沪江

自适应学习
机器学习在开心词场中的应用

王新义@HJ

日程

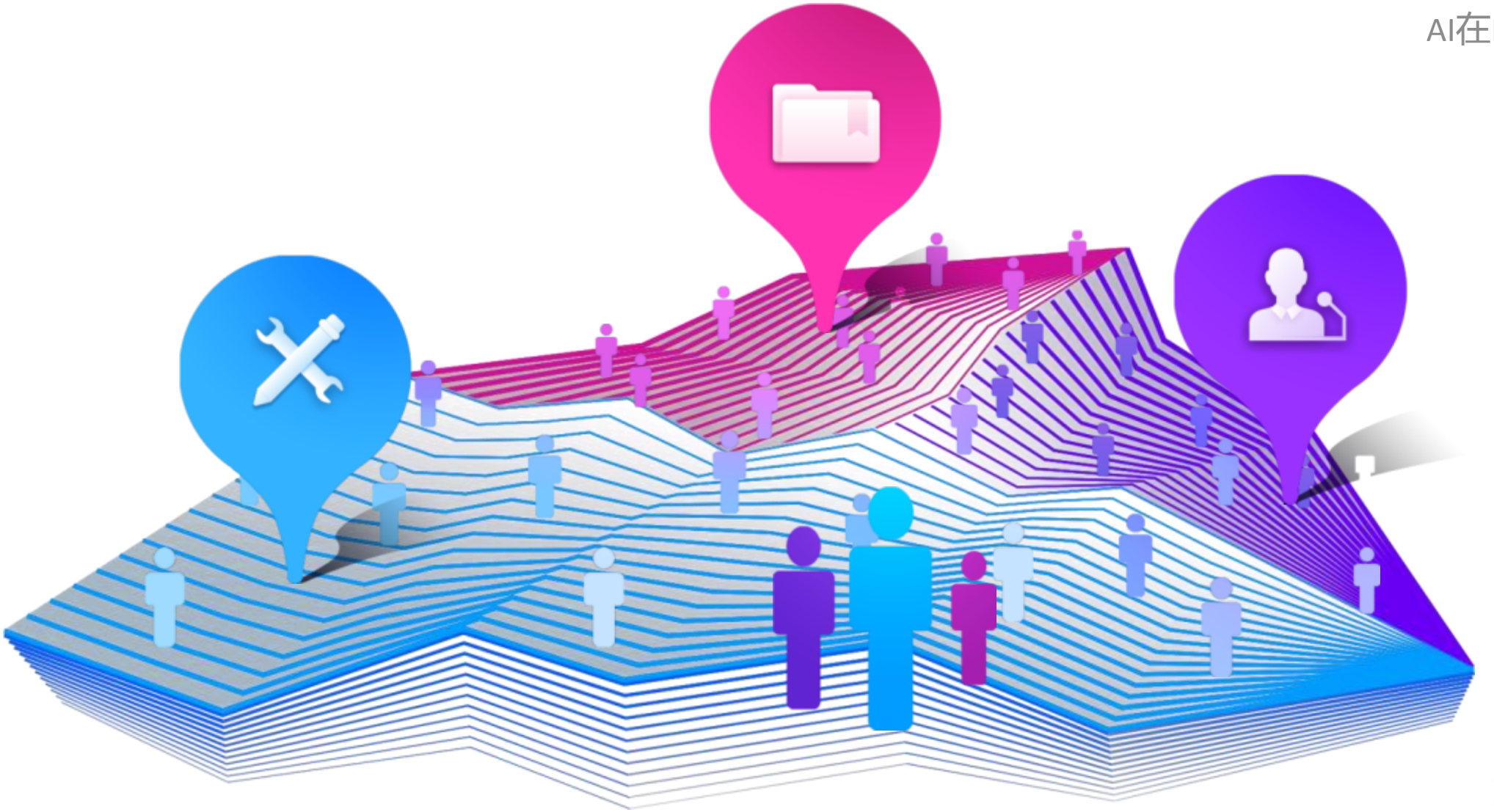
- 关于教育及互联网教育
- 机器学习在沪江的应用
- 机器学习在开心词场中应用
  - 自适应词汇量测试
  - 记忆模型
  - 词性标注

关于教育及互联网教育

- 教育是最传统、最复杂、涉及面广的社会活动
- 教育痛点：公平、效率、痛苦
- 互联网教育：低频、高交互
- 使命：使用机器学习(AI)技术改造和促进人类自身学习(提高学习效率和学习效果)

