Resources:

1. Scheduled zoom link https://illinois.zoom.us/j/823397955
   以后都会使用此链接。
2. Google Calendar
   https://calendar.google.com/event?action=TEMPLATE&tmeid=ZXRyaDlsZnFkNTg0ajRhMzh1MG5jZjNndWtfMjAyMDAzMDNUMjEzMDAwWiBndWFueWluZ2p1bi5kYXRqdUBt&tmsrc=guanyingjun.datju%40gmail.com&scp=ALL
3. GitHub links: [data mining read group]
4. Course webpage: http://www.phontron.com/class/nn4nlp2020/schedule.html

---

03/05 Group Study - Lecture 8: Embedding

Attendance: Yingjun Guan, Yixing Hu, Haoran Zhang, Ziqi Jiang, Yiren Liu, Lan Wang, Gaozheng Liu
Leading: Yixing Hu
Course Material:

Discussion Notes:
[Q] P3: What does it mean for initially random
[D] P3: Bridging Supervised learning and Unsupervised learning.
[D] P6: projection between large dataset -> a way to solve the unlabeled/ limited training
[D] P10: connecting Distributional / Non-Distributional to Inductive/Deductive modeling.
[D] P10: differences on Distributed/ Non-Distributed representation
https://www.quora.com/Deep-Learning-What-is-meant-by-a-distributed-representation
https://dl.acm.org/doi/10.5555/1858681.1858721
https://www.morganclaypool.com/doi/abs/10.2200/S00762ED1V01Y201703HLT037
Chapter 10.4

Pointwise mutual information (**PMI**),[1] or **point mutual information**, is a measure of association used in information theory and statistics. In contrast to mutual information (MI) which builds upon PMI, it refers to single events, whereas MI refers to the average of all possible events.

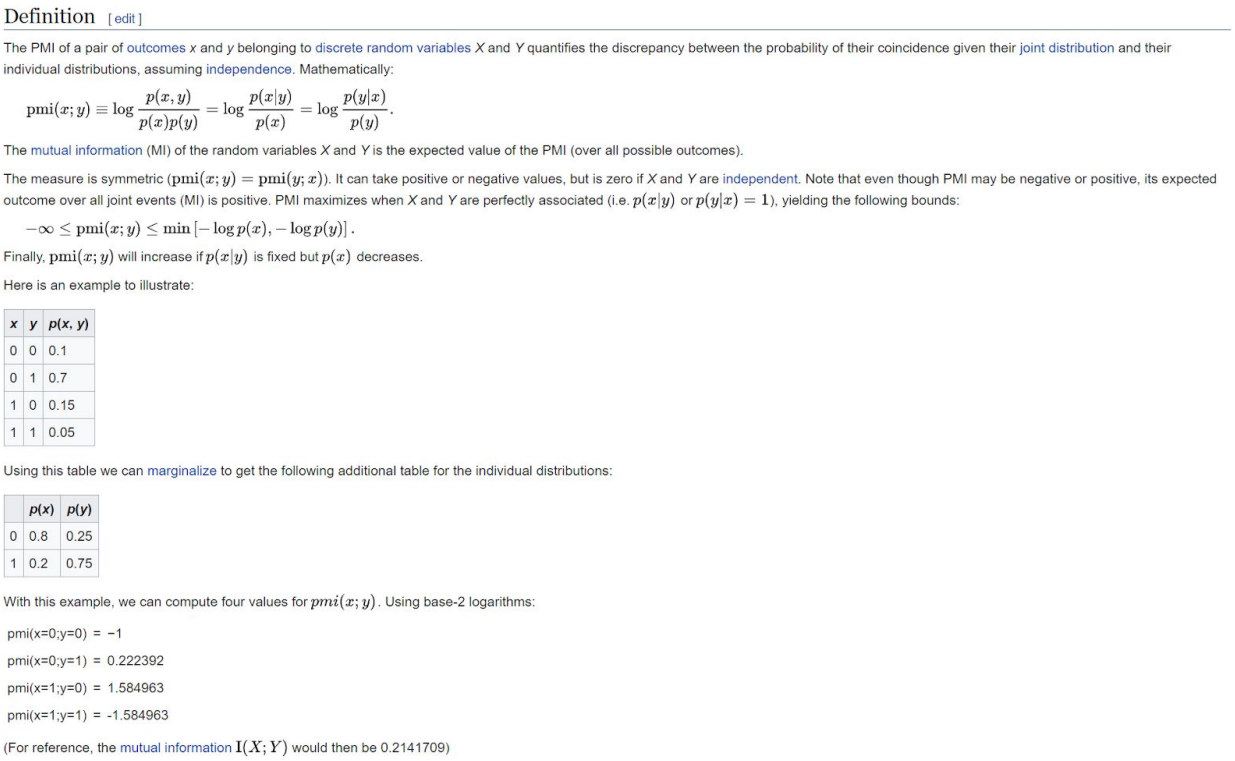
Paper on GloVe: https://www.aclweb.org/anthology/D14-1162.pdf

