

Spell Correction:

Problem 1:

Misspelled: sirrow, Option: swallow

w	6	5	4	5	6	7	8	7
o	5	4	5	6	7	8	7	8
r	4	3	4	5	6	7	8	9
r	3	2	3	4	5	6	7	8
i	2	1	2	3	4	5	6	7
s	1	0	1	2	3	4	5	6
#	0	1	2	3	4	5	6	7
	#	s	w	a	l	l	o	w

Misspelled: sirrow, Option: sorrow

w	6	5	4	5	4	3	2
o	5	4	3	4	3	2	3
r	4	3	4	3	2	3	4
r	3	2	3	2	3	4	5
i	2	1	2	3	4	5	6
s	1	0	1	2	3	4	5
#	0	1	2	3	4	5	6
	#	s	o	r	r	o	w

Misspelled: sirrow, Option: serow

w	6	5	6	5	4	3
o	5	4	5	4	3	4
r	4	3	4	3	4	5
r	3	2	3	2	3	4
i	2	1	2	3	4	5
s	1	0	1	2	3	4
#	0	1	2	3	4	5
	#	s	e	r	o	w

جواب: با توجه به این الگوریتم تطبیق بهینه برابر با کلمه SORROW میباشد که فاصله ویرایش کمینه آن برابر با ۲ است.

Problem 2:

Seq 1: CTCAAACCAT

Seq 2: CAAGACC

		C	A	A	G	A	C	C
	0	-2	-4	-6	-8	-10	-12	-14
C	-2	1	-1	-3	-5	-7	-9	-11
T	-4	-1	0	-2	-4	-6	-8	-10
C	-6	-3	-2	-1	-3	-5	-5	-7
A	-8	-5	-2	-1	-2	-2	-4	-6
A	-10	-7	-4	-1	-2	-1	-3	-5
A	-12	-9	-6	-3	-2	-1	-2	-4
C	-14	-11	-8	-5	-4	-3	0	-1
C	-16	-13	-10	-7	-6	-5	-2	1
A	-18	-15	-12	-9	-8	-5	-4	-1
T	-20	-17	-14	-11	-10	-7	-6	-3

CA – A G A C C – –
CT C A A A C C A T

Score: -3

Generative and Discriminative Model:

Problem 1:

در ابتدا باید اشاره کرد که هر دو مدل برای مسائل یادگیری با نظارت استفاده میشوند. هر دو مدل باید تابع f را تخمین بزنند که به صورت زیر تعریف میشود:

$f: X \rightarrow Y$

Labels: $Y = y$

Features: $X = \{x_1, x_2, \dots, x_n\}$

همچنین تخمین این تابع را به صورت پیش بینی احتمال شرطی $P(Y|X)$ مینویسند.

این دو برای تخمین احتمال شرطی از دو رویکرد متفاوت استفاده میکنند.

Discriminative model

این مدل مرز تصمیمی که کلاس ها را از هم جدا میکند یاد میگیرد. در واقع $P(Y|X)$ به فرم یک تابع فرض میشود و پارامتر های این تابع به صورت مستقیم از دادگان آموزشی یاد گرفته میشوند. به عنوان مثال میتوان به الگوریتم logistic regression یکی از مدل های Discriminative است.

برای درک بهتر میتوان الگوریتم logistic regression با یک مثال توضیح داد.

فرض کنیم میخواهیم پیش بینی کنیم که یک ایمیل اسپم هست (1) یا خیر (0). پس ما مسئله را به این صورت مدل خواهیم کرد:

Labels = 0 or 1

Hypothesis: $Z = WX + B$

$\hat{Y} = P(Y|X; W) = \sigma(Z)$

در این مسئله پارامترهای W و B باید یادگرفته شوند و تابعی که ما برای احتمال شرطی فرض کرده ایم یک معادله خطی است که روی آن تابع فعالسازی سگموئید اعمال شده است.