真正的互联网教育是具备大规模、复杂交互行为的普惠性知识学习和教育。——阿诺

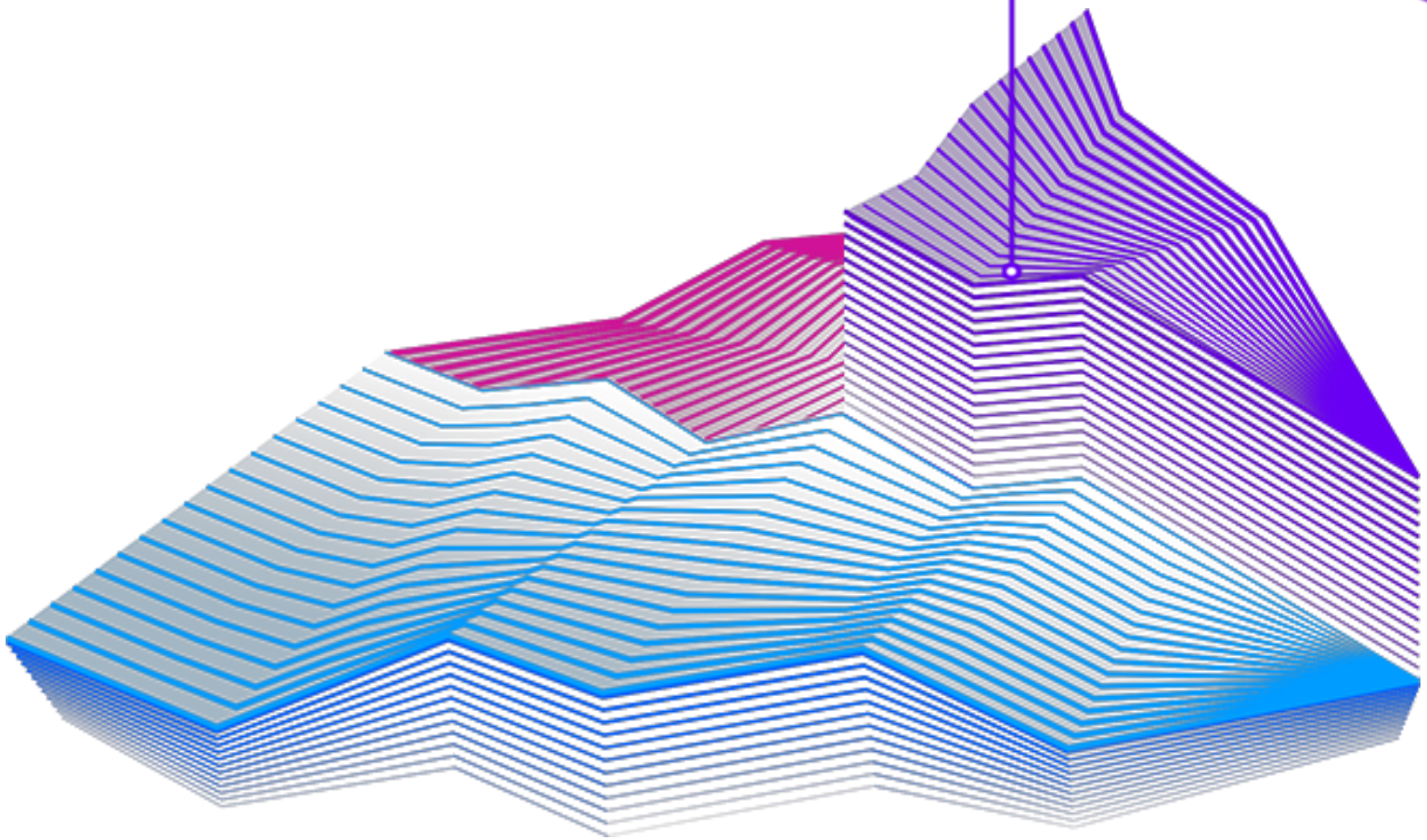
机器学习在沪江的应用



AI在HJ的四个应用场景：

- 自适应学习
- 人机交互
- 教学过程监控
- 内容加工

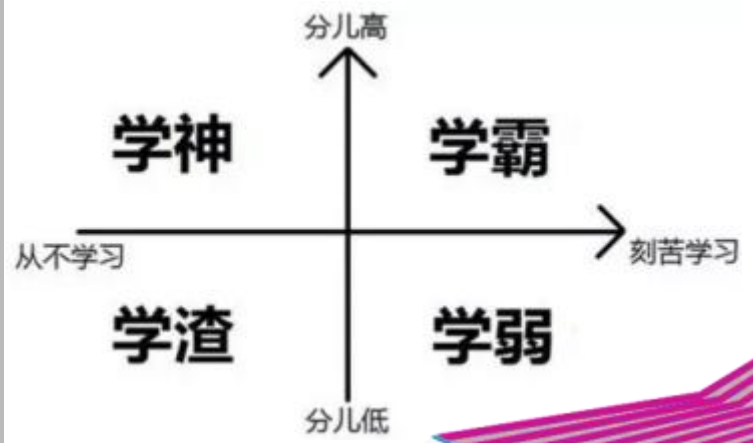
机器学习在沪江的应用



• 网 师

- 洞察学生
- 洞察自己
- 洞察市场

机器学习在沪江的应用



• 学习者

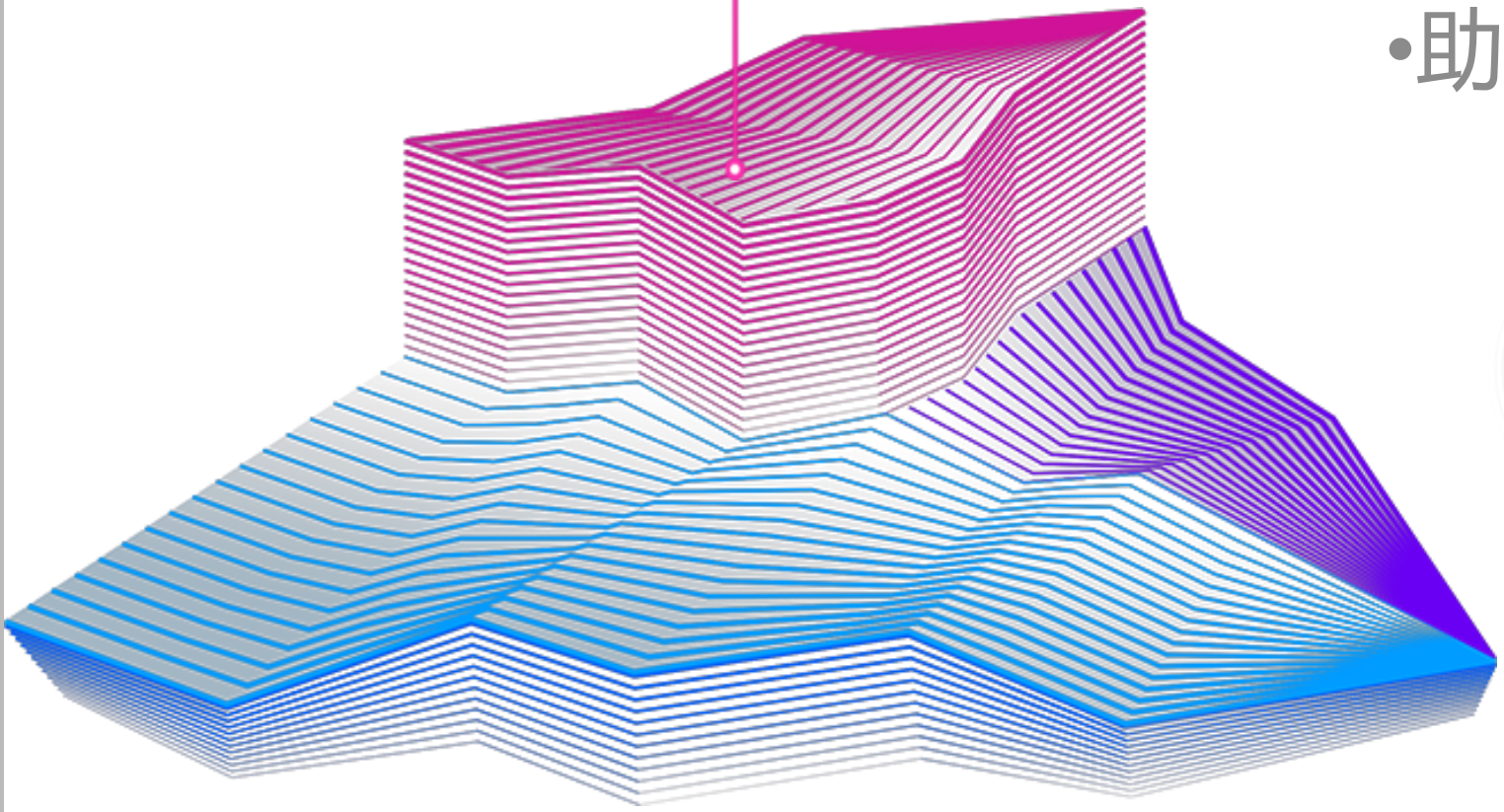
- 自适应学习：智能导学服务、智能学习助手
- 用户模型：学习需求、学习意愿、学习能力、经济能力、学习毅力等
- 丰富多样的学习体验

