism.
We can also get a range of other advantages.

Advantages of the model

- Template-free
- Can model arbitrary electron redistribution.
- Better than *Bradshaw et al. 2018*, *Sacha et al. 2020* *Do et al. 2018*
- Can train end-to-end.
- Better than *Qian et al. 2020*, *Jin et al. 2017*
- Predictions are interpretable by chemists.
- Better than *Schwaller et al. 2019*
- **Naturally predicts side products**



Setup of the paper

The authors called their framework NERF.

It is written in Pytorch 1.8.1.

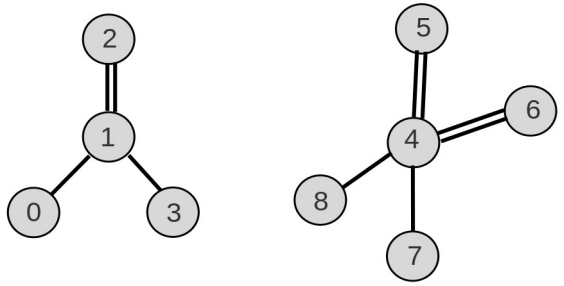
The model is mostly a transformer,

but over **atoms**, not text tokens

Setup of the paper

- Dataset:
- USPTO_MIT, atom mapping is required.
- Representation:
- Adjacency list of a graph of all involved atoms + atom features.
- Atom features:
- Atomic number, aromaticity, formal charge, some masks and flags
- **Assume each atom has 6 bonds at most.**

$$G_r = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 2 & 3 & 1 & 1 \\ 1 & 1 & 2 & 2 & 2 & 2 \\ 1 & 3 & 3 & 3 & 3 & 3 \\ 5 & 5 & 6 & 6 & 7 & 8 \\ 4 & 4 & 5 & 5 & 5 & 5 \\ 4 & 4 & 6 & 6 & 6 & 6 \\ 4 & 7 & 7 & 7 & 7 & 7 \\ 4 & 8 & 8 & 8 & 8 & 8 \end{bmatrix}$$



Example of a reaction graph and its adjacency list

Objective of the model

Build the adjacency matrices using adjacency lists.

Predict the difference in the reaction adjacency matrix before and after the reaction.

$$e^p_{ij} = \begin{bmatrix} 5 & 1 & 0 & 0 & 0 \\ 1 & 4 & 1 & 0 & 0 \\ 0 & 1 & 4 & 0 & 1 \\ 0 & 0 & 0 & 6 & 0 \\ 0 & 0 & 1 & 0 & 5 \end{bmatrix} \quad e^r_{ij} = \begin{bmatrix} 5 & 1 & 0 & 0 & 0 \\ 1 & 4 & 1 & 0 & 0 \\ 0 & 1 & 4 & 1 & 0 \\ 0 & 0 & 1 & 5 & 0 \\ 0 & 0 & 0 & 0 & 6 \end{bmatrix} \quad \Delta = e^p_{ij} - e^r_{ij} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 \\ 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & 1 & 0 & -1 \end{bmatrix}$$

$$\delta_{ij} = \Delta - \Delta\tilde{w}_{ij}$$

Calculated based on AAM

Output of the model (based on atom features)

Loss function: $\mathcal{L}_{bond} = \sum_{i,j \in V} \delta^2_{ij}$

Message-passing GNN

USI/SUPSI
Istituto Dalle Molle di studi sull'intelligenza artificiale
IDSIA

Idea: Representation learning.

Learn the best node representations taking the information of the local connectivity into account.

Update each node embedding using a learnable function of its neighbours' embeddings.

Several MPGNN layers can be stacked together.

Each node embedding is updated based on the information carried by its neighbours.

Particular example:

$$m_i^{r(l)} = \text{RELU}(W^{(l)} \cdot \text{SUM}\{h_u^{r(l-1)} | u \in \mathcal{N}_i\})$$