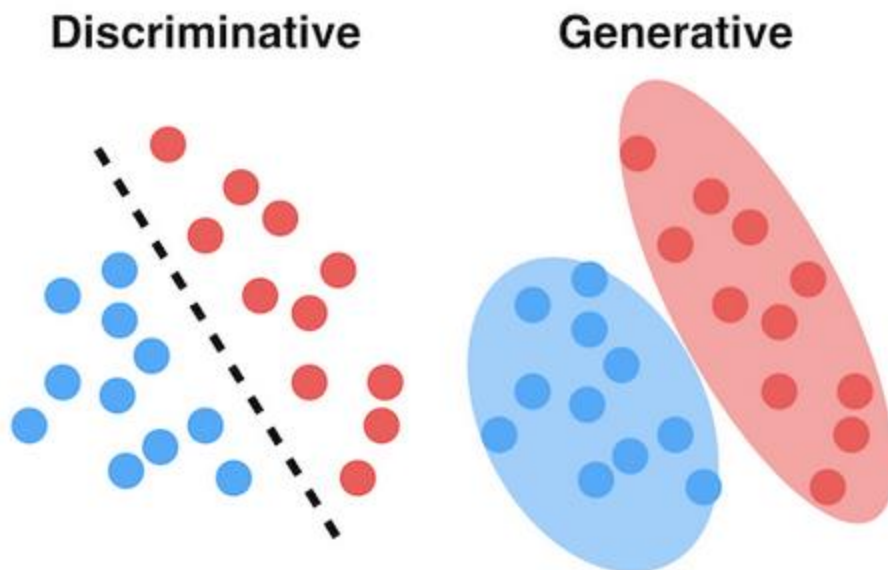
Generative model

در این مدل توزیع واقعی هر کلاس صریحاً مدل خواهد شد. این مدل توزیع احتمال توأم یا $P(X, Y)$ را یاد میگیرد. برای پیش بینی $P(Y|X)$ از تئوری بیز استفاده میکنند که محاسبه آن به صورت فرمول زیر میباشد:

$$posterior = \frac{prior \times likelihood}{evidence} \Rightarrow P(Y|X) = \frac{P(Y) \cdot P(X|Y)}{P(X)}$$

به عنوان مثال naïve bayse یکی از الگوریتم های مدل Generative است.

در شکل زیر به صورت بصری میتوان تفاوت بین این مدل را مشاهده کرد:



Problem 2:

در naïve bayes مشاهده هایی که همبستگی بالایی دارند مدل میشوند. با توجه به اینکه احتمال posterior از فرمول زیر محاسبه میشود:

$$c_{NB} = \operatorname{argmax}_{c_j \in C} P(c_j) \prod_{i \in \text{positions}} P(x_i | c_j)$$

با توجه به این فرمول احتمال هر ویژگی به شرط کلاس در هم ضرب میشوند، حتی ویژگی های چند کلمه ای هستند و یک چیز را بیان میکنند. برای مثال کلمه Hong Kong دارای چند کلمه ای است و احتمال $P(\text{Kong}|c)$ و $P(\text{Hong}|c)$ در هم ضرب میشوند.

در حالیکه در maxent classifier ما با ساختن feature ها و وزن دار کردن آن ها از بروز چنین مشکلی جلوگیری میکنیم. بنابراین خروجی مدل با انتظار ما در واقعیت، به یک دیگر نزدیک میشوند. فرمول این دسته بند به صورت زیر است:

$$\text{vote}(c) = \sum \lambda_i f_i(c, d)$$

که f ویژگی ها و λ وزن ها هستند.

Problem 3:

برای حل این مسئله از یک Generative model استفاده میکنیم. معماری این مدل به شکل زیر است:

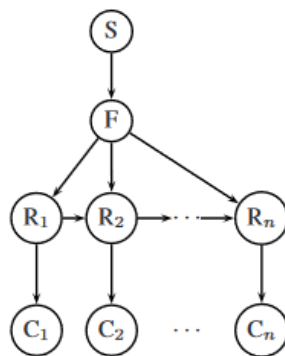


Fig. 1. Role Tagger

هدف این معماری پیدا کردن نقش ها و فریم ها است و ورودی جملات زبان طبیعی و مسند ها هستند. مشکلی مدل های Discriminative دارند این است که نمیتوانند نمونه های null را تشخیص دهند. در این مدل باید رابطه بین فریم و نقش تشخیص داده شود و بفهمد که هر جز اصلی از جمله احتمالا برای کدام مسند نقش دارد. برای حل این مسئله از مدل مخفی مارکوف استفاده میشود که یک توزیع احتمالی توام را بر روی مسند ها، فریم ها، نقش ها و جز ها تعریف میکند. طبق شکل ابتدا مسند (S) انتخاب میشود که فریم (F) را تولید میکند. سپس فریم دنباله نقش را تولید میکند که از R1 تا Rn است. در ادامه جز های اصلی جمله از C1 تا Cn تولید میشوند. رابطه توزیع احتمال توام به صورت زیر است:

$$P(\mathbf{C}, \mathbf{R}, F, S) = P(S) \times P(F|S) \times P(\mathbf{R}|F, S) \times P(\mathbf{C}|\mathbf{R}, F, S),$$