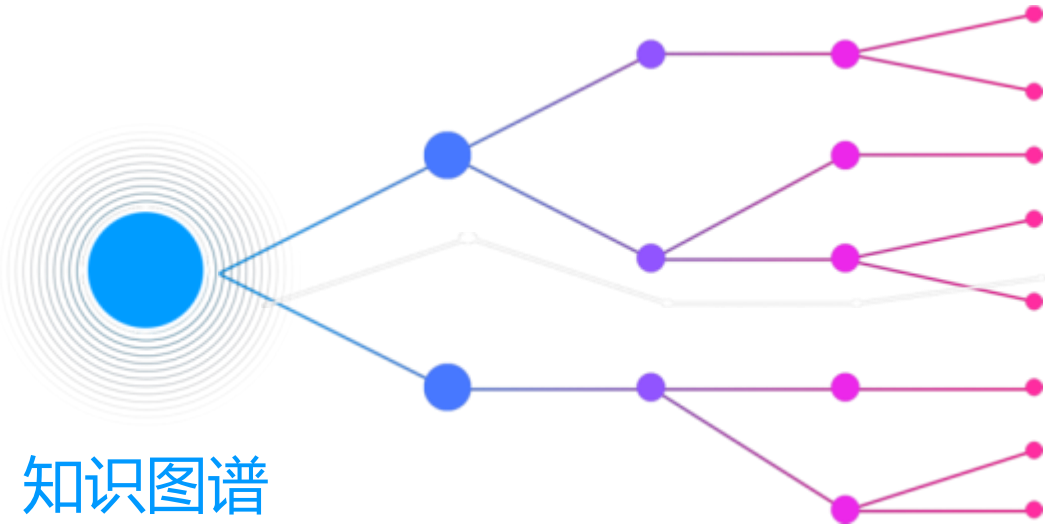
机器学习在沪江的应用

• 内 容

•加速内容产品化

•助推内容商品化



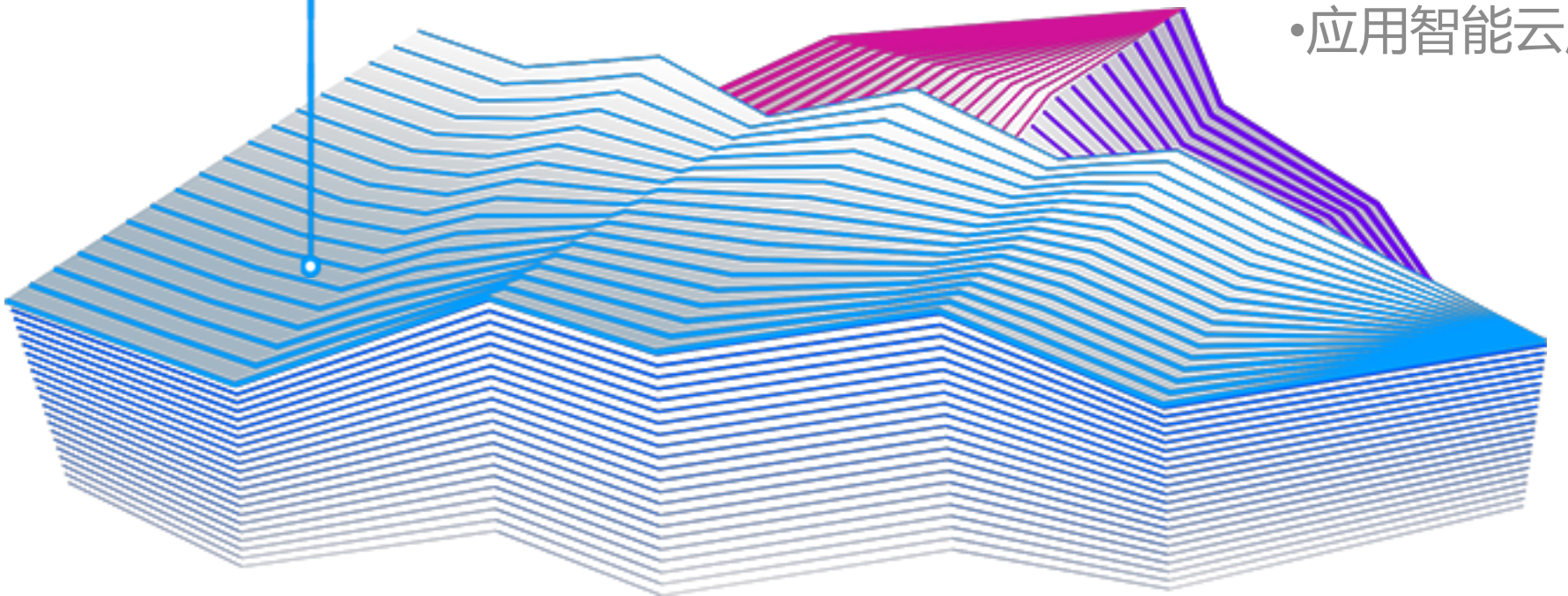


知识图谱

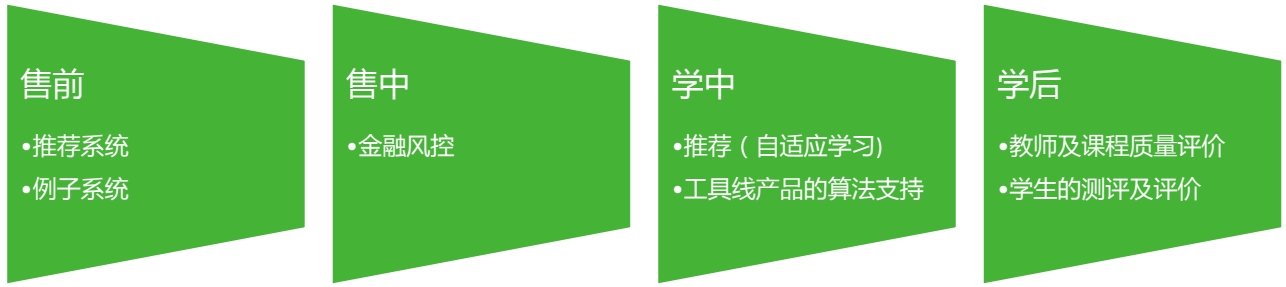
机器学习在沪江的应用

•工 具

- 通过开放接口与教学场景深度融合
- 用户行为感知和数据分析
- 应用智能云服务



机器学习在沪江的应用



基础能力层

用户画像　推荐引擎　自适应学习(导学）　内容打标　DMP

行为及特征模型

学习信用分　质量评价　转化预测模型　知识图谱　时间序列预测模型

基础模型算法

集成学习　深度学习框架　文本、图片、语音基础算法

Nature Language Process:NLP　Point Processes　Deep Neural Network

机器学习在开心词场中应用

学习，成为更好的自己

机器学习在开心词场中应用

- 自适应词汇量测试
- 记忆模型
- 词性标注

自适应词汇量测试

自适应词汇量测试

☐ skirt	☐ sword	☐ blink	☐ natter
☐ chop	☐ bounce	☐ bait	☐ deflect
☐ stall	☐ stance	☐ juggle	☐ flurry
☐ hunt	☐ bat	☐ tinker	☐ whim
☐ storm	☐ mercy	☐ drought	☐ shaggy
☐ rope	☐ cellar	☐ dangle	☐ reproach
☐ stake	☐ bald	☐ outright	☐ trundle
☐ constraint	☐ portray	☐ beware	☐ rampant
☐ hint	☐ grim	☐ wriggle	☐ judicious
☐ lawn	☐ tan	☐ glide	☐ whiff
☐ hedge	☐ refurbish	☐ ledge	☐ tandem
☐ nest	☐ saddle	☐ loot	☐ scathing
☐ truck	☐ scent	☐ feeble	☐ rind
☐ drift	☐ wallet	☐ seep	☐ strife
☐ harvest	☐ meadow	☐ greed	☐ grouse