$$h_i^{r(l)} = m_i^{r(l)} + h_i^{r(l-1)}$$

Transformer

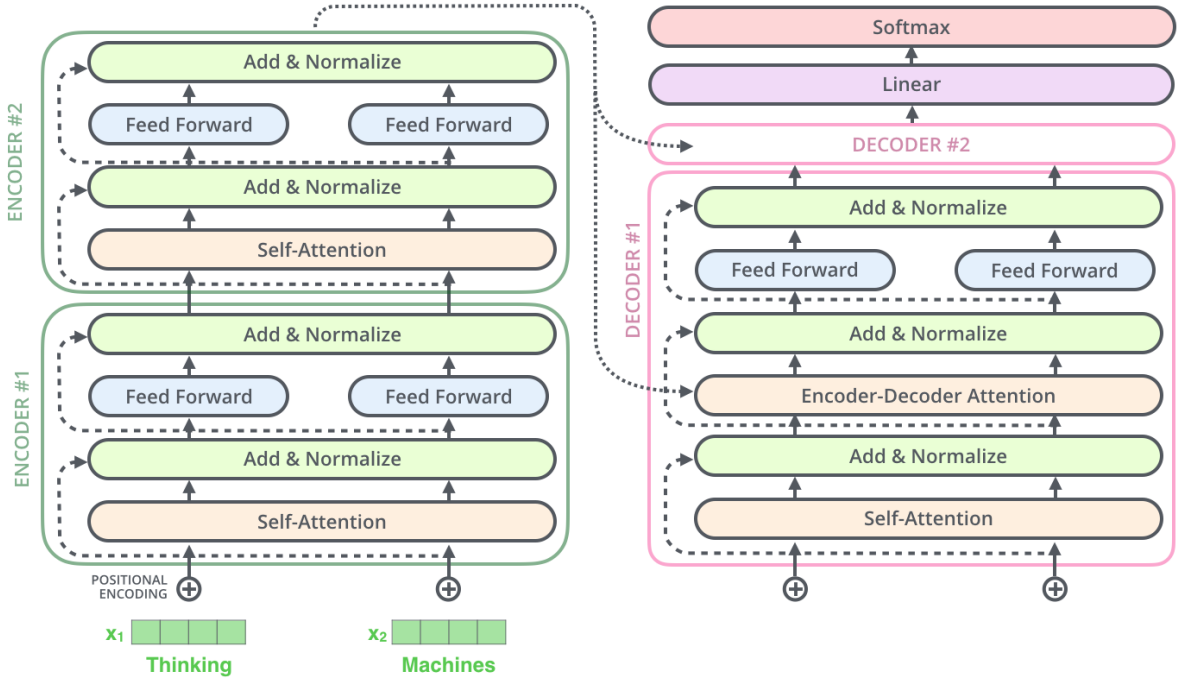
A powerful architecture first used in neural machine translation.

Relies on *multi-head self-attention mechanism* for representation learning.

In the machine translation setting receives embeddings of the text tokens as input.

This is also the case for molecular transformer.

However, the input embeddings could represent something else, e.g. atom features.



Picture from https://jalammar.github.io/illustrated-transformer/

Vaswani, A. et al. *Attention is all you need.* 2017. Advances in Neural Information Processing Systems 30.

Self-attention

USI/SUPSI
Istituto Dalle Molle di studi sull'intelligenza artificiale
IDSIA

A method of representation learning. Multihead attention works like a GNN, but on a fully connected graph.

Text modeling: emdeddings of tokens are updated with the function of embeddings of all other tokens in a sequence.

An embedding of a token is updated with a weighted sum of value vectors of other tokens

$$\boldsymbol{h}_j = \sum_{i \in S} \alpha_i \boldsymbol{v}_i$$

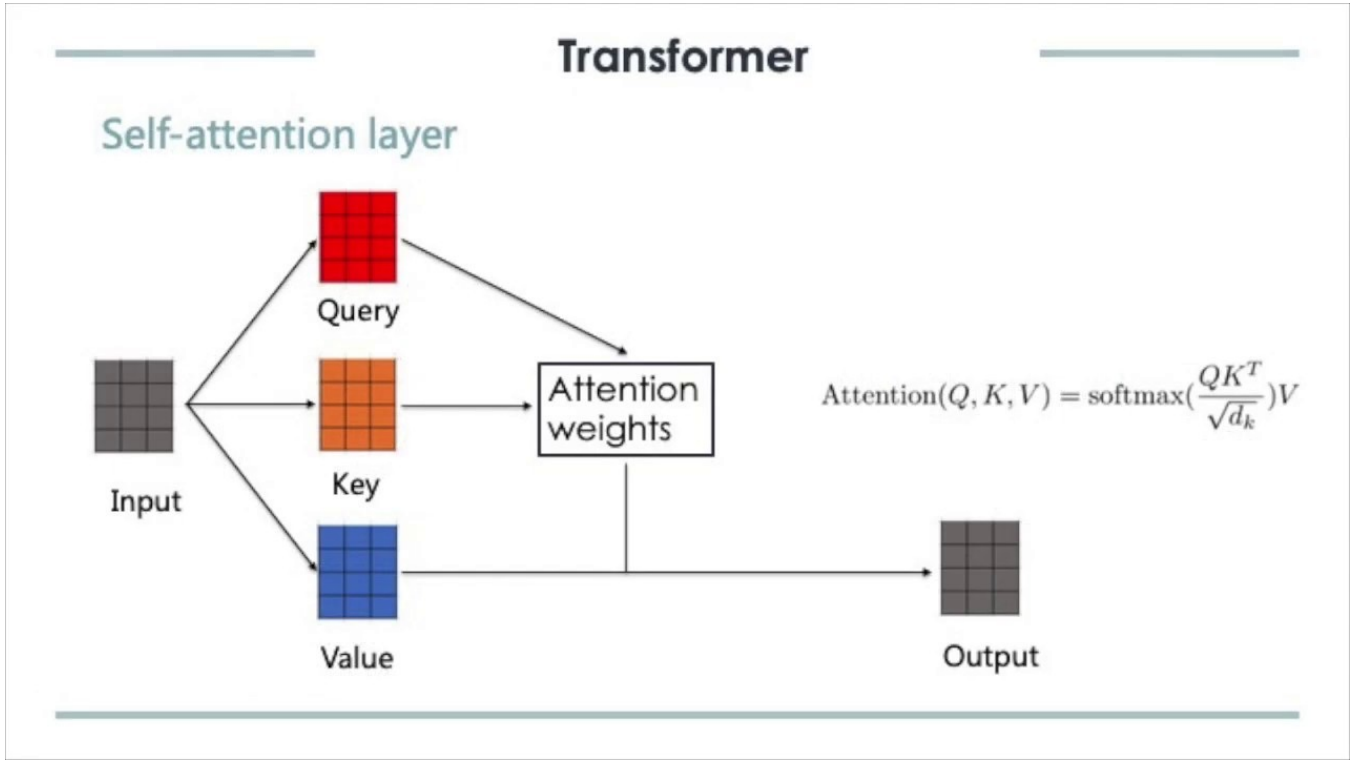
$$\alpha_i = \frac{e^{(\boldsymbol{q}_i^T \boldsymbol{k}_i)}}{\sum_{j \in S} e^{(\boldsymbol{q}_j^T \boldsymbol{k}_j)}}$$