برای ساده سازی از فرض مارکوف استفاده میشود و هر نقش فقط به نقش قبلی آن وابسته میشود:

$$P(\mathbf{R}|F, S) = \prod_i P(R_i|R_1 \cdots R_{i-1}, F) \approx \prod_i P(R_i|R_{i-1}, F)$$

فرض دیگر این مدل این است که جز اصلی جمله از فریم ها و مسند های داده شده مستقل است که فرمول آن به صورت زیر است:

$$P(\mathbf{C}|F, S, \mathbf{R}) \approx P(\mathbf{C}|\mathbf{R}) = \prod_i P(C_i|R_i),$$

برای آموزش مدل بیشتر پارامتر های مدل با استفاده از تخمین ساده maximum likelihood تخمین زده میشوند که مبتنی بر داده های آموزشی کامل برچسب خورده میباشد.

مقاله مطالعه شده برای روش پیشنهادی:

[A Generative Model for Semantic Role Labeling](#)

Problem 4:

Tabel 3) Generative algoritim tabel

$P(a, b)$	$b = 1$	$b = 2$	$b = 3$
$a = 0$	$1/10$	$1/10$	$2/10$
$a = 1$	$2/10$	0	$1/10$
$a = 2$	$1/10$	$1/10$	$1/10$

Tabel 3) Discriminative algoritim tabel

$P(a b)$	$b = 1$	$b = 2$	$b = 3$
$a = 0$	$1/4$	$1/2$	$2/4$
$a = 1$	$2/4$	0	$1/4$
$a = 2$	$1/4$	$1/2$	$1/4$

Text Classification:

Problem 1:

Naïve bayes یک در الگوریتم ساده و قدرتمند برای مدل های پیش بینی است. این الگوریتم naïve یا ساده لوح نامیده میشود به این دلیل که فرض میکند هر متغیر ورودی یا همان ویژگی ها از یک دیگر مستقل هستند. این فرض، یک فرض قوی و غیرواقعی برای داده های واقعی میباشد اما در عین حال برای بازه بزرگی از مسائل پیچیده تکنیک موثری است.

Problem 2:

New sample in train set:

Today I read a amazing book. Label: Happy

New sample in test set:

This book is very fun. Label: Happy

$P(c \mid \text{I feel relaxed and lucky after my job interview}) = ?$

$P(\text{sadness} \mid \text{I feel relaxed and lucky after my job interview}) \propto P(I \mid \text{sadness}) *$

$P(\text{feel} \mid \text{sadness}) *$

$P(\text{relaxed} \mid \text{sadness}) *$

$P(\text{and} \mid \text{sadness}) *$

$P(\text{lucky} \mid \text{sadness}) *$

$P(\text{after} \mid \text{sadness}) *$

$P(\text{my} \mid \text{sadness}) *$

$P(\text{job} \mid \text{sadness}) *$

$P(\text{interview} \mid \text{sadness}) *$

$P(\text{sadness})$

برای محاسبه هر یک از این likelihood ها از فرمول زیر استفاده میکنیم:

$$\hat{P}(w \mid c) = \frac{\text{count}(w, c) + 1}{\text{count}(c) + |V|}$$

$$|V| = 30$$

$$P(I \mid \text{sadness}) = (2+1) / (18+30) = 3/48$$

$$P(\text{feel} \mid \text{sadness}) = (0+1) / (18+30) = 1/48$$

$$P(\text{relaxed} \mid \text{sadness}) = (0+1) / (18+30) = 1/48$$

$$P(\text{and} \mid \text{sadness}) = (1+1) / (18+30) = 2/48$$

$$P(\text{lucky} \mid \text{sadness}) = (0+1) / (18+30) = 1/48$$

$$P(\text{after} \mid \text{sadness}) = (0+1) / (18+30) = 1/48$$

$$P(\text{my} \mid \text{sadness}) = (1+1) / (18+30) = 2/48$$

$$P(\text{job} \mid \text{sadness}) = (1+1) / (18+30) = 2/48$$

$$P(\text{interview} \mid \text{sadness}) = (0+1) / (18+30) = 1/48$$

$$P(\text{sadness}) = 3/7$$

$$P(\text{sadness} \mid I \text{ feel relaxed and lucky after my job interview})$$