☐ lick	☐ clap	☐ shrine	☐ bawl
☐ knit	☐ tickle	☐ stoop	☐ spool
☐ beam	☐ hollow	☐ numb	☐ pittance
☐ dull	☐ forbid	☐ bustle	☐ shrivel
☐ withdraw	☐ drip	☐ burp	☐ easel
☐ shave	☐ warranty	☐ swivel	☐ mayhem
☐ crush	☐ strive	☐ drab	☐ rascal
☐ summit	☐ cute	☐ throttle	☐ fledgling
☐ carve	☐ alley	☐ streamline	☐ signet
☐ lap	☐ junk	☐ napkin	☐ dowdy
☐ harsh	☐ hamper	☐ resilient	☐ lore
☐ clutch	☐ gleam	☐ topple	
☐ scrub	☐ plank	☐ blunder	
☐ toss	☐ grumble	☐ stead	

http://testyourvocab.com/
静态勾选
需要做很多题目

个性化，交互式
~40 道题目

- 减少测试时间
- 提升用户体验

帮助学生推荐合适的
- 词书
- 句子
- 文章

基本原理

静态考卷

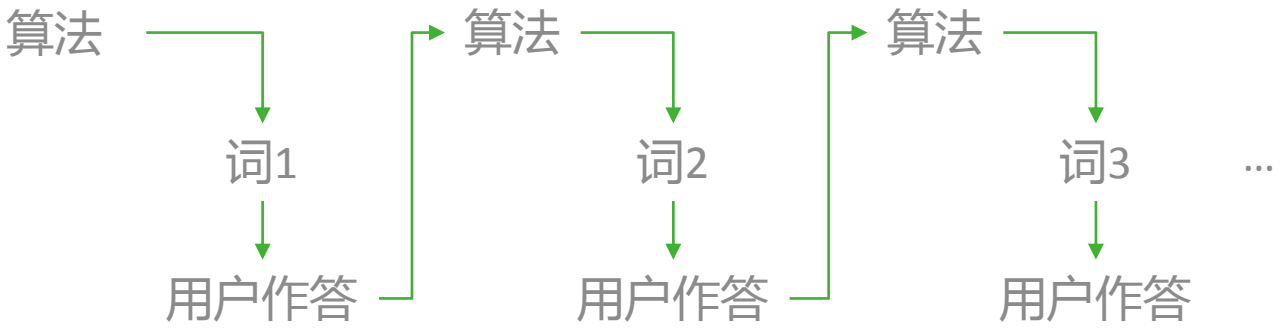
- 每个学生所做的题目相同
- 在以下题目上浪费较多时间，影响用户体验
  - 肯定会做的容易题
  - 肯定不会做的难题

动态交互式测试

- 每个学生所做的题目不相同
- 下一道题目根据历史做题的反馈动态改变
- 算法可以聚焦于算法不确定的题目请学生回答，而避免在肯定会做和肯定不会做的题目浪费太多的时间

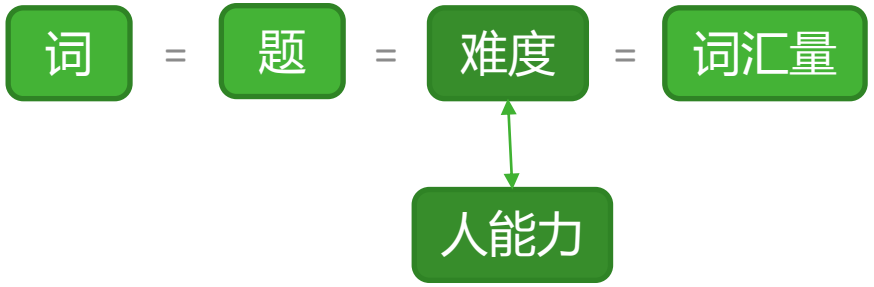
基本假设：
- 单词和单词之间有相对顺序
- 难度接近的单词，用户的作答也接近

流程



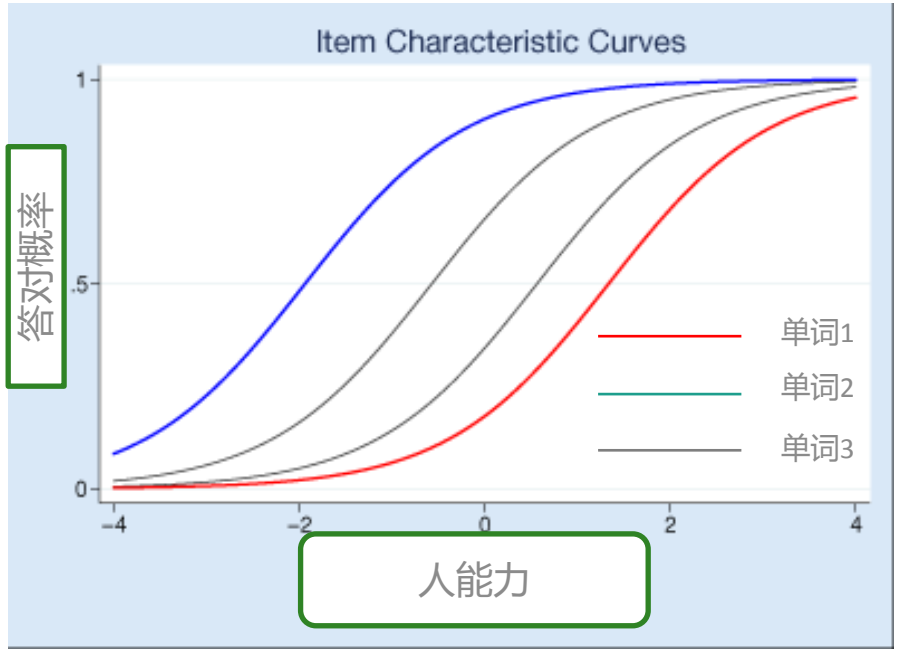
算法:

1. 根据做题记录，估计人能力
2. 出下一道题



- 词，题，难度，词汇量是对应的
- 难度已知:
  - 冷启动的时候，由词频保序变换而来
  - 有了用户作答数据，可以通过计算而得到
- 人能力和词难度可比。有了人能力，通过对应的难度，可以算得词汇量