$$\boldsymbol{v} = f_v(\boldsymbol{h})$$

$$\boldsymbol{k} = f_k(\boldsymbol{h})$$

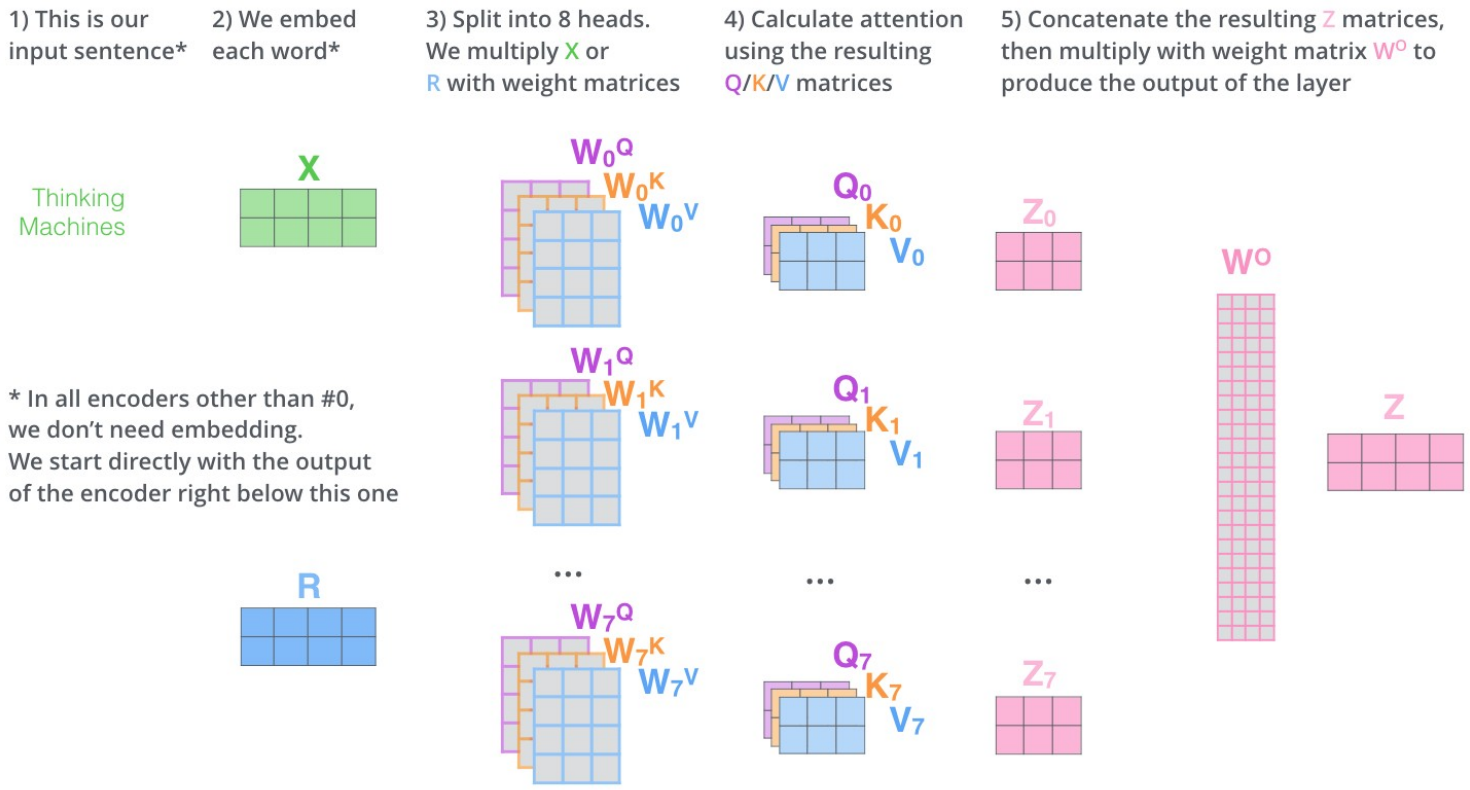
$$\boldsymbol{q} = f_q(\boldsymbol{h})$$

Learnable functions



Multi-head self-attention

Multi-head just means that instead of one self-attention layer we use several smaller self-attention layers in parallel and concatenate the results.