$$\propto \left(\frac{3}{7} * \frac{3}{48} * \frac{1}{48} * \frac{1}{48} * \frac{2}{48} * \frac{1}{48} * \frac{1}{48} * \frac{2}{48} * \frac{2}{48} * \frac{1}{48} \right) = 7.6043714e - 15$$

$$P(\text{Happy} \mid I \text{ feel relaxed and lucky after my job interview}) \propto P(I \mid \text{Happy}) *$$

$$P(\text{feel} \mid \text{Happy}) *$$

$$P(\text{relaxed} \mid \text{Happy}) *$$

$$P(\text{and} \mid \text{Happy}) *$$

$$P(\text{lucky} \mid \text{Happy}) *$$

$$P(\text{after} \mid \text{Happy}) *$$

$$P(\text{my} \mid \text{Happy}) *$$

$$P(\text{job} \mid \text{Happy}) *$$

$$P(\text{interview} \mid \text{Happy}) *$$

$$P(\text{Happy})$$

$$P(I \mid \text{Happy}) = (2+1) / (23+30) = 3/53$$

$$P(\text{feel} \mid \text{Happy}) = (1+1) / (23+30) = 2/53$$

$$P(\text{relaxed} \mid \text{Happy}) = (1+1) / (23+30) = 2/53$$

$$P(\text{and} \mid \text{Happy}) = (1+1) / (23+30) = 2/53$$

$$P(\text{lucky} \mid \text{Happy}) = (1+1) / (23+30) = 2/53$$

$$P(\text{after} \mid \text{Happy}) = (0+1) / (23+30) = 1/53$$

$$P(\text{my} \mid \text{Happy}) = (1+1) / (23+30) = 2/53$$

$$P(\text{job} \mid \text{Happy}) = (0+1) / (23+30) = 1/53$$

$$P(\text{interview} \mid \text{Happy}) = (0+1) / (23+30) = 1/53$$

$$P(\text{Happy}) = 4/7$$

$$P(\text{Happy} \mid I \text{ feel relaxed and lucky after my job interview}) \\ \propto \left(\frac{4}{7} * \frac{3}{53} * \frac{2}{53} * \frac{2}{53} * \frac{2}{53} * \frac{2}{53} * \frac{1}{53} * \frac{2}{53} * \frac{1}{53} * \frac{1}{53} \right) = 1.6624568e - 14$$

Predicted Class: Happy

P(c | This movie was boring and predictable with no fun) = ?

P(sadness | This movie was boring and predictable with no fun) \propto P(This | sadness) *

P(movie | sadness) *

P(was | sadness) *

P(boring | sadness) *

P(and | sadness) *

P(predictable | sadness) *

P(with | sadness) *

P(no | sadness) *

P(fun | sadness) *

P(sadness)

$$P(\text{This} \mid \text{sadness}) = (0+1) / (18+30) = 1/48$$

$$P(\text{movie} \mid \text{sadness}) = (0+1) / (18+30) = 1/48$$

$$P(\text{was} \mid \text{sadness}) = (1+1) / (18+30) = 2/48$$

$$P(\text{boring} \mid \text{sadness}) = (1+1) / (18+30) = 2/48$$

$$P(\text{and} \mid \text{sadness}) = (1+1) / (18+30) = 2/48$$

$$P(\text{predictable} \mid \text{sadness}) = (0+1) / (18+30) = 1/48$$

$$P(\text{with} \mid \text{sadness}) = (0+1) / (18+30) = 1/48$$

$$P(\text{no} \mid \text{sadness}) = (0+1) / (18+30) = 1/48$$

$$P(\text{fun} \mid \text{sadness}) = (0+1) / (18+30) = 1/48$$

$$P(\text{sadness}) = 3/7$$

$$P(\text{sadness} \mid \text{This movie was boring and predictable with no fun})$$