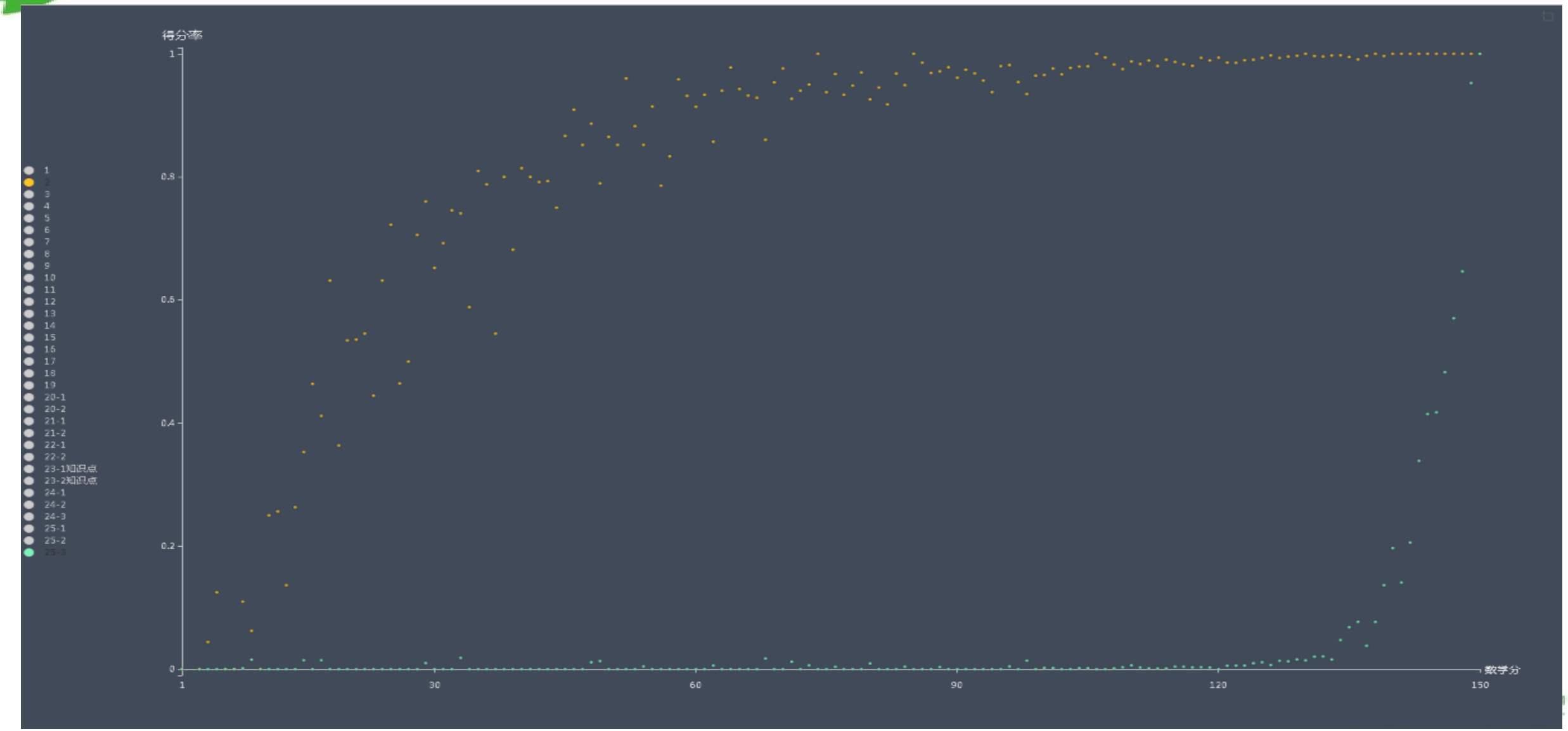
IRT (Item Response Theory)



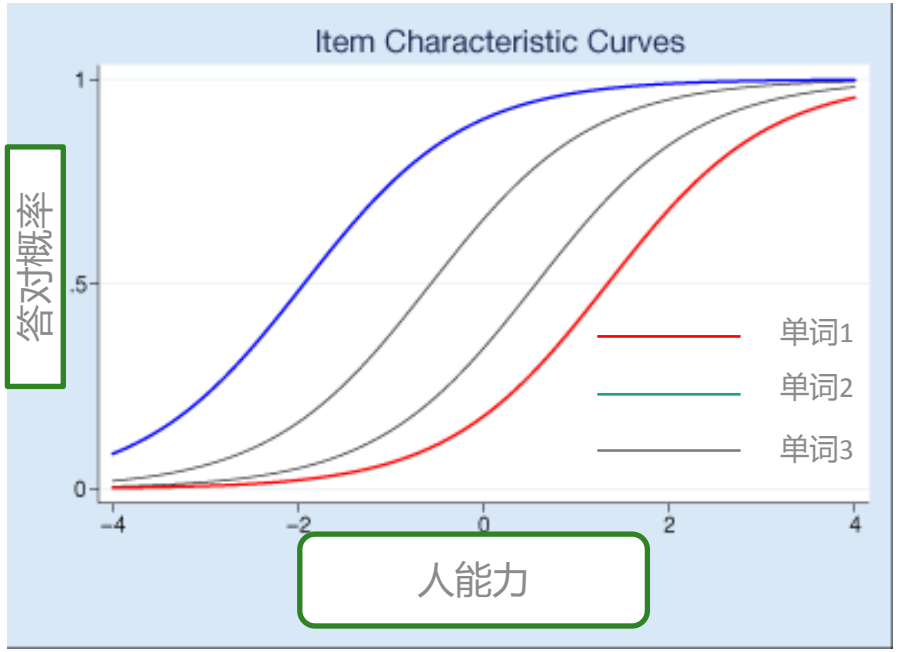
$$P(\text{答对}\mid \text{词难度，人能力}) = \frac{1}{1+e^{-(\text{人能力}-\text{词难度})}}$$

- 人能力越高，答对概率越大
- 词难度越高，答对概率越小，曲线越靠右
- 人能力=词难度, 则答对概率0.5

IRT (Item Response Theory)



如何估计人能力？



$$P(\text{答对}|\text{词难度，人能力}) = \frac{1}{1+e^{-(\text{人能力}-\text{词难度})}}$$

$$P(\text{答错}|\text{词难度，人能力}) = 1 - \frac{1}{1+e^{-(\text{人能力}-\text{词难度})}}$$

word	correct	词难度
depress	1	0.9
take	0	0.1
delight	1	0.3
explain	1	0.4

极大似然估计

$$\max_{\text{人能力}} \underbrace{\left(\frac{1}{1+e^{-(\text{人能力}-0.9)}}\right)}_{\text{depress}} \underbrace{\left(1-\frac{1}{1+e^{-(\text{人能力}-0.1)}}\right)}_{\text{take}} \underbrace{\left(\frac{1}{1+e^{-(\text{人能力}-0.3)}}\right)}_{\text{delight}} \underbrace{\left(\frac{1}{1+e^{-(\text{人能力}-0.4)}}\right)}_{\text{explain}}$$