Picture from https://jalammar.github.io/illustrated-transformer/

Variational Autoencoder

USI/SUPSI
Istituto Dalle Molle di studi sull'intelligenza artificiale
IDSIA

A generative model. Given data ***X***, trains to model the distribution *P(**X**)* of data using a latent variable ***z***.

$$P(X) = \int_z P(X|z)P(z)dz = \mathbb{E}_{z\sim P(z)}P(X|z)$$

Approximate $P(X)$ by $\mathbb{E}_{z\sim Q(z|X)}P(X|z)$

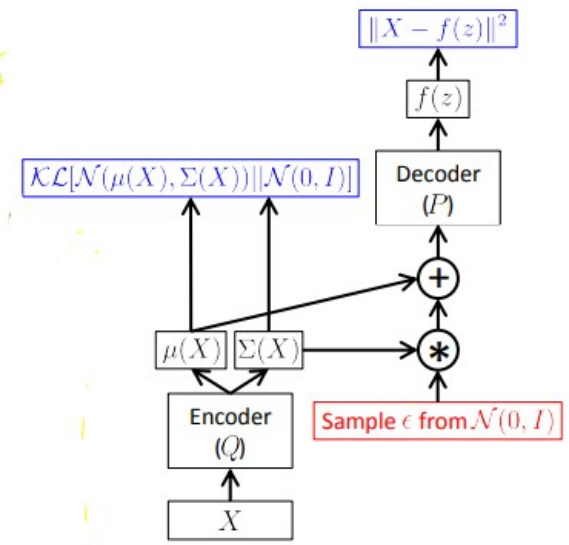
$$P(X|z) = \mathcal{N}(f_z(z), \sigma^2 I)$$

$$Q(z|X) = \mathcal{N}(f_\mu(X), f_\sigma(X)I)$$

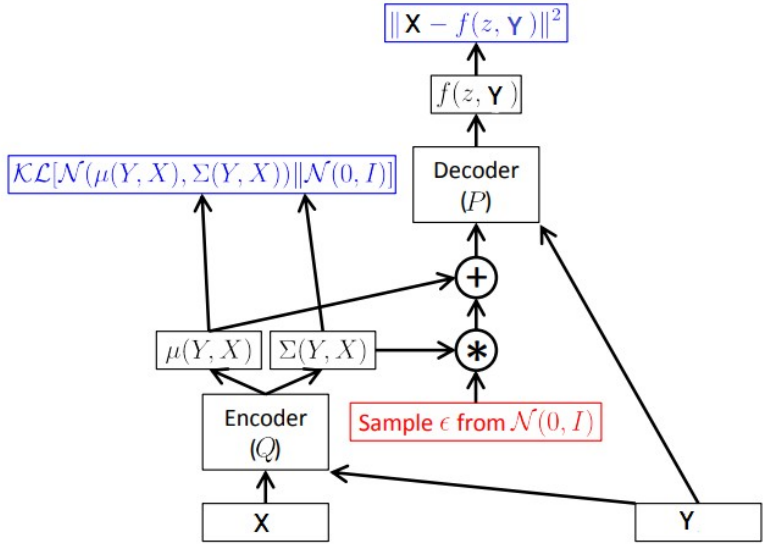
$$P(z) = \mathcal{N}(0, I)$$

f_z, f_μ, f_σ - trainable functions.

$$\log P(X) \geqslant -\mathbb{E}_{z\sim Q(z|X)}(\log P(X|z)) + D_{KL}(Q(z|X)||P(z))$$