$$\propto \left(\frac{3}{7} * \frac{1}{48} * \frac{1}{48} * \frac{2}{48} * \frac{2}{48} * \frac{2}{48} * \frac{1}{48} * \frac{1}{48} * \frac{1}{48} * \frac{1}{48} \right) = 2.5347905e - 15$$

$$P(\text{Happy} \mid \text{This movie was boring and predictable with no fun}) \propto P(\text{This} \mid \text{Happy}) *$$

$$P(\text{movie} \mid \text{Happy}) *$$

$$P(\text{was} \mid \text{Happy}) *$$

$$P(\text{boring} \mid \text{Happy}) *$$

$$P(\text{and} \mid \text{Happy}) *$$

$$P(\text{predictable} \mid \text{Happy}) *$$

$$P(\text{with} \mid \text{Happy}) *$$

$$P(\text{no} \mid \text{Happy}) *$$

$$P(\text{fun} \mid \text{Happy}) *$$

$$P(\text{Happy})$$

$$P(\text{This} \mid \text{Happy}) = (0+1) / (23+30) = 1/53$$

$$P(\text{movie} \mid \text{Happy}) = (1+1) / (23+30) = 2/53$$

$$P(\text{was} \mid \text{Happy}) = (0+1) / (23+30) = 1/53$$

$$P(\text{boring} \mid \text{Happy}) = (0+1) / (23+30) = 1/53$$

$$P(\text{and} \mid \text{Happy}) = (1+1) / (23+30) = 2/53$$

$$P(\text{predictable} \mid \text{Happy}) = (0+1) / (23+30) = 1/53$$

$$P(\text{with} \mid \text{Happy}) = (0+1) / (23+30) = 1/53$$

$$P(\text{no} \mid \text{Happy}) = (0+1) / (23+30) = 1/53$$

$$P(\text{fun} \mid \text{Happy}) = (1+1) / (23+30) = 2/53$$

$$P(\text{Happy}) = 4/7$$

$$P(\text{Happy} \mid \text{This movie was boring and predictable with no fun})$$

$$\propto \left(\frac{4}{7} * \frac{1}{53} * \frac{2}{53} * \frac{1}{53} * \frac{1}{53} * \frac{2}{53} * \frac{1}{53} * \frac{1}{53} * \frac{1}{53} * \frac{2}{53} \right) = 1.3853806e - 15$$

Predicted Class: sadness

$$P(c \mid \text{This book is very fun}) = ?$$

$$P(\text{sadness} \mid \text{This book is very fun}) \propto P(\text{This} \mid \text{sadness}) *$$

$$P(\text{book} \mid \text{sadness}) *$$

$$P(\text{is} \mid \text{sadness}) *$$

$$P(\text{very} \mid \text{sadness}) *$$

$$P(\text{fun} \mid \text{sadness}) *$$

$$P(\text{sadness})$$

$$P(\text{This} \mid \text{sadness}) = (0+1) / (18+30) = 1/48$$

$$P(\text{book} \mid \text{sadness}) = (1+1) / (18+30) = 2/48$$

$$P(\text{is} \mid \text{sadness}) = (0+1) / (18+30) = 1/48$$

$$P(\text{very} \mid \text{sadness}) = (0+1) / (18+30) = 1/48$$

$$P(\text{fun} \mid \text{sadness}) = (0+1) / (18+30) = 1/48$$

$$P(\text{sadness}) = 3/7$$

$$P(\text{sadness} \mid \text{This book is very fun}) \propto \left(\frac{3}{7} * \frac{1}{48} * \frac{2}{48} * \frac{1}{48} * \frac{1}{48} * \frac{1}{48} \right) = 3.36393057e - 9$$

$$\begin{aligned}
 P(\text{Happy} \mid \text{This book is very fun}) &\propto P(\text{This} \mid \text{Happy}) * \\
 &\quad P(\text{book} \mid \text{Happy}) * \\
 &\quad P(\text{is} \mid \text{Happy}) * \\
 &\quad P(\text{very} \mid \text{Happy}) * \\
 &\quad P(\text{fun} \mid \text{Happy}) * \\
 &\quad P(\text{Happy})
 \end{aligned}$$