Paper on Retrofitting of Embeddings:
https://www.cs.cmu.edu/~hovy/papers/15HLT-retrofitting-word-vectors.pdf

Paper on Sparse Embeddings:
https://www.aclweb.org/anthology/C12-1118.pdf

Paper on De-biasing word embeddings:

https://papers.nips.cc/paper/6228-man-is-to-computer-programmer-as-woman-is-to-homemaker-debiasing-word-embeddings.pdf



Definition [edit]

The PMI of a pair of outcomes x and y belonging to discrete random variables X and Y quantifies the discrepancy between the probability of their coincidence given their joint distribution and their individual distributions, assuming independence. Mathematically:

$$\operatorname{pmi}(x;y) \equiv \log \frac{p(x,y)}{p(x)p(y)} = \log \frac{p(x|y)}{p(x)} = \log \frac{p(y|x)}{p(y)}.$$

The mutual information (MI) of the random variables X and Y is the expected value of the PMI (over all possible outcomes).

The measure is symmetric ($\operatorname{pmi}(x;y) = \operatorname{pmi}(y;x)$). It can take positive or negative values, but is zero if X and Y are independent. Note that even though PMI may be negative or positive, its expected outcome over all joint events (MI) is positive. PMI maximizes when X and Y are perfectly associated (i.e. $p(x|y)$ or $p(y|x) = 1$), yielding the following bounds:

$$-\infty \leq \operatorname{pmi}(x;y) \leq \min\left[-\log p(x), -\log p(y)\right].$$

Finally, $\operatorname{pmi}(x;y)$ will increase if $p(x|y)$ is fixed but $p(x)$ decreases.

Here is an example to illustrate:

x	y	p(x, y)
0	0	0.1
0	1	0.7
1	0	0.15
1	1	0.05

Using this table we can marginalize to get the following additional table for the individual distributions:

	p(x)	p(y)
0	0.8	0.25
1	0.2	0.75

With this example, we can compute four values for $pmi(x;y)$. Using base-2 logarithms:

pmi(x=0;y=0) = −1

pmi(x=0;y=1) = 0.222392

pmi(x=1;y=0) = 1.584963

pmi(x=1;y=1) = -1.584963

(For reference, the mutual information $\mathrm{I}(X;Y)$ would then be 0.2141709)

t-SNE viz:
https://distill.pub/2016/misread-tsne/?_ga=2.135835192.888864733.1531353600-1779571267.1531353600
Try the intro below (including some hands-on practice of t-SNE on PYTHON)
https://medium.com/@violante.andre/an-introduction-to-t-sne-with-python-example-47e6ae7dc58f

[Q] P32: Why is it basic useless in Language modeling?

03/03 Group Study - Lecture 7: Attention

Attendance: Yingjun Guan, Haoran Zhang, Mengfei Lan, Yixing Hu, Lan Li, Lan Wang
Leading: Yiren Liu
Course Material:
- 辅助ppt: [暂略]
(https://docs.google.com/presentation/d/1-zGY-VR5ryhKlPALJLE6oVe5s1AEyAG8t9blxESl4jM/edit?usp=sharing)

- 视频学习 https://www.youtube.com/watch?v=jDaJYOmF2iQ
- 浏览课件 http://phontron.com/class/nn4nlp2020/assets/slides/nn4nlp-07-attention.pdf
- python代码 https://github.com/neubig/nn4nlp-code/tree/master/09-attention
- 阅读材料 Neural Machine Translation and Sequence-to-Sequence Models Chapter 8
- 如果你对Attention的基础知识不很了解，请移步这里的其他资源
http://phontron.com/class/nn4nlp2020/schedule/attention.html

Discussion Notes:

- Alignment - has to be on the same-dimension vector
- How about using zero padding [or negative infinity padding] to fix dimensions problem?
- New paper recommendation: Multi-head attention. {forcing multiple attention learning different things}
- How about word to phrase translation. [One to many, rather than one to one.]
    - Try score vs sum of score
    - Try parsing first
- Pointer indicates a more local information, while vocabulary to a more global info.
- [Q] is relationship pairwised? Check the embedding heat table [on Slides P4]
- Coverage [one attention] vs multi-head [multi attention]
- [Q] bidirectional training [should results on both directions intuitively the same?]
- [Q] Hard attention application - Reading Comprehension & QA etc.
- [D] benefits on masking while training. Strictly forbids cheating on foreseeing future info.
- [Q] QKV application (different K and V), check paper on Vaswani 2017
- [D] relative embedding 2.0 function in Pytorch.
-

D: Discussion; Q - Question

02/27 Group Study - Lecture 6: Conditioned Generation

Attendance: Yingjun Guan, Haoran Zhang, Ziqi Jiang, Yixing Hu, Lan Li, Yiren Liu, Liri Fang
Leading: Ziqi Jiang
Course Material:
- 视频学习 https://www.youtube.com/watch?v=Og3_LngQE_4&t=2s
- 浏览课件 http://www.phontron.com/class/nn4nlp2020/assets/slides/nn4nlp-06-condlm.pdf

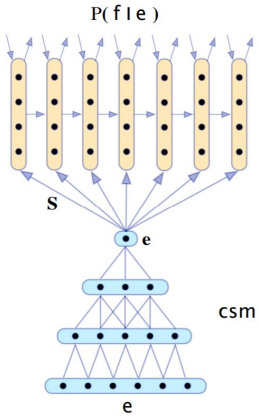
- python代码 https://github.com/neubig/nn4nlp-code/tree/master/08-condlm
- 阅读材料 Neural Machine Translation and Sequence-to-Sequence Models Chapter 7； 本章其他文章都很不错，鼓励大家积极阅读和讨论。
- 如果你对Encoder-Decoder Models, Conditional Genreation and Search, Ensembling, Evaluation, Types of Data to Condition On 的基础知识不很了解，请移步这里的其他资源
http://www.phontron.com/class/nn4nlp2020/schedule/conditioned-lm.html

Discussion Notes:

Q: [P6] What's the condition? Example? Understanding?
A: Check P3.

Q: [P9] Why / How could the hidden layer include the information of all the previous information.
A: Check basic information on encoder.

Q: [P9, method 3]



$$\mathbf{s} = \mathbf{S} \cdot \mathrm{csm}(\mathrm{e}) \tag{8a}$$
$$h_1 = \sigma(\mathbf{I} \cdot \mathrm{v}(\mathrm{f}_1) + \mathbf{s}) \tag{8b}$$
$$h_{i+1} = \sigma(\mathbf{R} \cdot h_i + \mathbf{I} \cdot \mathrm{v}(\mathrm{f}_{i+1}) + \mathbf{s}) \tag{8c}$$
$$o_{i+1} = \mathbf{O} \cdot h_i \tag{8d}$$

Q: [P11] How to randomly generate?
A: Good example: A-0.1, B-0.2, C-0.6. 按照概率，ABC都有可能生成。

Q: [P7] idea starting from Ensembling: Maybe we could compare models; calculate the correlationship
A: [Haoran] judge whether two models are both learning effectively on the correct feature. There's a recent paper comparing

Q: [P22] Video about sequence level knowledge distillation.
https://vimeo.com/239248521

Q: stacking, using prediction from different models as input to train.

Lan:
Q: [P9] How to pass the hidden state? Three ways

Q: Compare greedy search (whole) and argmax search (Time stamp)

Q: Viterbi may share a similar idea with argmax [comparison as below]

Probabilistic CKY: Viterbi

Like standard CKY, but with probabilities.
Finding the most likely tree is similar to Viterbi for HMMs:

Initialization:

– [*optional*] Every chart entry that corresponds to a **terminal** (entry w in `cell[i][i]`) has a Viterbi probability $P_{VIT}(w_{[i][i]}) = 1$ (*)

– Every entry for a **non-terminal** `X` in `cell[i][i]` has Viterbi probability $P_{VIT}(X_{[i][i]}) = P(X \rightarrow w \mid X)$ [and a single backpointer to $w_{[i][i]}$ (*)]

Recurrence: For every entry that corresponds to a **non-terminal** `X` in `cell[i][j]`, keep only the highest-scoring pair of backpointers to any pair of children (`Y` in `cell[i][k]` and `Z` in `cell[k+1][j]`):
$P_{VIT}(X_{[i][j]}) = \text{argmax}_{Y,Z,k}\ P_{VIT}(Y_{[i][k]}) \times P_{VIT}(Z_{[k+1][j]}) \times P(X \rightarrow Y\ Z \mid X)$

Final step: Return the Viterbi parse for the start symbol S in the top `cell[1][n]`.