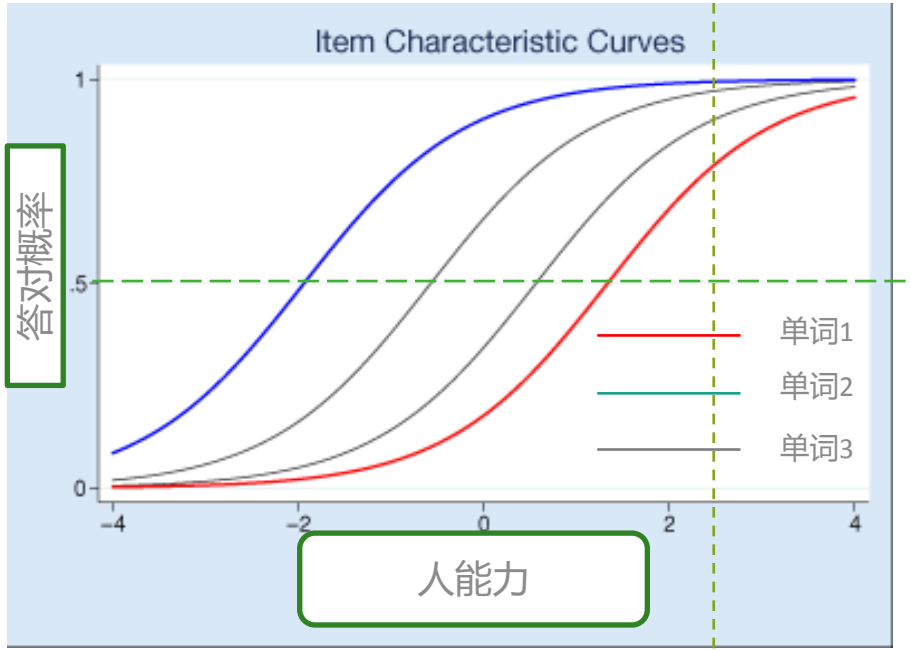
词 = 题 = 难度 = 词汇量

人能力

如何选下一个词？

- 选难度和当前能力估计最接近的，且用户没有回答过的词
  - 对于该词，P(答对|词难度，人能力)~0.5，即最不确定
  - 从而避免两种可能浪费时间的词(肯定会做，肯定不会做)

$$P(\text{答对}\,|\,\text{词难度，人能力}) = \frac{1}{1+e^{-(\text{人能力}-\text{词难度})}}$$



选红色的单词1

使用IRT根据用户数据进行难度校准

y	take	explain	huge	read	talk	term
用户1	1	0	1			
用户2	0	0		1		
用户3	1		1		1	
用户4		1		1	1	1
用户5	0	1	1			1
用户6			1	0	0	1

为什么不对每个词的正确率求平均得到难度值？

- 因为在词汇量测试里，每个用户所做的词都不一样
- 总共1000个词，每个用户只做了少于40道题目

$$\max_{u,d} \prod_{(i,m)\in S} \left(\frac{1}{1+\exp(-(u_i-d_m))}\right)^{y_{im}} \left(1-\frac{1}{1+\exp(-(u_i-d_m))}\right)^{1-y_{im}}$$

- u: 用户难度
- d: 单词难度
- 此问题具有全局最优解
- 在数据量很大的时候，可以采用随机梯度下降的方法优化该代价函数
- 计算出来的单词难度和冷启动时设置的大体趋势一样但是有区别
- 细节区别往往由于词和题目之间的存在gap
  - 简单的词出的选项混淆太强，导致学生选错
  - 难的词由于例句原因，可以猜出答案

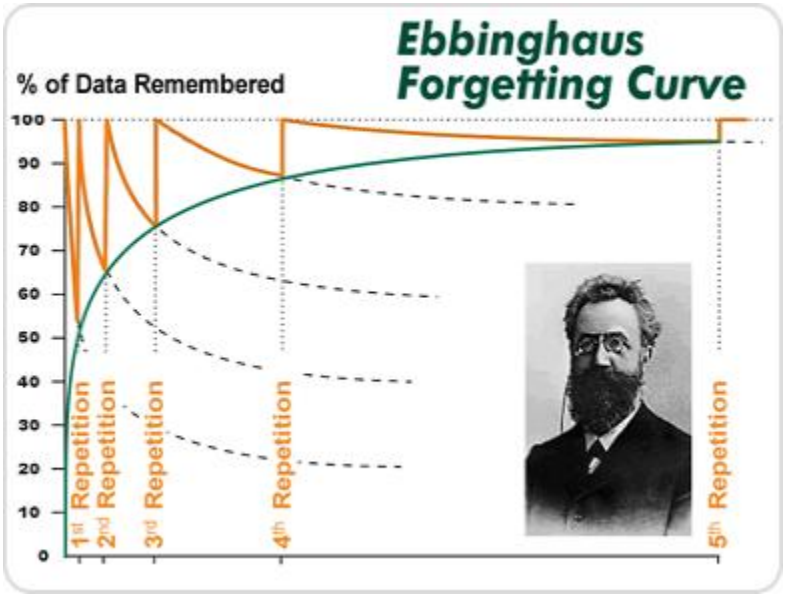
可以改进的方向

- IRT模型的改进，如何更好的建模用户答题(用户-词)矩阵
  - 混合IRT (如果存在多个学习路径)
  - 深度神经网络
    - 可以很好的解释现有的数据
- 推题策略的改进：
  - MDP?

记忆模型

记忆模型

艾宾浩斯记忆曲线



Supermemo 2004

anki

- 个性化复习策略
- 非个性化模型

机器学习模型

- Act-R
- IRT
- MCM
- Duolingo
- ...
  - 个性化复习策略
  - 个性化的模型

c.f.

- Predicting and Improving Memory Retention: Psychological Theory Matters in the Big Data Era
- [ACL16] A Trainable Spaced Repetition Model for Language Learning

记忆模型

艾宾浩斯记忆曲线→ 间隔效应→Act-R→MCM

$$P_r(recall) = m(1+ht)^{-f}$$

概率随着时间指数衰减:
其中m,h,f是常数，分别解释为初始学习的程度（0<m<1）,时间的缩放因子(h>0),以及记忆的衰减指数(f>0)
艾宾浩斯记忆曲线:y=1-0.56x^0.06

$$\text{Optimal ISI} = 0.097\text{RI}^{0.812}$$

多次学习对记忆的影响:间隔效应(Spacing effect)
两次学习的间隔记作ISI（intersession interval）,第二次学习和最后的测验的时间记作RI（retention interval）