$$P(\text{This} \mid \text{Happy}) = (0+1) / (23+30) = 1/53$$

$$P(\text{book} \mid \text{Happy}) = (0+1) / (23+30) = 1/53$$

$$P(\text{is} \mid \text{Happy}) = (1+1) / (23+30) = 2/53$$

$$P(\text{very} \mid \text{Happy}) = (1+1) / (23+30) = 2/53$$

$$P(\text{fun} \mid \text{Happy}) = (1+1) / (23+30) = 2/53$$

$$P(\text{Happy}) = 4/7$$

$$P(\text{Happy} \mid \text{This book is very fun}) \propto \left(\frac{4}{7} * \frac{1}{53} * \frac{1}{53} * \frac{2}{53} * \frac{2}{53} * \frac{2}{53} \right) = 1.09313196e - 8$$

Predicted Class: Happy

Warm-Up for Deep Learning:

1- Softmax activation یک تابع فعالسازی مناسب برای مسائل دسته بندی چند کلاسه است. فرمول آن به صورت زیر میباشد:

$$\sigma(\vec{z})_i = \frac{e^{z_i}}{\sum_{j=1}^K e^{z_j}}$$

این تابع به چند دلیل مناسب است:

- a. خروجی شبکه بعد از اعمال این تابع حتما مثبت خواهد بود و برای کلاس ها امتیاز منفی نخواهیم داشت. این کار با استفاده از تابع نمایی انجام میشود که یکنوا است و ترتیب اعداد را به لحاظ بزرگی حفظ میکند.
- b. امتیاز همه کلاس به شکل احتمال در می آید که بین ۰ تا ۱ هستند و جمع آن ها برابر با ۱ خواهد شد. به این صورت خروجی کلاس ها با یک دیگر قابل مقایسه خواهد بود.

2- معماری cuda در یادگیری عمیق انجام محاسبات در شبکه هایی مانند شبکه های کانولوشنال و ترنسفورمر ها را موازی سازی میکند. به این معنی که یک task در این شبکه ها به تعداد زیادی subtask تبدیل میشود و از یک واحد پردازش به اسم GPU برای انجام این subtask ها به صورت همزمان بهره میبرند. این موازی سازی زمان محاسبات را در شبکه های عصبی عمیق بسیار کاهش میدهد.

3- ترنسفورمر یک مدل یادگیری عمیق است که از مکانیسم توجه به خود (self-attention) استفاده میکند و اهمیت هر بخش از داده های ورودی را به طور متفاوت وزن می کند. این ایده الهام گرفته از مغز انسان است. در مغز انسان توجه یا attention یک فرایند شناختی است که به صورت انتخابی روی یک یا تعداد کمی مورد تمرکز میکند و بقیه موارد را نادیده میگیرد. برای مثال اگر یک عکس از عکس های دوران مدرسه را به شما نشان دهند که تعدادی دانش آموز به همراه معلمشان ایستاده باشند. سپس این سوال را مطرح کنند که چه تعداد آدم در این عکس وجود دارد؟ مغز شما برای شمارش فقط به تعداد سر ها توجه میکند و بقیه موارد در تصویر را دور میریزد. همچنین برای ترجمه یک جمله از زبانی به زبان دیگر، هنگام ترجمه یک کلمه به زبان مقصد، به کلمات خاصی از زبان مبدا توجه میکنیم.

4- Symbolic AI: در این رویکرد فرض میشود که میتوان دنیا با یک بازنمایی ساختار یافته نشان داد. در این پارادایم نماد ها بیانگر اشیا و چیزهای دنیا هستند. همچنین ما برای استنتاج از یک پایگاه دانش استفاده میکنیم که شامل مجموعه ای از قوانین است.

Connectionism: در این پارادایم تصمیم گیری های هوشمندانه از طریق یه سیستم گرفته میشود که سیستم متشکل از نود ها پردازشی مانند نورون ها است که به یک دیگر متصل هستند و اتصال این نورون ها وزن دار است. معروفترین تکنیک در این دسته شبکه های عصبی مصنوعی هستند که از تعداد لایه با نود تشکیل شده اند که به آن ها نورون گفته میشود. این نورون ها سیگنال های ورودی را پردازش و با یک دیگر ترکیب میکنند. جنبه کلیدی این پارادایم این است که کاربر نیازی به مشخص کردن قوانین

دامنه ای که باید مدل شود ندارد. این سیستم فقط به تعداد کافی از داده های نمونه نیاز دارد تا بتواند به صورت آماری دنیا را مدل کند.

