2020.4月总结

月报主要分为两部分，第一部分为对4月工作的总结，第二部分为对5月工作的计划。

1.4月工作总结

本月工作主要分为部分：
1. 多模态相关论文阅读
2. 多模态平台网站建设
3. 视觉实体链接论文转投
4. 代码实践（数据挖掘竞赛）
5. 相关在线讲座的参与和笔记

以下是每项工作的具体进展：

1. 多模态相关论文阅读

（1）Who, Where, and What to Wear? Extracting Fashion Knowledge from Social Media

这是一篇发表在2019年ACM Multimedia上面的一篇关于知识抽取的文章，在这篇论文中，提出了一种新的方法来自动获取社交媒体上的时尚知识，从图像、文本和元数据的多种模式中，统一了三种场合、人物和服装的发现任务。和我们的视觉实体链接任务形式上面较为相似，均是提出一个之前没有的较为新颖的任务，作者的数据集以及实验设定有一定的可学习之处。

为了减轻人工标注的沉重负担，此文引入了一个弱标签建模模块，它可以有效地利用机器标记的数据，是过滤后的高质量数据补充。在实验中，提供了一个基准数据集，并从定量和定性的角度进行了实验，结果证明了该模型在服装概念预测中的有效性，以及通过综合分析提取知识的有效性。

此文的挑战主要分为以下几点：

1.时尚概念的缺乏：首先，从社交媒体内容中提取时尚知识很大程度上依赖于对时尚概念预测的表现，这是因为各类用户在社交媒体上发布的视觉图像大多拍摄自自然场景，很难从中发现时尚概念(如服装属性、场合等)。因此如何联合检测自然场景图像中的时尚概念进行知识建构是一项困难而又关键的任务。

2.数据集的缺乏：社交媒体数据缺乏足够的时尚概念标签，而这些标签对于时尚知识的构建至关重要。自动获取的时尚知识的质量在很大程度上取决于语义水平的时尚概念学习，手动注释大量数据是昂贵和耗时的。现有的数据集主要来源于电子商务网站，只关注一组特定的布属性，不能用于检测场合类型或人员身份。

本文的主要贡献如下：

1. 此文提出了一种基于情境化的服装概念学习模块的服装知识提取新方法，该模块能够捕捉场景、服装类别和属性之间的依赖关系。

2. 利用带有弱标签的机器标签数据来丰富学习模型，使用标签校正模块来控制噪声。

3. 提供了一个基准数据集，并从定量和定性的角度进行了大量的实验，以证明模型在时装概念预测方面的有效性和提取的知识的有用性。

本文提出的模型架构：

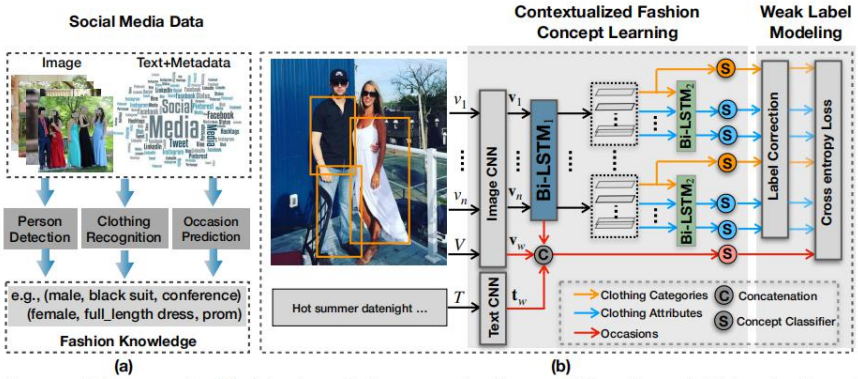
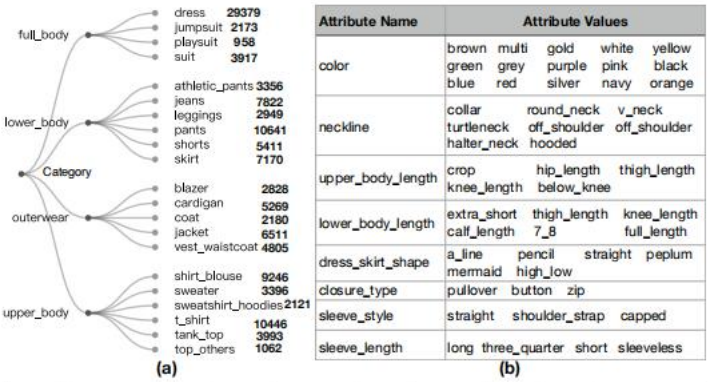


Figure 2: (a) The overall framework of fashion knowledge extraction from social media, and (b) the pipeline of our proposed contextualized fashion concept learning model. Note that the extraction of person attributes (*e.g.*, gender and age) is performed using an off-the-shelf tool for simplicity.