Act-R：

$$m_n = ln\left(\sum_{k=1}^{n} b_k t_k^{-d_k}\right) + \beta$$

ACT-R假设每次学习会有不同的记忆概率轨迹，而且记忆概率随着时间的增长成幂函数衰减：t_k,d_k指的是第k条轨迹的记忆时间和衰减指数，β是和学生或者记忆事物有关的影响记忆强度的参数。 b_k指的是每条记忆轨迹的显著性，这个数越大表示一次学习的效果越好。

$$d_k(m_{k-1}) = ce^{m_{k-1}} + \alpha$$

轨迹的衰减和学习发生的时间点有关：这里c和α是常数，如果第k次学习和前一次的间隔比较短，会导致当前的一条衰减的比较快。

$$P_r(recall) = 1/(1+e^{\frac{\tau - m}{s}})$$

回忆的概率和记忆活性m单调相关：其中和是相应的参数。整个模型有6个自由的参数。

记忆模型

MCM提出了一个假设，每次新的学习学到的东西是分别存储在不同的轨迹中，而且会按照不同的速率衰减。虽然每条迹会指数衰减，这些轨迹的和随着时间的衰减是一个幂函数，举例来说，第i条轨迹，x_i的衰减如下面公式所示：

$$x_i(t+\Delta t) = x_i(t)\exp\left(-\Delta t/\tau_i\right)$$

其中是衰减时间常数，而且后续的轨迹具有比较小的衰减时间常数，,轨迹1-k使用了一个加权平均，最后合成了一个总的轨迹强度。

$$s_k = \frac{1}{\Gamma_k}\sum_{i=1}^{k}\gamma_i x_i$$

其中$\Gamma_k = \sum_{i=1}^{k}\gamma_i$ 。γ_i 是一个权重因子，代表了第i条轨迹的贡献，在总共k条轨迹中，记忆的概率是其中的最小值：

$$P_r(recall) = \min\left(1, s_k\right)$$

间隔效应发生的主要原因是轨迹的更新规则（Staddon et al.,2002）。一条轨迹只有在其它轨迹无法保持对材料的记忆的时候才会更新。这个规则影响了信息在不同发生频率和不同环境下的记忆效果。当一个材料被学习的时候，第i条轨迹贡献的上升和前面轨迹的总强度负相关：

$$\Delta x_i = \epsilon(1 - s_i)$$

其中是 ϵ 步长。

个性化策略 vs 个性化模型

学生历史记录

time	word	correct
1/1/2012 13:00	depress	1
1/1/2012 13:10	depress	0
1/1/2012 13:20	depress	1
1/2/2012 13:00	depress	1
1/2/2012 13:40	depress	0
1/3/2012 13:50	depress	0
1/3/2012 14:00	depress	1

- 每个学生的历史记录都不一样

\+

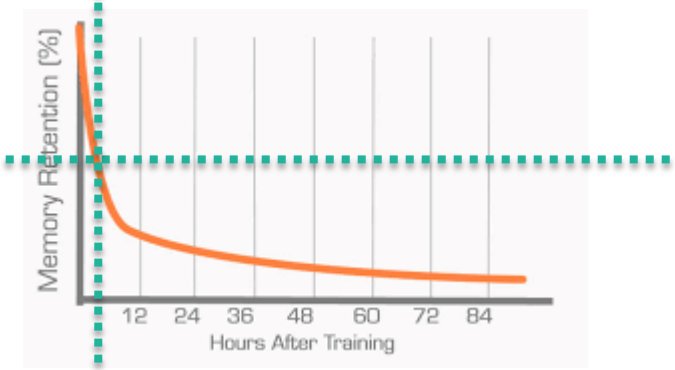
记忆模型

输出: 遗忘曲线

- 非个性化记忆模型
  - 艾宾浩斯曲线
  - Supermemo
- 个性化记忆模型
  - 每个学生的记忆能力不一样
  - 每个学生记忆模型也不相同

=

遗忘曲线



下次复习时间

time	word	correct
1/7/2012 13:00	depress	1

- 每个学生的历史记录都不一样
- 同样的模型➔不同的复习策略
- 不同的模型➔不同的复习策略

当前词场采用的复习机制

1

单词1历史记录
单词2历史记录
单词3历史记录
单词4历史记录

+

记忆模型

=

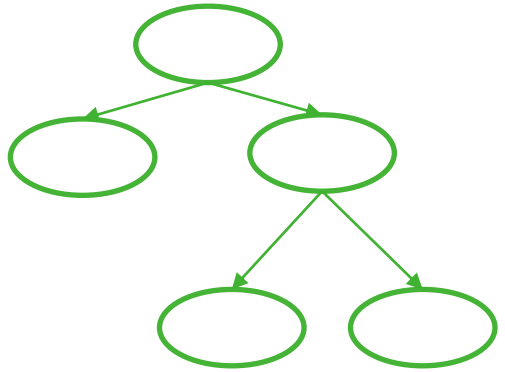
下次复习时间

下次复习时间	单词
1/7/2012 13:00	单词1
1/7/2012 13:00	单词2
1/8/2012 13:00	单词3
1/9/2012 13:00	单词4