VAE

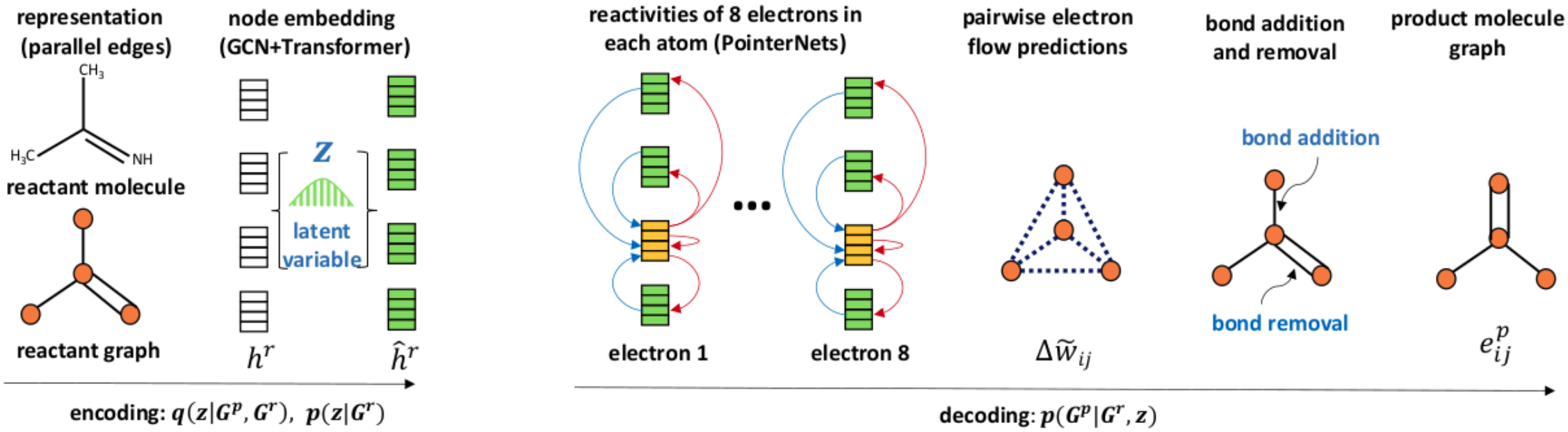


CVAE

Architecture of the paper

Istituto Dalle Molle di studi sull'intelligenza artificiale
USI/SUPSI
IDSIA

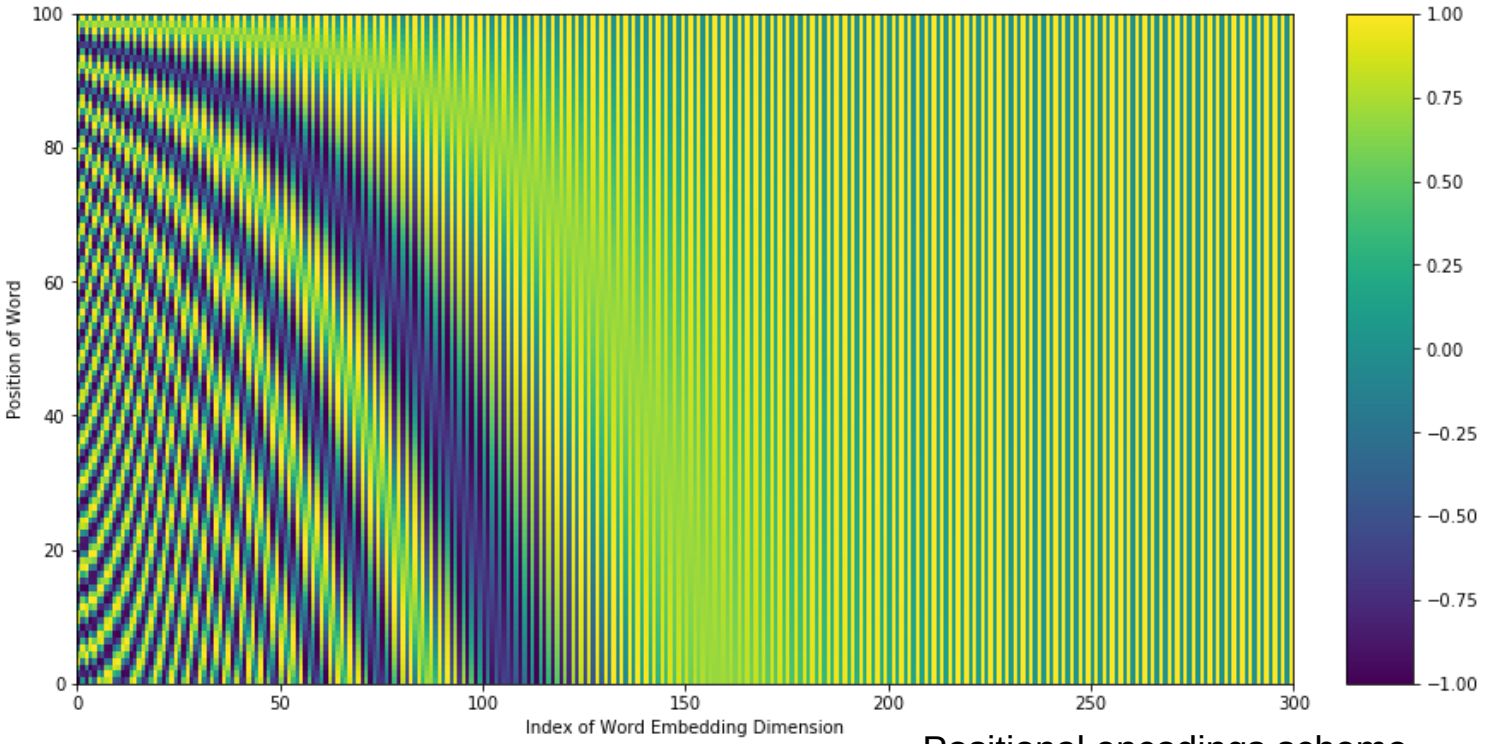
Model the probability of product graphs G^p conditioning on the reactant graphs G^r, i.e. P(G^p| G^r).
1. Learn atom representations with GCN and Transformer-encoder layers.
2. Squeeze them into a reaction embedding. Use it as an input to CVAE. Train a latent distribution.
3. Sample a latent variable, add it to atom representations, pass them into multi-head self-attention.
4. The attention scores are the output of the model.



Encoder: GNN

USI/SUPSI
Istituto Dalle Molle di studi sull'intelligenza artificiale
IDSIA

The encoder starts with a one message-passing layer.
1). Every atom feature gets an embedding by the means of `torch.nn.Embedding`.
2). The sum of those embeddings + positional embedding = atom embedding.
3). Atom embeddings get updated by one message passing layer.



Positional encodings scheme

$$\boldsymbol{h}_i \leftarrow f\Big(\sum_{j \in \mathcal{N}(i)} \boldsymbol{h}_j\Big)$$

Here *f* is an MLP based on 1D-convolutions.

Every batch has *L* atoms, every atom has an embedding vector of length *D.*

Encoder: Transformer encoder

After that the atom embeddings are passed into a transformer encoder.

In pytorch is is very straightforward:

```
layer = TransformerEncoderLayer(dim, nhead, dim, dropout)
self.transformer_encoder = TransformerEncoder(layer, nlayer)
encoder_output = self.transformer_encoder(embedding, src_key_padding_mask=mask)
```

All atom embeddings get updated taking the global graph information into account.