*this is unnecessary for simple PCFGs, but can be helpful for more complex probability models

The Viterbi algorithm

A dynamic programming algorithm which finds the best (=most probable) tag sequence $\mathbf{t}^*$ for an input sentence $\mathbf{w}$: $\mathbf{t}^* = \text{argmax}_{\mathbf{t}}\ P(\mathbf{w} \mid \mathbf{t})P(\mathbf{t})$

Complexity: linear in the sentence length.
With a bigram HMM, Viterbi runs in $O(T^2N)$ steps for an input sentence with N words and a tag set of T tags.

The independence assumptions of the HMM tell us how to break up the big search problem (find $\mathbf{t}^* = \text{argmax}_{\mathbf{t}}\ P(\mathbf{w} \mid \mathbf{t})P(\mathbf{t})$) into smaller subproblems.

The data structure used to store the solution of these subproblems is the trellis.

* Similarity: global optimum, linear sequential structure.
* Difference: Viterbi 对长度有要求【输入输出必须相同】；MT 不一定。

function *VITERBI*$(O, S, \Pi, Y, A, B) : X$
    for each state $i = 1, 2, \ldots, K$ **do**
        $T_1[i, 1] \leftarrow \pi_i \cdot B_{iy_1}$
        $T_2[i, 1] \leftarrow 0$
    end for
    for each observation $j = 2, 3, \ldots, T$ **do**
        for each state $i = 1, 2, \ldots, K$ **do**
            $T_1[i, j] \leftarrow \max_k (T_1[k, j-1] \cdot A_{ki} \cdot B_{iy_j})$
            $T_2[i, j] \leftarrow \arg\max_k (T_1[k, j-1] \cdot A_{ki} \cdot B_{iy_j})$
        end for
    end for
    $z_T \leftarrow \arg\max_k (T_1[k, T])$
    $x_T \leftarrow s_{z_T}$
    for $j = T, T-1, \ldots, 2$ **do**
        $z_{j-1} \leftarrow T_2[z_j, j]$
        $x_{j-1} \leftarrow s_{z_{j-1}}$
    end for
    return X
end function

Probabilistic CKY

Input: POS-tagged sentence

John_N eats_V pie_N with_P cream_N

John	eats	pie	with	cream	
Noun NP 1.0 0.2	S 0.8·0.2·0.3	S 0.8·0.2·0.06		S 0.2·0.0036·0.8	John
	Verb VP 1.0 0.3	VP 1·0.3·0.2 = 0.06		VP max(1.0 ·0.008·0.3, 0.06·0.2·0.3)	eats
		Noun NP 1.0 0.2		NP 0.2·0.2·0.2 = 0.008	pie
			Prep 1.0	PP 1·1·0.2	with
				Noun NP 1.0 0.2	cream

```
S     → NP VP          0.8
S     → S conj S       0.2
NP    → Noun           0.2
NP    → Det Noun       0.4
NP    → NP PP          0.2
NP    → NP conj NP     0.2
VP    → Verb           0.3
VP    → Verb NP        0.3
VP    → Verb NP NP     0.1
VP    → VP PP          0.3
PP    → Prep NP        1.0
Prep  → P              1.0
Noun  → N              1.0
Verb  → V              1.0
```


Haoran: teacher forcing
Exposure bias
: bi-directional encoder
XL-NET : (maybe useful) permutation model

Argmax & attention : compare

02/25 Group Study - Lecture 5: Efficiency Tricks

Attendance: Yingjun Guan,Haoran Zhang, Yixing Hu, Yiren Liu, Xiaoliang Jiang, Mengfei Lan

Course Material:
- 辅助ppt: [暂略]
- 视频学习 https://www.youtube.com/watch?v=eokkF3qv8_U
- 浏览课件 http://www.phontron.com/class/nn4nlp2020/assets/slides/nn4nlp-05-efficiency.pdf
- python代码 https://github.com/neubig/nn4nlp-code/tree/master/04-efficiency
- 阅读材料 Notes on Noise Contrastive Estimation and Negative Sampling (Dyer 2014)
- 如果你对softmax approximations; parallel training; GPU training 的基础知识不很了解，请移步这里的其他资源 http://www.phontron.com/class/nn4nlp2020/schedule/efficiency.html

Discussion Notes:
Yingjun Guan:

1. GPU resources (待补充)：Amazon（定量免费）， Microsoft (按量付费), Google Colab (学生定量免费)
2. Negative sampling 中negative 部分的定义？
3. NCE 与negative sampling 比较与介绍文章： https://blog.zakjost.com/post/nce-intro/
4.

