

### **顾客至尚 (Customer Obsession)**

领导者从客户入手，再反向推动工作。他们努力地工作以赢得并维系客户对他们的信任。虽然领导者会关注竞争对手，但是他们更关注客户。

### **主人翁精神 (Ownership)**

领导者是主人翁。他们会从长远考虑，不会为了短期业绩而牺牲长期价值。他们不仅仅代表自己的团队，而且代表整个公司行事。他们绝会说“那不是我的工作”。

### **创新简化 (Invent and Simplify)**

领导者期望并要求自己的团队进行创新和发明，并始终寻求使工作简化的方法。他们了解外界动态，从各处寻找新的创意，并且不局限于“非我发明”的观念。当我们开展新事物时，我们要接受被长期误解的可能。

### **决策正确 (Are Right, A Lot)**

领导者在大多数情况下都能做出正确的决定。他们拥有卓越的业务判断能力和敏锐的直觉。

### **好奇求知 (Learn and Be Curious)**

领导者从不停止学习，并不断寻找机会以提升自己。领导者对各种可能性充满好奇并付于行动加以探索。

### **选贤育能 (Hire and Develop the Best)**

领导者不断提升招聘和晋升员工的标准。他们表彰杰出的人才，并乐于在组织中通过轮岗磨砺他们。青出于蓝，冰源于水，领导者培养的也是领导者，而且他们严肃地对待自己育才树人的职责。

### **最高标准 (Insist on the Highest Standards)**

领导者有着近乎严苛的标准——这些标准在很多人看来可能高得不可理喻。领导者不断提高标准，激励自己的团队提供优质产品、服务和流程。领导者会确保任何问题不会蔓延，及时彻底解决问题并确保问题不再出现。

### **远见卓识 (Think big)**

局限性思考只能带来局限性的结果。领导者大胆提出并阐明大局策略，由此激发良好的成果。他们从不同角度考虑问题，并广泛寻找服务客户的方式。

### **崇尚行动 (Bias for Action)**

速度对业务影响至关重要。很多决策和行动都可以改变，因此不需要进行过于广泛的推敲。我们提倡在深思熟虑前提下进行冒险。

### **勤俭节约 (Frugality)**

我们尽量不在与客户无关的地方花钱。勤俭节约可以让我们开动脑筋、自给自足并不断创新。在人员数量、预算规模或固定开支方面，没有额外的投入。

### **赢得信任 (Earn Trust)**

领导者会真正敞开胸怀、认真倾听，并愿意谦逊地审视自己最坚定的信念。

### **刨根问底 (Dive Deep)**

领导者深入各个环节，随时掌控细节，并经常进行审核。不遗漏任何工作。

### **敢于谏言 服从大局 (Have Backbone; Disagree and Commit)**

领导者必须要能够不卑不亢地质疑他们无法苟同的决策，哪怕这样做让人心烦意乱，精疲力尽。领导者要信念坚定，矢志不移。他们不会为了保持一团和气而屈就妥协。一旦做出决定，他们就会全身心地致力于实现目标。他们愿意支持不受欢迎或难获理解的意见。

### **达成业绩 (Deliver Results)**

领导者会关注其业务的关键决定条件，确保工作质量并及时完成。尽管遭受挫折，领导者依然勇于面对挑战，从不气馁。

为什么亚马逊

吹亚麻 top online retailing company, many of its products, Alexa, Kindle, fire tablet and TV, Amazon cloud service, etc. really make people's life more convenient.

to be world's most customer-centric company inspired me a lot. I've heard a story that in early days, when Amazon couldn't promise that an order would arrive by a certain date. Amazon invested a lot of resources to develop the capability to make aggressive delivery estimation and keep them. In many ways, this was what Prime was born of. Based on common business sense I think a company with such emphasis on customers will always be successful. 吹企业文化 Good corporate culture. Many talented engineers work with new founder's passion to build world scale innovative services.

Therefore, this is a place where I can learn a lot, and use my skills and experience to make things happen with a team.

## 紧急的截止日期

### Data Next 实习

从 slicing 升级到 erosion and dilute, 效率大提升

Manager 和客户说了, 客户要求下周展示

虽然算法部分可以更新完成, 但数据清洗部分要有很大改动, 因为原先的输入是, 现在的是解决办法 先完成算法部分, 做好数据清洗的初步流程图, 按流程手工清洗几个例子用于展示, 之后完成 coding 与测试和修正

### 最近工作

Release 到了, 队友活没干完, 一看熟悉我来干。有的是给客户的, 有的是帮 PM 在 stg 试验, 重要优先, 给客户的不折不扣。 其中一个是提供团队成员报表, 一个是现成的 api, 大领导显示慢。自己写新的服务, 改动大只有半天来不及。询问产品经理, 大领导十几个不重要, 先用现成的上 Business server。之后再花一天写好新的服务测试完打上去