Authors use 6 layers and 8 self-attention heads.

Encoder: CVAE

Again, authors aim to model the conditional distribution $P(G^p|G^r)$.

$$\log p(G^p|G^r) \geq \mathrm{E}_{q(z|G^p,G^r)}[\log p(G^p|G^r,z)] - KL(q(z|G^p,G^r)||p(z|G^r)),$$

They do it just like one does in VAE, except the input is special:

The input to VAE is an embedding of an entire reaction.

To build it, authors do the following:

1). Pass the initial molecules through the model encoder and obtain embeddings $\boldsymbol{h}^r$ (already done).
2). Pass the target molecules through the model encoder and obtain embeddings $\boldsymbol{h}^p$.
3). Pass $\boldsymbol{h}^p$ and $\boldsymbol{h}^r$ into a Transformer decoder. Update the atom embeddings $\boldsymbol{h}^r$ by attending to $\boldsymbol{h}^p$ *and get* $(\boldsymbol{h}^r)^*$.
4). Calculate a mean of the new $(\boldsymbol{h}^r)^*$ and use it as an input to CVAE.

Decoder: Pointer Networks

USI/SUPSI
Istituto Dalle Molle di studi sull'intelligenza artificiale
IDSIA

Now we can use atom embeddings to predict the change in connectivity.

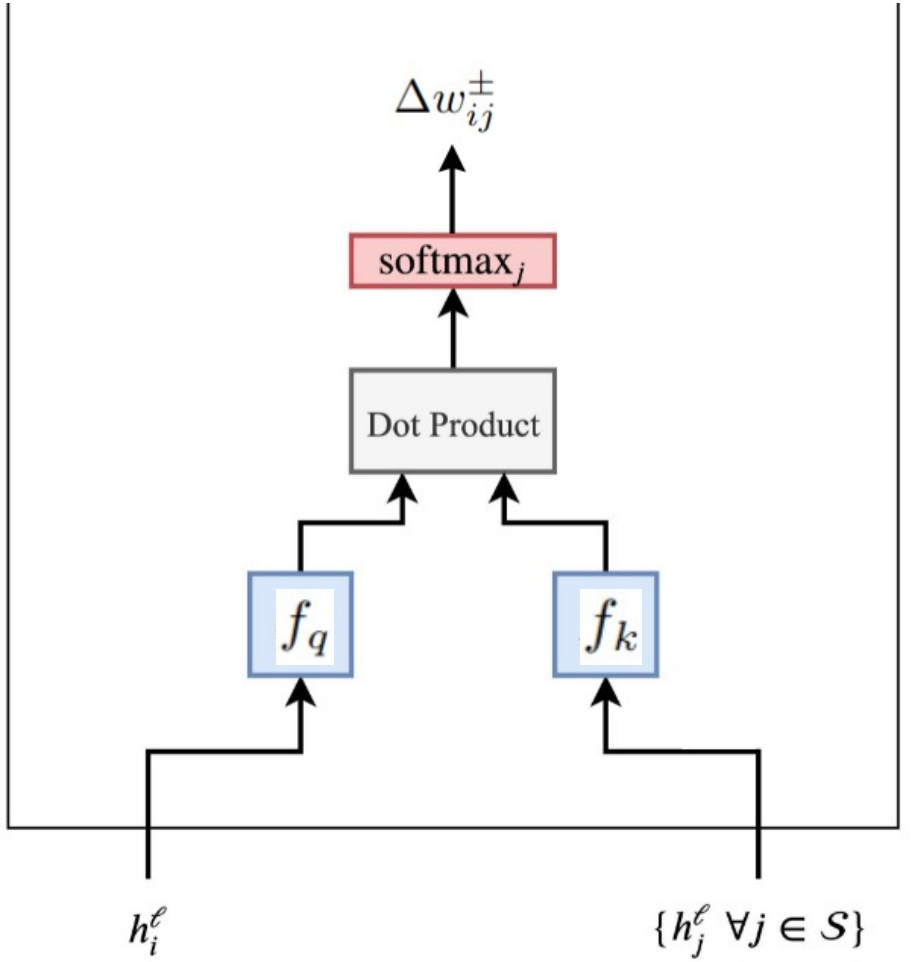
$$\boldsymbol{h}_z^r = \boldsymbol{h}^r + \boldsymbol{z}$$

$$\tilde{\boldsymbol{h}}_z^r = \mathrm{TransformerEncoder}(\boldsymbol{h}_z^r)$$

$$\Delta\tilde{w}_{ij} = \mathrm{BondDecoder}(\tilde{\boldsymbol{h}}_z^r)$$

Bond Decoder consists of two parallel self-attention layers

$$\Delta\tilde{w}_{ij} = \sum_{d=1}^{4} w_{ij}^{+d} - \sum_{d=1}^{4} w_{ij}^{-d}$$



Decoder: Additional features

Authors also use $\tilde{\boldsymbol{h}}_z^r$ to predict atom features like aromaticity and formal charge in a supervised manner.

