Predicting Chemical Reaction Type and Reaction Products with Recurrent Neural Networks

Anvita Gupta {avgupta}@stanford.edu

Abstract

- Synthesis of chemical molecules is crucial in medicine, environmental science, materials, and more disciplines. However, predicting the outcomes of synthesis reactions is low accuracy, & mistakes are costly and time consuming.
- Here, we use deep **recurrent and convolutional neural networks** to
  1. **Classify the type of a chemical reaction given the reactants.**
  2. **Generate the correct products given the reactant**s
- Reaction prediction classifier achieves >90% accuracy and AUPRC
- Generated products have high degree of chemical similarity to actual products, and we outperform the oracle (rule based prediction).

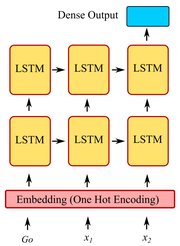
Data

- **50,000 Curated Reactions** from US Patents - **across 50 Reaction Types**. Reactions represented as strings using the SMILES grammar [1]

Data Preprocessing and Featurization

- Removal of Atom Mapping, **Canonicalize & Sort Reactants**
- Preprocessing by Length - Middle 80% of Data was retained by length, no reactions where |react| < 30 char.
- Removal of Salts - Left over from synthesis conditions (to stabilize molecule in some state); unnecessary for core reaction
- Tokenization- **54 total tokens**, each an element or number. "G" and "E" are sentinels for "start" and "end", respectively.

Models: Reaction Type Classification



- Baseline: Logistic Regression (Weighted for Class Imbalance)
  - Features: Fingerprint, bitstring with presence of substructures
- Recurrent Neural Network Classifier
  - **Learn features directly from SMILES string!**
  - Two LSTM Layers with Hidden State = 512 x 1
  - Batch Norm, Dropout = 0.3 (Layer 1), 0.5 (Layer 2)
  - Output Dense Layer for probability of 50 types
- Data split 60% Train, 20% Validation, 20% Test Set

Figure 1: Architecture of Recurrent Neural Network for Reaction Type Classification.

Results

Reaction Type Classification

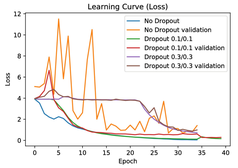


Figure 5: Learning curves for classifiers with varying dropout rates.

Model	Train Accuracy	Test Acc	Precision (Macro)	Recall (Macro)	AUPRC (Micro)
No Dropout	0.988	0.9083	0.9131	0.9076	0.96
Dropout 0.1/0.1	0.951	0.9591	0.9596	0.9589	0.99
Dropout 0.3/0.3	0.777	0.7607	0.7903	0.7605	0.85
Dropout 0.3/0.5	0.8393	0.8834	0.8907	0.8777	0.95
Logistic	0.8740	0.8277	0.8231	0.8263	0.65

Table 1: Training accuracy, Test Accuracy, Precision, Recall, and AUPRC. Precision and recall calculated with macro-averages, and AUPRC with micro. Macro and micro averages were in smiliar range.

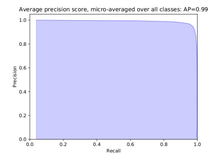


Figure 6: Precision Recall Curve (AUPRC = AP = 0,99) for best performing classifier.

Reaction Product Generation

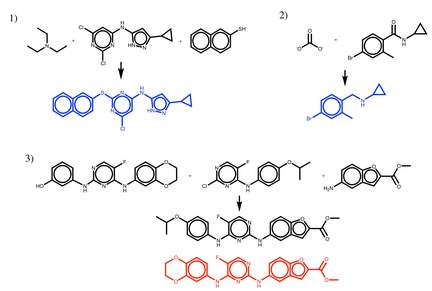


Figure 7: Sampled reactions from Seq2Seq, Attn, RNN Encoder model. Product is blue when predicted output is correct. In reaction 3, the target is black and the predicted product is red.

Model	Training Loss (20k iteration)	% Valid Molecules	Tanimoto Similarity
Baseline: N-grams per Rxn Type	N/A	7.27%	0.0946
Vanilla Seq2Seq	1.148	86.00%	0.6355
Attn Model (RNN Encoder/Decoder)	0.2772	83.33%	0.8788
Attn Model (CNN Encoder/RNN Decoder)	1.409	72.00%	0.5176
Oracle:Rule Based System	N/A	100.00%	0.8079

Table 2: Training loss (measured after 20k iterations to capture speed of convergence), Percentage Valid SMILES strings, and Tanimoto coefficient. Validity and Tanimoto coefficients averaged for last 5 samples to reduce noise.

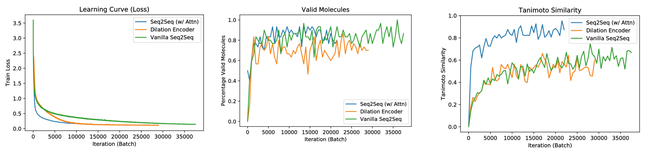


Figure 8 : Learning curves for three models. X axis shows iterations (3800 iterations per epoch). Left: Training Loss. All models trained until convergence. Middle: Percentage Valid sampled molecules. Right: Tanimoto Similarity for sampled molecules.

Discussion

- **Reaction Type Classifier (Dropout 0.1/0.1) has >95% accuracy and AUPRC of 0.99**, significantly outperforms baseline
- **Diagnosed overfitting** and improved generalization by **varying levels of dropout** (Fig 8)
- **Error Analysis - Sought to understand performance as function of class/imbalance**. AUPRC above 0.9 for all classes (Fig 10), not significantly worse for imbalanced classes (Fig 9).