2

对于下次复习时间接近的单词，根据以下特征的组合进行重新排序

- 上次复习时间
- 上次复习是否作对
- 上次反应时间
- 历史做错次数
- 历史作对次数
- …



正在研究的复习机制

- 个性化记忆模型

通过分析大量数据，考察记忆和以下特征之间的关系

- 背词间隔
- 历史准确率
- 一次学习的量 (疲劳控制)
- 答题反应时间
- 以前背过的单词
- …

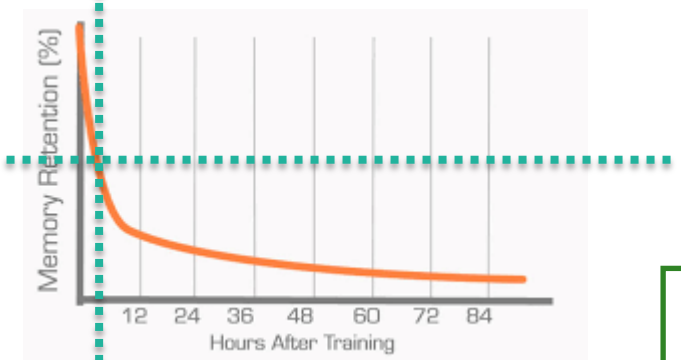
采用的技术

- Logistic Regression
- Multiple Task Learning/协同过滤

正在研究的复习机制

- 优化的复习/学习策略

遗忘曲线



下次复习时间

time	word	correct
1/7/2012 13:00	depress	1

智能和个性化的优化策略

<u>在特定的约束条件下</u>，寻找最优的复习方案

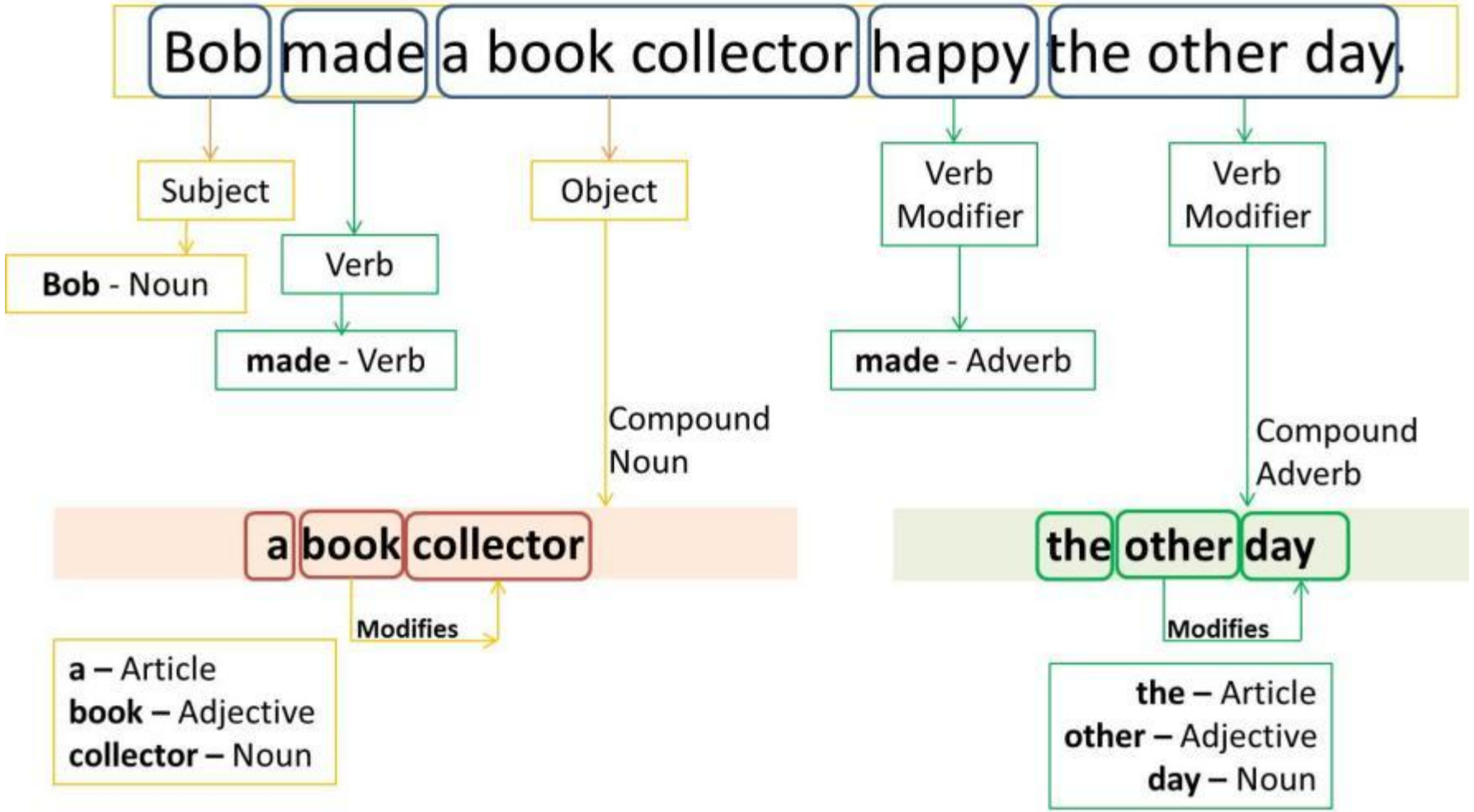
- 学生只愿意每天学习/复习少量的词，如何调整词的顺序？

采用的技术

- MDP (给定记忆模型)
- Reinforcement Learning (Model Free)

词性标注

词性标注



帮助学生理解复杂句子

- 区分多义词
- 推荐更加合适的例句
- 方便词书制作

词性标注 - 现状

- 英语，现成的软件
  - Nltk
  - Stanford Pos Tagger (https://nlp.stanford.edu/software/tagger.shtml)
- 其他语种，词性标注作为分词的附加结果
  - 中文
    - 分词：Jieba, Mecab
  - 日韩:
    - 分词：Mecab
  - 其他语种：
    - 构建语料库，打标
    - 训练自己的Pos模型

分词 - 难点

- 一个字
  - 既可以和前面的字构成一个词
  - 又可以和后面的字构成一个词
  - 还可以单独成为一个词

“结婚的和尚未结婚的”

- 结婚的/和/尚未/结婚的
  - NPN
- 结婚的/和尚/未结婚的
  - NNN

分词 - approach

- 基于匹配的
  - 不断匹配最长的词，直到把句子划分完
  - “北京大学/生前/来/应聘”
  - “北京/大学生/前来/应聘”
- 最少词数法
  - “为人/民办/公益”
  - “为/人民/办/公益”
- …
- 上述方法的组合

- 基于序列标注的机器学习方法
  - 结婚的/和/尚未结婚的
    - NPN
  - 结婚的/和尚/未结婚的
    - NNN
  - 似乎NPN更加符合语法规则
  - 序列化推断问题

分词 – 使用序列化标注模型训练自己的分词器

大数据挖掘工程师

词	词性	拼音特征
大数据	n	da_shu_ju
挖掘	v	wa_jue
工程师	n	gong_cheng_shi

- 对于每个句子, 需要标注
  - 分词
  - 每个词的词性
  - 可能有助于分词的特征 (x)
- 特征
  - Unigram特征
  - Bigram特征

$\vec{s} = s_1, s_2, \ldots s_n \qquad \vec{o} = o_1, o_2, \ldots o_n$

HMM

$$P(\vec{s},\vec{o}) \propto \prod_{t=1}^{|\vec{o}|} P(s_t \mid s_{t-1}) P(o_t \mid s_t)$$

+x ↓

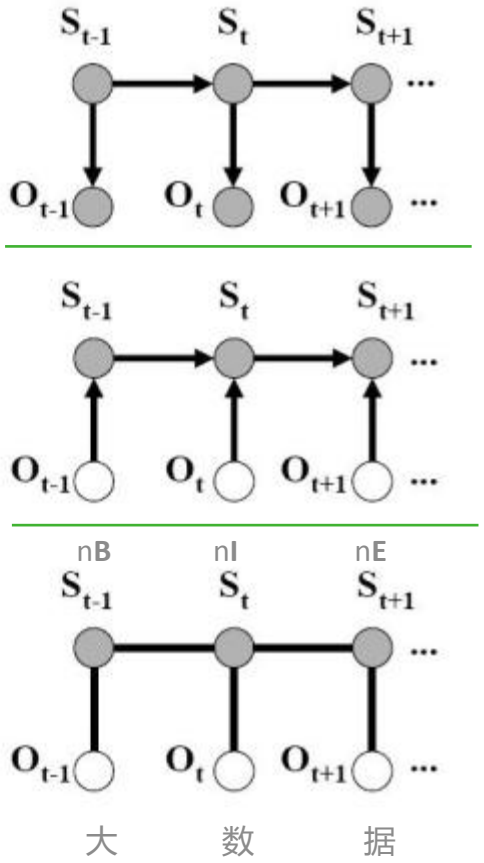
MEMM

$$P(\vec{s} \mid \vec{o}) \propto \prod_{t=1}^{|\vec{o}|} P(s_t \mid s_{t-1}, o_t)$$

$$\propto \prod_{t=1}^{|\vec{o}|} \frac{1}{Z_{s_{t-1},o_t}} \exp\left(\sum_j \lambda_j f_j(s_t, s_{t-1}) + \sum_k \mu_k g_k(s_t, x_t) \right)$$

CRF

$$P(\vec{s} \mid \vec{o}) \propto \frac{1}{Z_{\vec{o}}} \prod_{t=1}^{|\vec{o}|} \exp\left(\sum_j \lambda_j f_j(s_t, s_{t-1}) + \sum_k \mu_k g_k(s_t, x_t) \right)$$



B: 词开始
I: 词中
E: 词节数

Q&A

学习，成为更好的自己