Haoran Zhang:

1. What's so-called good "operation"?
    a. +/-/*/\
    b. able to be parallelized
    c. Appropriate for matrix calculation.
2. 数据并行 vs 程序并行
3. GPU目前最大单卡可达48GB

Liri Fang:

1. Mini batch vs 数据并行？
2. 数据并行和程序并行可以同时使用。

Mengfei Lan:

02/20 Group Study - Lecture 4: RNN for text

Attendance: Lan Li, Lan Jiang, Yingjun Guan, Ziqi Jiang, Yiren Liu, Yixing Hu, Xiaoliang Jiang

Course Material:
- 辅助ppt: [RNN]

- 视频学习 https://youtu.be/aA14_J-8HXU
- 浏览课件 http://www.phontron.com/class/nn4nlp2020/assets/slides/nn4nlp-04-rnn.pdf
- 阅读材料 GoldBerg's NNM4NLP Chap14-16
https://www.morganclaypool.com/doi/abs/10.2200/S00762ED1V01Y201703HLT037
- 如果你对Recurrent Networks, Vanishing Gradient and LSTMs, Strengths and Weaknesses of Recurrence in Sentence Modeling, Pre-training for RNNs的基础知识不很了解，请移步这里的其他资源 http://www.phontron.com/class/nn4nlp2020/schedule/rnn.html

Study group adjustments (for future).

1. Group members will lead discussion in turn (presenting in teams is ok).
    a. Some future leadings: {Conditioned generation: Ziqi, Attention: Yiren}
2. Codes are suggested in future lectures to help better understand the material. According to the discussion, the group will spend 5-10 min on codes. Members should try running codes before class.
3. Reorganizing the slides like today is highly recommended, but not required.
4. We could start using github to share interesting codes, tests, projects, etc.
5. Responsibility for presenters:
    a. [Required] Lead discussion; ask questions; run codes successfully and lead.
    b. [suggested] make slides; read and summarize readings; recommend other relative materials.
6. Responsibility for attenders:
    a. [Required] Watch youtube videos; check slides; try to run codes.
    b. [suggested] Reading; deep understanding; sharing interesting projects and ideas.

Discussion Notes:

Why RNN….

Liu: sequential + long-distance
Lan Jiang: coreference : 同样词指代不同东西

Loss function ?

Regression Loss Functions:
Mean Squared Error Loss
Mean Squared Logarithmic Error Loss
Mean Absolute Error Loss

Binary Classification Loss Functions:
Binary Cross-Entropy
Hinge Loss
Squared Hinge Loss

Multi-Class Classification Loss Functions:
Multi-Class Cross-Entropy Loss

Sparse Multiclass Cross-Entropy Loss
Kullback Leibler Divergence Loss

Total loss→ separate loss

Backpropagation的目的：调整separate loss.

Lan:
Wish to discuss Attention in more detail in future. (Yes. In two more weeks, we will do.)

Ziqi jiang:
Shared Parameter & Accumulated derivatives
Updating process: total loss→ loss i (where i = 1,2,...,m) → derivative^(i) → use sum(derivative^(i)) update the weight
Advantage:

1. Updating parameters for the whole sentence
2. Computational complexity O(n) while updating in every step would require O(n^2) complexity.

02/18 Group Study - Lecture 3: CNN for Text

Attendance: Yiren Liu, Yingjun Guan, Liri Fang, Lan Li, Yixing Hu, Mengfei Lan, Gaozheng Liu.

Course Material
第三课： CNN for text
- 领讲：房丽日
- 视频学习
- 浏览课件
- 阅读材料 GoldBerg's NNM4NLP Chap13
- 如果你对n-gram, CNN, structured convolution, CNN visualization的基础知识不很了解，请移步这里的其他资源

Discussion Notes:
Fang:

Concept: Pooling

- **Pooling** is an aggregation operation, aiming to select informative features
- **Max pooling:** "Did you see this feature anywhere in the range?" (most common)
- Average pooling: "How prevalent is this feature over the entire range"
- **k-Max pooling:** "Did you see this feature up to k times?"
- **Dynamic pooling:** "Did you see this feature in the beginning? In the middle? In the end?"