首先定义了三个子任务，是基于特定情境（时尚领域）的一些知识图谱类的子任务，分别为人物属性检测、服装类别属性检测、场景预测。针对每一个子任务进行不同的实验设计，这样可以使得实验更加的饱满丰富。

数据集构建：构建过程较为普通，先是从一个专业的时尚穿搭网站上面爬取数百万帖子，同时对帖子进行自动过滤和手动过滤（这是构建数据集过程中我比较关心的一点，其实也是数据集构建的关键之处），首先利用预训练的对象检测模型来检测人的身体和面部，然后过滤掉那些没有脸和身体的图像，对齐在同一图像的身体和脸，并删除那些没有正常大小的身体或脸的图像，最后确保图片是真正由用户生成的，但不是广告或海报（手动过滤）。

同时为数据集构建一个本体作为数据集的标注描述，通过本体的形式展示数据集的格式会比较清晰而且学术化。



Attribute Name	Attribute Values
color	brown multi gold white yellow green grey purple pink black blue red silver navy orange
neckline	collar round_neck v_neck turtleneck off_shoulder off_shoulder halter_neck hooded
upper_body_length	crop hip_length thigh_length knee_length below_knee
lower_body_length	extra_short thigh_length knee_length calf_length 7_8 full_length
dress_skirt_shape	a_line pencil straight peplum mermaid high_low
closure_type	pullover button zip
sleeve_style	straight shoulder_strap capped
sleeve_length	long three_quarter short sleeveless

(b)

Dating	Daily Work	Travel	Sports	Conference	Prom	Shopping	Party	School	Wedding Guest
9003	7634	7713	6858	9139	9334	6877	8010	7327	8734

(c)

Figure 3: The statistics of the FashionKE dataset, consisting of 21 categories, 8 types of clothing attributes, and 10 common types of occasions.

（2）Semantic Video Entity Linking based on Visual Content and

Metadata

这是一篇 2015 年发表在 ICCV 的关于视频实体链接的文章。视频实体链接是将在线视频连接到语义知识库中的相关实体，它可以支持各种基于视频的应用，包括视频检索和视频推荐。在提出的框架中，视频首先使用基于文本的方法链接到实体候选对象。接下来，根据可视内容对实体候选项进行验证和重新排序。为了正确处理视觉内容匹配中的巨大变化，使用了学习多个实例度量，学习针对这个特定匹配问题的“set to sequence”度量进行评测。

此文的挑战主要分为以下几点：

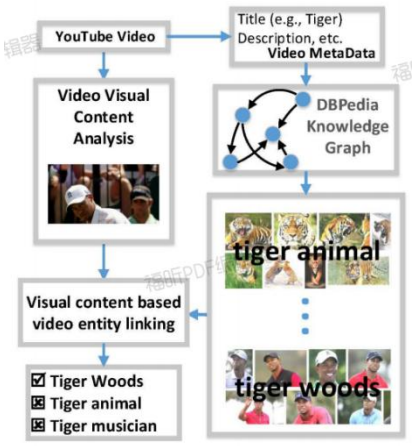
1. 实体种类的集合是非常多样化的，比如人、地方、艺术作品和人工制品，而词汇量和视觉变化使得开发基于分类的模型变得不可行。
2. 视觉内容信息只是视频的一部分。例如，链接某些实体的信息可能只包含在音频轨道中。
3. 传统的视频分析技术对于在线应用来说是非常昂贵的。

此文的主要贡献如下：

1. 演示了如何在视频领域使用可视化内容来增强实体链接。
2. 提出一种用于序列匹配的度量学习范式 MIML-Struct，以解决具有结构平稳性的视频实体连接问题。
3. 提出了一个新的开源数据集和视频发现框架，用于未来视频实体的链接研究。

在此文中，提出通过度量学习来解决视频实体链接，通过学习成对图像之间的语义距离来自适应地度量相似场景或传感器布局。对成对的图像学习度量，对(集合、序列)对给出监督标签，指示一个实体(由图像集表示)是否与视频(由图像序列表示)匹配。使用多实例度量学习(MIML)来处理这个问题中的标签模糊性。进一步提出了 MIML 的变种来捕获视觉实体点样结构中观察到的两种约束（结构平滑和时间平滑）。