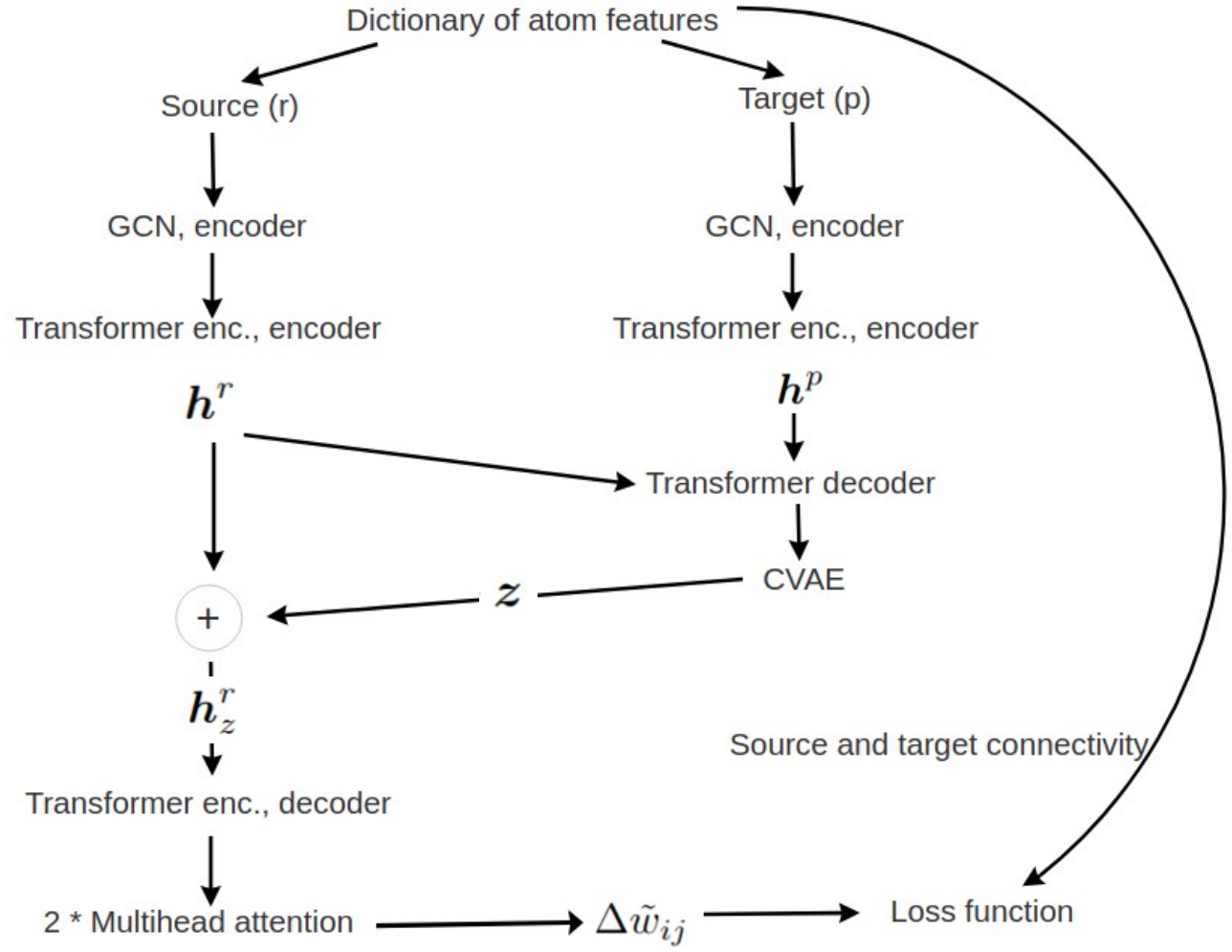
This is necessary to construct a resulting reaction graph. Authors also suggest predicting chirality.

Therefore, the overall loss function of the model is as follows:

$$\mathcal{L} = -\sum_{i,j \in V} (e_{ij}^p - e_{ij}^r - \Delta\tilde{w}_{ij})^2 + D_{KL}(Q(z|G^p, G^r)||P(z|G^r)) + \mathcal{L}_{BCE,charge} + \mathcal{L}_{BCE,aromaticity}$$

Overall architecture

USI/SUPSI
Istituto Dalle Molle di studi sull'intelligenza artificiale
IDSIA



Results: performance

USI/SUPSI
Istituto Dalle Molle di studi sull'intelligenza artificiale
IDSIA

Table 1. Top-k accuracy on USPTO-MIT; Best results in **bold**. We also show if the comparison model can be parallelly trained in an end-to-end fashion. † indicates that the results were copied from its published paper. The bracket indicates the method's learning taxonomy: "combinatorial" for parallel optimization, "graph" for graph translation, and "sequence" for an auto-regressive generation.