## 最挑战/最有趣/最骄傲的项目/项目中遇到的问题/学新东西

### 中科院科研

一开始啥不懂, 看论文查谷歌看技术博客

对之前人的工作有观点后, 和导师与另外一个研究生讨论, 获得更全面的看法

想到一个觉得可行的点子, 马上去代码实现, 虽然经常被证实效果并不好

然后有一次: I found a new method which is several times faster than SOTA empirically, and the next step is to do lots of experiments. In the experiments, the brute force method is used as a benchmark. We measure each method's performance by how many times it is faster than the brute force method. When doing the experiments, a strange situation appeared. The brute force method's performance seems to be faster than usual. Thus the SOTA methods cannot reach the performance as they claimed in their papers in my experiment. Though my method outperforms the SOTAs significantly in my experiments, the results may be challenged because this. After consulted with advisor, he confirmed that my codes are correct, and said not to worry about the results as they are the real results on that specific server setup. But I'm

still curious about why brute force search is that fast on that machine. I want my research paper to be at the highest standard, and do not neglect details that may be hard to explain. So I began to check the influence of hardware, compilers and coding techniques. After several days endeavor, I finally found out why brute force method is that fast on this machine. The compiler is set to O2 level optimization, this level optimization will optimize for loops on arrays like `data[i]`, but will not optimize for loops on arrays like `data[idArray[i]]`. The solution to this is to repeat the search loop for multiple times. By repeating, the compiler will optimize for loops of later kind. And that was the last major challenge in my experiments. Though I spent a lot of time redo most of the experiments because of this, my paper finally published on October's Pattern Recognition Journal.

### 最近的工作

要做 MQ 的功能，屁都不懂。问队友，不清楚。队友给了示例代码，仿写能跑。各种谷歌博客，高度封装，读代码。又来了新任务也能写了

### 帮助队友/做额外的工作/最近的项目/快速决定/有限参考资料下决定/承担风险

#### 信息检索小组项目

我干前端图表分析，后端的同学出了故障。Solr 服务可以由浏览器端访问，但不能从后端服务器访问

我在尝试了直接把后端框架装在 Solr 服务器上，换用其他的后端框架，不成功。因为这个故障拖了好久，DDL 临近

在调查思考后，我发现仅用 JS 和 AJAX 应该可以实现所有要求的功能，而不需要后端部分。但我不是很确定。在咨询 TA 和朋友确认几个关键功能可以实现后，我建议队友采用前端加 Solr 的形式

我们很快用 JS 实现了所有的功能，项目在 DDL 前优质完成

之后我去查看的他们的代码，发现由于服务器端口设置原因无法接受外部请求，由于在 django 的原因 python 无法像平常一样读取 solr，修正后可以了

### 最近的工作

某次更新后我突然发现网页对于特定输入慢，组会上前端演示说没有问题。检查后端 API 速度没问题，去找前端，他说浏览器问题。我查了了前端的 sourcetree 确实没有更新，非常疑惑，用空闲时间找了两天。终于发现某个 NG 的 API 在更换 NG 版本后超级慢，开始手写 css 代替他。此时用户报告加载不出，我火速 hotfix。

### 长短期利益/错过 DDL/未经领导同意/失败经历

我们基本做好了，manager 向客户说了，客户要求下周上线交付。某博士技术顾问来了一趟，提到 CTPN 效果不错。为了给客户性能最好的，我们讨论后决定尝试

我们先收尾了原方案再测试 CTPN

DDL 前两天，我们告诉 manager 搞 CTPN 有进展，请 manager 选择。Manager 给了 DDL 延展让我们继续做下去

经过多种测试 CTPN 略有性能提升，但是自身运算开销和维护难度增加，成本增加，加上可解释性不强对性能稳定性难以预测，放弃

Manager 肯定了我们的尝试，表示 AI 方法是大趋势

## 重要决策

### 中科院科研

一开始提了个降维到实数再搜索 ANN 的方法，与导师交流，导师认为可行

做了个 PCA 和 LDA 的，效果不好

邻座的研究生给了我篇降维的综述，我从实现简单的开始一个个试

然后有一次：I found a new method which is several times faster than SOTA empirically，汇报导师后决定就搞这个

## 错误

### 中科院科研

最后一个实验是三维重建。我的方法用于两张图片描述子的匹配，每两张要互相匹配一次  
因为自己疏忽，我的第一次和第二次匹配最邻近和次临近都是有设置距离比最大阈值的

跑完之后看对照组的代码发现第二次匹配是不设最邻近次临近阈值的

虽然从匹配结果来看已经能证明我的方法优于原有方法，而且跑匹配非常慢，一个数据集就要 2, 3 天

但我还是立刻报告导师我的错误，并立即重新跑所有的匹配

在紧迫的时间下赶上了投稿的 DDL

### 最近工作

有个 API 提供完成比例，不小心多留了一行+=1，虽一般情况不会改变百分比，数据特别小会。（客户无小事）QA UAT 都没发现。主动向 manager 认罪，重新 deploy prod

### 问问题

What kind of process do you have when designing features?

When evaluation an engineer during performance review, what do you care the most?

what's the most unique part about working at xxx that you've never experienced before.