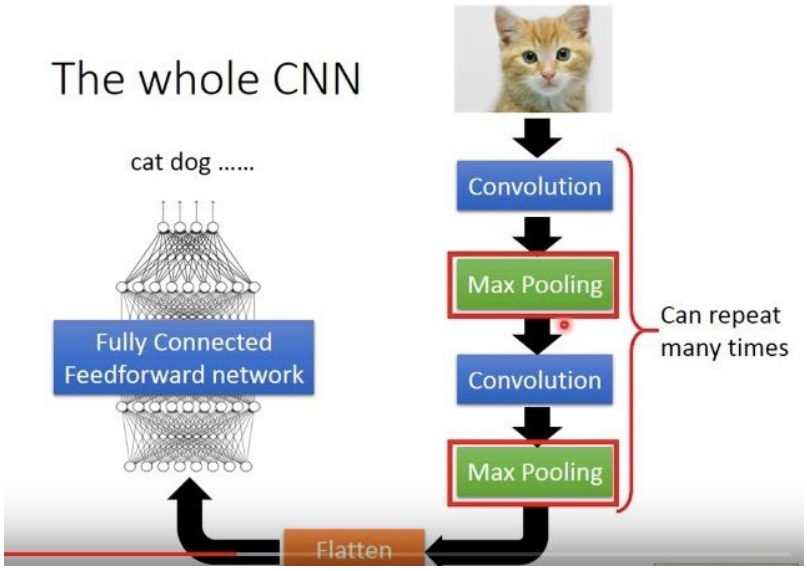
Workflow for NLP: (**Guan**)
Embedding (what's embedding?)→ Training → Testing
Related work:
BERT
General : Fine-tuning and Feature-based

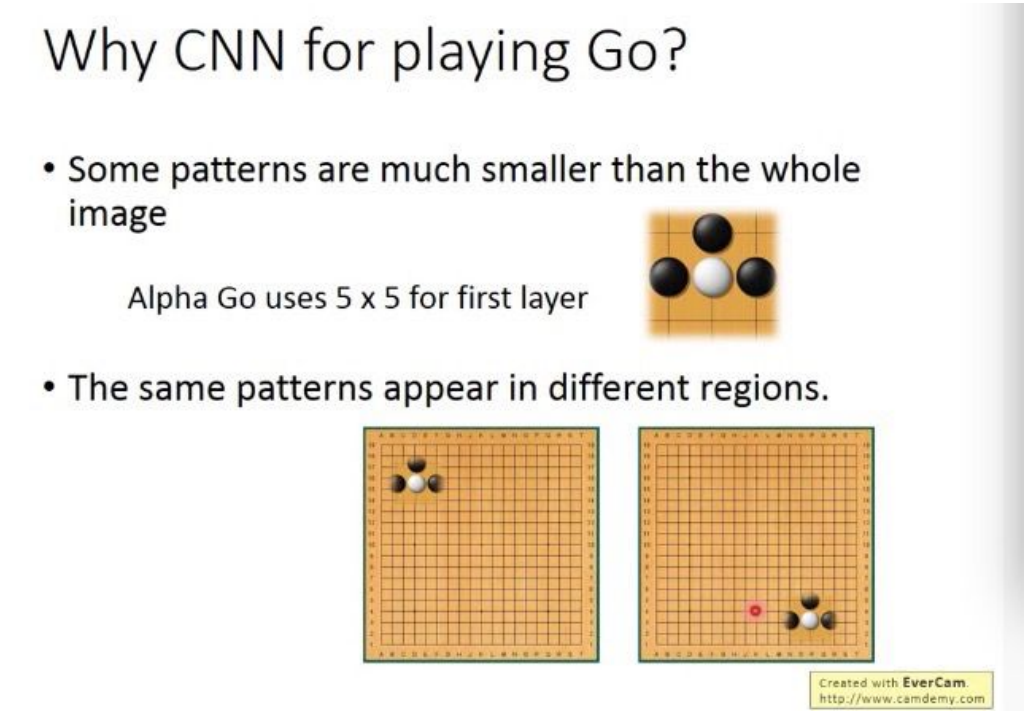
Yiren Liu: suggested : first fine-tuning with BERT, then training
(https://github.com/google-research/bert)

> Using BERT has two stages: *Pre-training* and *fine-tuning*.
>
> **Pre-training** is fairly expensive (four days on 4 to 16 Cloud TPUs), but is a one-time procedure for each language (current models are English-only, but multilingual models will be released in the near future). We are releasing a number of pre-trained models from the paper which were pre-trained at Google. Most NLP researchers will never need to pre-train their own model from scratch.
>
> **Fine-tuning** is inexpensive. All of the results in the paper can be replicated in at most 1 hour on a single Cloud TPU, or a few hours on a GPU, starting from the exact same pre-trained model. SQuAD, for example, can be trained in around 30 minutes on a single Cloud TPU to achieve a Dev F1 score of 91.0%, which is the single system state-of-the-art.