	Accuracies(%)					
Model Name(scheme)	Top-1	Top-2	Top-3	Top-5	parallel	end-to-end
WLDN† (combinatorial)	79.6	-	87.7	89.2	✓	×
GTPN †(graph)	83.2	-	86.0	86.5	×	✓
Transformer-base † (sequence)	88.8	92.6	93.7	94.4	×	✓
MEGAN†(graph)	89.3	92.7	94.4	**95.6**	×	✓
Transformer-augmented †(sequence)	90.4	**93.7**	**94.6**	95.3	×	✓
Symbolic† (combinatorial)	90.4	93.2	94.1	95.0	✓	×
NERF	**90.7±0.03**	92.3±0.22	93.3±0.15	93.7±0.17	✓	✓

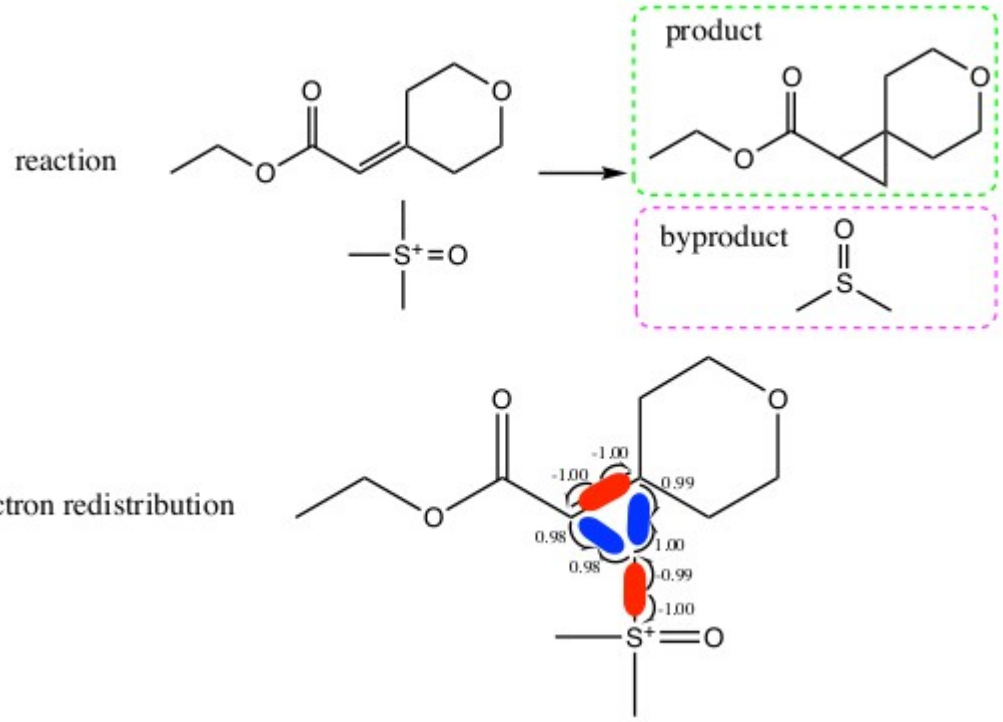
Statistical significance tests vere conducted.

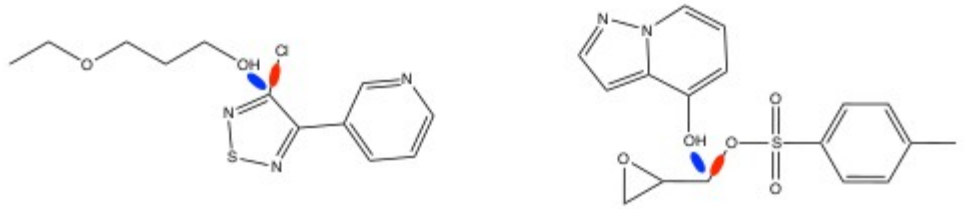
Table 2. Computation speedup (compared with Transformer)

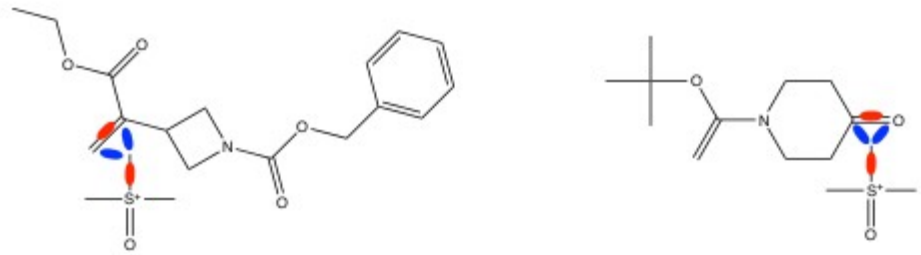
Model Name	Wall-time	Latency	Speedup
Transformer (b=5)	9min	448ms	1 ×
MEGAN (b=10)	31.5min	144ms	0.29 ×
Symbolic	>7h	1130ms	0.02 ×
NERF	20s	17ms	27×

Results: interpretability

USI/SUPSI
Istituto Dalle Molle di studi sull'intelligenza artificiale
IDSIA







Basically we predict *condensed graphs of reactions.*
So even *side products* are predicted naturally.

Conclusion / Summary

USI/SUPSI
Istituto
Dalle Molle
di studi
sull'intelligenza
artificiale
IDSIA

1). The NERF framework established a new state of the art in reaction prediction.
2). The model yields interpretable predictions and naturally predicts side products.
3). It is faster on inference compared to previous SOTA models.

> It's architecture is mostly transformer, but over atom embeddings.
> The attention mechanism is ubiqitous in the architecture, even the output is att. scores.
> It uses conditional variational autoencoder as a sublayer to model the distribution of product graphs given the reactant graphs.

> It can be probably used as a framework for other models, e.g. for retrosythesis prediction.

Thank you for your $\text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$