معایب symbolic AI: این پارادایم ساده است و مسائل خاصی و محدودی که برای آن ها طراحی شده است را به خوبی حل میکند. اما عیب اولیه ای که هوش مصنوعی نمادین این است که نمیتواند به خوبی تعمیم دهی را انجام دهد. در اینجا ما با یک مجموعه از نماد ها و قوانین رو به رو هستیم که برای یک وظیفه خاص ساخته شده است و نمیتواند برای وظیفه های دیگر قابل تعمیم باشد. همچنین این پارادایم شکننده است زیرا اگر یک فرض یا قانون برقرار نباشد ممکن سیستم استنتاج به شکست باینگامد. مسئله بعدی این است که آیا سیستم هوش مصنوعی نمادین یک یادگیرنده واقعی است یا اینکه با استفاده از یک سری قوانین سطحی تصمیم گیری میکند؟ با توجه به توضیحاتی برای connectionist AI توضیح داده شد و همینطور معایبی که برای Symbolic AI گفته شد، استفاده از الگوریتم های شبکه عصبی و یادگیری عمیق در حال حاضر فراگیر شده است. البته با توجه به اینکه هر سیستم معایب و مزایای خود را دارد استفاده ترکیبی از آن ها میتواند از مزایا هر دو بهره برد که به آن hybrid system میگویند.

5- مفاهیم:

Tensor: یه آرایه چند بعدی است. این آرایه تعمیمی از بردار و ماتریس است.

Embedding: یک فضای نسبتا با بعد کوچک است که میتوان بردار های با بعد بزرگ را در آن ترجمه کرد. با استفاده از embedding ها، کار یادگیری ماشین روی ورودی های بزرگی مثل بردار های خلوتی که کلمات را بازنمایی میکنند آسان تر میشود. embedding مفاهیم معنایی ورودی را جمع آوری میکند و در فضای embedding، ورودی هایی که شبیه به یک دیگر هستند را نزدیک هم قرار میدهد. Embedding میتواند یاد گرفته شود و در مدل ها مجددا استفاده میشود.

Representation: به صورت ابتدایی، یادگیری بازنمایی یک مجموعه از ویژگی ها است که مفاهیم را توصیف میکنند. مثلا ما میتوانیم یک شی با استفاده از رنگ ها، اشکال، اندازه و بازنمایی کنیم. با استفاده از بازنمایی میتوان به مفاهیمی که متفاوت هستند تفاوت قائل شد یا حتی شباهت بین آن ها را پیدا کرد. یک مثال خوب بازنمایی کلمات است. word2Vec یکی از این بازنمایی ها است و کمک میکند م به چنین چیزی برسیم:

ملکه = زن + مرد - پادشاه

به این معنا که این بازنمایی یاد گرفته است که تفاوت بین ملکه و پادشاه، تفاوت بین زن و مرد است. در شبکه های عصبی ای بازنمایی ها در لایه های مخفی هم انجام میشود. خروجی هم لایه مخفی یک نگاشت از ورودی اصلی به یک انتزاع سطح بالا است که ویژگی های مفیدی را یاد گرفته است.

Optimizer: توابع ضرر یک معیار ریاضی است که به ما میگوید که شبکه های عصبی به چه اندازه در پیش بینی های خود اشتباه داشته اند. در فرایند آموزش ما پارامتر ها یا همان وزن های شبکه طوری تغییر میدهیم که مقدار تابع ضرر کمینه شود و پیش های شبکه تا حد ممکن صحیح باشد. اینکه این وزن ها با چه سیاستی عوض شود را بهینه ساز تعیین میکند. از جمله بهینه ساز های معروف **Gradient Descent**، **Rmsprop** و **Adam** هستند.

Scheduler: برای تنظیم نرخ یادگیری در طول آموزش استفاده میشود و با توجه به یک برنامه ریز از پیش تعریف شده نرخ یادگیری را به مرور کاهش میدهد. برنامه ریز های رایج عبارتند از:

Time-based decay

Step decay

Exponential decay