Why padding? For boundary :
If the final number for the filter is 0, the last will not be sampled...

Lan: **https://www.youtube.com/watch?v=FrKWiRv254g**





Mengfei:
In NLP, can vector be parsed and CNNed?

02/13 Group Study - Lecture 2: Language Modeling

Attendance: Yingjun Guan, Xiaoliang Jiang, Ziqi Jiang, Lan Li, Liri Fang, Yiren Liu
Notification:

5. Scheduled zoom link
   https://zoom.us/j/400853409 以后都会使用此链接。
6. Google Calendar
   https://calendar.google.com/event?action=TEMPLATE&tmeid=N2Zvam11djVhYXFrZTRrMDhydnFjMzNpbWRfMjAyMDAyMTNUMjEzMDAwWiBndWFueWluZ2p1bi5kYXRqdUBt&tmsrc=guanyingjun.datju%40gmail.com&scp=ALL
7. Other **Useful links: [data mining read group]**

Course material:
第二课： language modeling.
- 视频学习
- 浏览课件
- 阅读资料 GoldBerg's NNM4NLP chap8-9
- 如果你对language modeling, feed-forward NN, optimization, language model evaluation的基础知识不熟悉的话，请移步这里的其他资源：

Discussion Notes:
Lan:
- potential related work：[Attention is all you need]

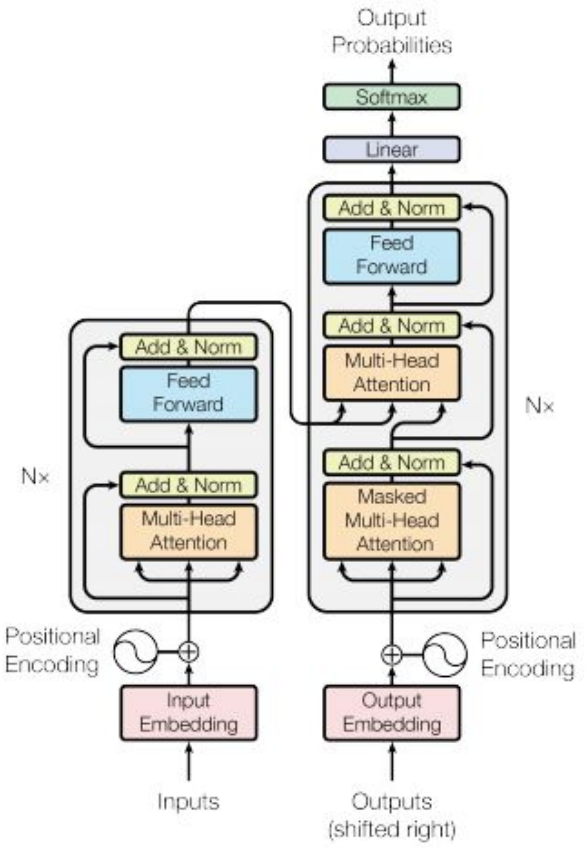


Figure 1: The Transformer - model architecture.

- Add on positional parameter
- Improve computation efficiency : parallelisation
- Page 6: self-attention could help with long-range dependencies
- The third is the path length between long-range dependencies in the network. Learning long-range dependencies is a key challenge in many sequence transduction tasks. One key factor affecting the ability to learn such dependencies is the length of the paths forward and backward signals have to traverse in the network. The shorter these paths between any combination of positions in the input and output sequences, the easier it is to learn long-range dependencies [12]. Hence we also compare the maximum path length between any two input and output positions in networks composed of the different layer types.

Xiaoliang:

1. **Count-based language modeling**
2. Bow, sequence/sequential data: the word after A and B vs the third word after 1st 2nd and before 4th 5th… ->speed up
3. **Generative model &&**
4. **Smoothing** https://en.wikipedia.org/wiki/Kneser%E2%80%93Ney_smoothing

5. **Back-off smoothing** **https://en.wikipedia.org/wiki/Katz%27s_back-off_model**
6. Log加和概率乘积
7. Loglikelihood: 整句， Per-word Log Likelihood: 除词个数， 词，

Ziqi
P26: What are Input Words, Output Words, context?