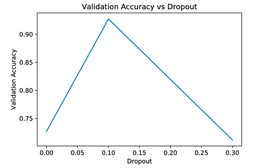


Figure 8: Validation accuracy for varying dropout levels (same dropout both layers)

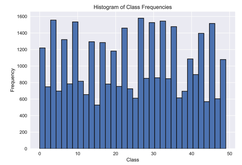


Figure 9: Frequencies of different classes of reaction types, with 50 classes total.

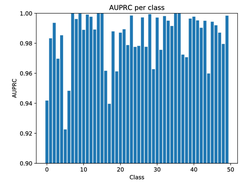


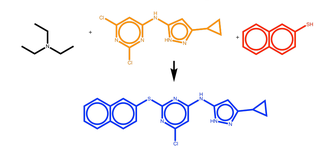
Figure 10: AUPRCs of different classes of reaction types, with 50 classes total.

- **Seq2Seq with Attention (RNN Encoder/Decoder)** is the best performing of the three models designed; **Tanimoto similarity averages 0.86 and reaches at 0.95**, **higher than "rule based" system** (0.80 tanimoto)!
- Why? Rule-based extraction needs correct extraction of reaction centers (place where reaction is taking place); requires a great deal of domain knowledge, libraries implementing this (rdkit) are buggy.
- Figure 5 shows that even on incorrect outputs, **the predicted output shows motifs from the actual output**
- Inaccuracy is higher on larger output molecules (as expected)
- Seq2Seq (CNN Encoder) surprisingly did not reach the same level of performance (or outperform Vanilla Seq2Seq)

Conclusion

Contributions

- Completely **data driven (rule independent) system for predicting reaction type and products, given reactants**; important as hard-coded rules have many exceptions and require a great deal of domain knowledge!
- **Reaction type prediction significantly improves upon existing models** in [1] that use reaction fingerprints
- Predicted products are **highly chemically similar to actual products**; useful for synthetic chemists to plan reaction syntheses
- Application of **dilated CNNs in the Encoder** - need for padding increases noise (see Future work)



CCN(CC)CC.Clc1cc(Nc2cc(C3CC3)n[nH]2)nc(Cl)n1.Sc1ccc2ccccc2c1>>Clc1cc(Nc2cc(C3CC3)n[nH]2)nc(Sc2ccc3ccccc3c2)n1

Figure 11: Reaction and its corresponding (preprocessed) SMILES string

Future Work

- **Combining reaction type with reactants to improve product prediction**
- **Redesigning CNN encoder architecture** without padding to improve performance
- **Interpretability** - check whether model's learned rules align with human intuition

Models: Reaction Product Generation

- **Baseline:** Ngrams model with n=5, frequencies computed per reaction type
- **"Oracle":** Rule-Based Prediction- Extraction of reaction center

Vanilla Seq2Seq Architecture

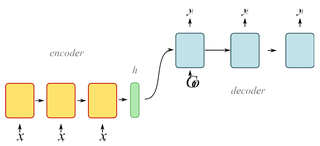


Figure 2: Vanilla Seq2Seq Architecture. Encoder takes in reactants and decoder outputs product sequence.

- Encoder: reactants input, latent state output
- Decoder: takes in latent state and reproduces products
- Both encoder and decoder have 2 GRU Layers
- Hidden state size 500 on which decoder is conditioned

- Challenge: **long length (226 max)**! Dependency on reactants goes down.
- High correlation between inputs/output (identity mapping), hard to learn with Vanilla.

- ***Implementation-*** Models coded in **Pytorch** (Vanilla Seq2Seq also implemented in **Tensorflow** with padding). Chemical analyses performed with **Rdkit library**.
- Each model trained on **Nvidia Tesla K80 GPU for 10-12 hours** (1 Azure Instance)

Seq2Seq with Attention (RNN Encoder)

- Encoder - Bidirectional GRU, 2 Layers, Dropout = 0.05, Hidden State=500
- Decoder - 2 GRU Layers; Hidden State=500
- Attention: At each step, decoder outputs "query".
- Attn Weights = softmax("query" x encoder_output)
- **Rxn type optionally added as special token before input**

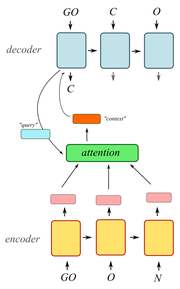
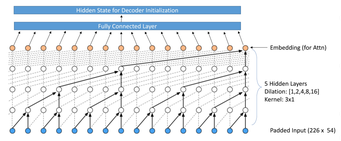


Figure 3: Architecture of Seq2Seq Model (RNN Encoder/Decoder + Attention).

Seq2Seq (Attn, CNN Encoder)



- Encoder is now (CNN); 5 layers with dilations [1,2,4,8,16]
- Dilated Convolutions exponentially increase receptive field size
- More info without vanishing gradients
- Wavenet (audio gen) & Bytenet (translation)

Figure 4: CNN encoder with dilated convolutions. Convolved output used as the encoder representation; hidden layer at final time step produces hidden state of the decoder.

References

[1] Nadine Schneider, Daniel M. Lowe, Roger A. Sayle, and Gregory A. Landrum. Development of a novel fingerprint for chemical reactions and its application to large-scale reaction classification and similarity.Journal of Chemical Information and Modeling, 55(1):39–53, 2015. PMID: 25541888

[2] Aäron van den Oord et al.Wavenet: A generative model for raw audio. CoRR, abs/1609.03499, 2016.

[3] Nal Kalchbrenner et al. Neural machine translation in linear time. CoRR,abs/1610.10099,2016.

I would like to acknowledge Microsoft for the Azure compute credit.