Liri:
Q: 同时生成第3词和第5词 paper:

- Non-Monotonic Sequential Text Generation by KH Cho, etc.

N: video on Momentum: https://www.youtube.com/watch?v=k8fTYJPd3_I

02/12 CBOW discussion

CBOW: page 117-118 [chap8-9]
CBOW is very similar to the traditional bag-of-words representation in which we discard order information, and works by either summing or averaging the embedding vectors of the corresponding features:

$$\text{CBOW}(f_1, \ldots, f_k) = \frac{1}{k} \sum_{i=1}^{k} v(f_i). \tag{8.1}$$

A simple variation on the CBOW representation is weighted CBOW, in which different vectors receive different weights:

$$\text{WCBOW}(f_1, \ldots, f_k) = \frac{1}{\sum_{i=1}^{k} a_i} \sum_{i=1}^{k} a_i v(f_i). \tag{8.2}$$

Here, each feature f_i has an associated weight a_i, indicating the relative importance of the feature. For example, in a document classification task, a feature f_i may correspond to a word in the document, and the associated weight a_i could be the word's TF-IDF score.

02/11 Group study - Lecture 1: General NLP

Attendence: Yingjun Guan, Liri Fang, Mengfei Lan, Yixing Hu, Yiren Liu, Xiaoliang Jiang, Lan Li

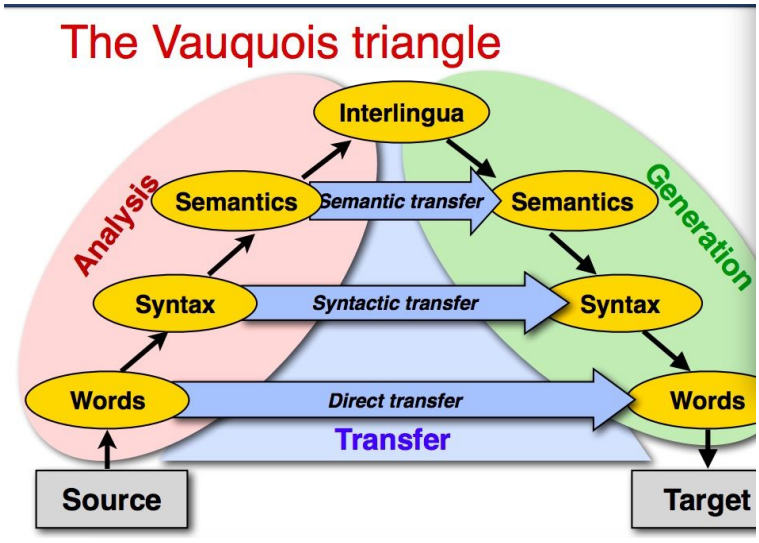
1. Self introduction.
2. http://phontron.com/class/nn4nlp2020/description.html

3. L1 revision. (http://phontron.com/class/nn4nlp2020/assets/slides/nn4nlp-01-intro.pdf)
4. L2 revision.
5. Phenomena to Handle
    a. **Morphology** eg. go/goed
    b. **Syntax** 句法？
        i. 主谓宾顺序？
        ii. 主谓宾缺失？
    c. **Semantics/World Knowledge**: eg. The store went to Jane.
    d. **Discourse**:
        i. 1.远距离？答非所问？Q：吃了啥 A:管老师好
        ii. 2. 代词？
    e. Pragmatics

    Pragmatic Analysis is part of the process of extracting information from text. Specifically, it's the portion that focuses on taking structures set of text and figuring out what the actual meaning was. It actually comes from the field of linguistics (as a lot of **NLP** does), where the context is considered from the text. → google search

    [dictionary]: the branch of linguistics dealing with language in use and the contexts in which it is used, including such matters as deixis, the taking of turns in conversation, text organization, presupposition, and implicature.

    f. Multilinguality



Lanjie 提到的三角形

Liri Fang: dialect → ?

Xiaoliang: 我就来记几个专业名词

1. **Sentence Classification**
2. **Computation Graphs**

3. **Forward propagation; backward propagation.**
4. **Keras**
5. **AutoGrad**
6.

BOW vs CBOW vs deepCBOW

6. Study plan in future:
Time: Tuesday, Thursday 3:30-4:40pm

https://github.com/uiuc-dm-group/DMRG-20SP