主要的在线框架如下所示：

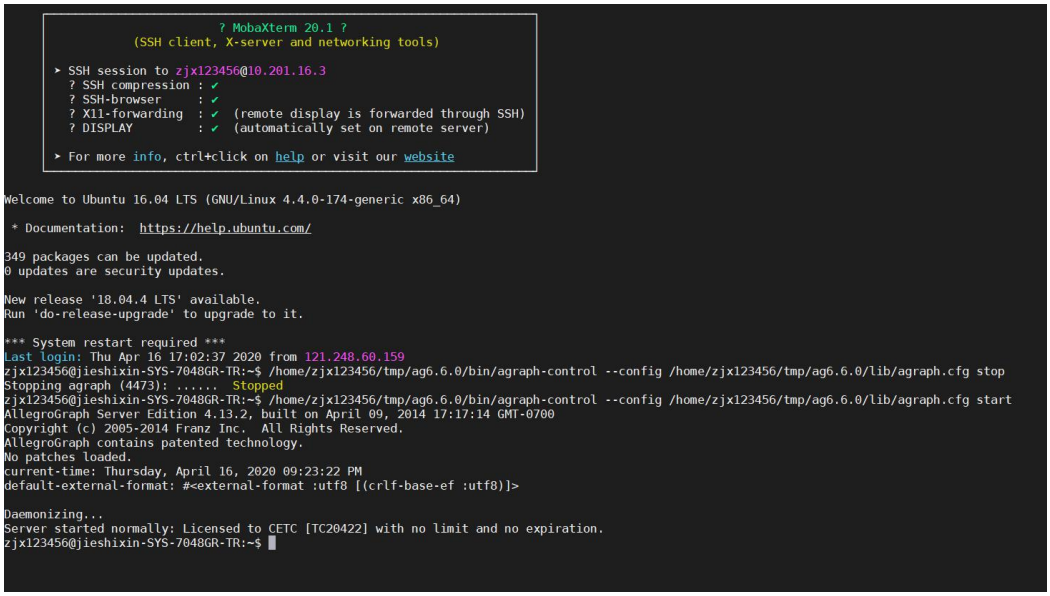


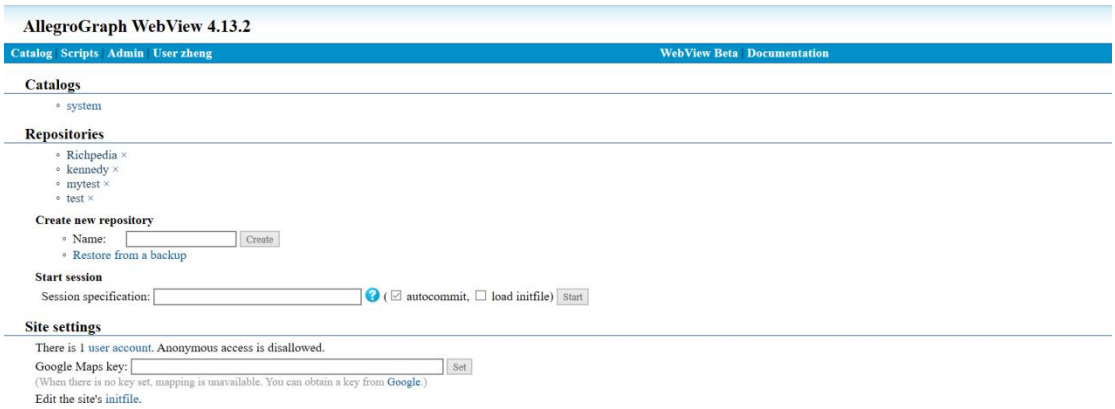
此文还通过 YouTube 爬取、实体种类分类、视频搜索查询选择、视频采集和实体注释和生成图像的视觉特征几个部分生成了一个视频实体链接数据集，生成过程值得我们借鉴。

2. 多模态平台网站建设

（1）构建 agraph 数据库并在服务器上进行安装和访问

这一部分了解图数据库 agraph 的相关知识，利用 agraph 数据库破解版来展示 Richpedia，将 Richpedia 三元组填充到 agraph 数据库中，在 linux 服务器上搭载了并配置了 agraph 服务器，并且成功的进行了局域网的访问。





（2）利用三元组的概念，构建了一个实际的数据库，并且实现了 sparql 查询

SPARQL 语言是用来处理 RDF 数据(三元组数据)的一种数据库语言。其可以定义命名空间，并且根据实现多种查询操作。命名空间在数据库中实际起标识资源归属的功能，但是在数据库本身中只是一种标识，只有结合实际的应用时才会产生意义。

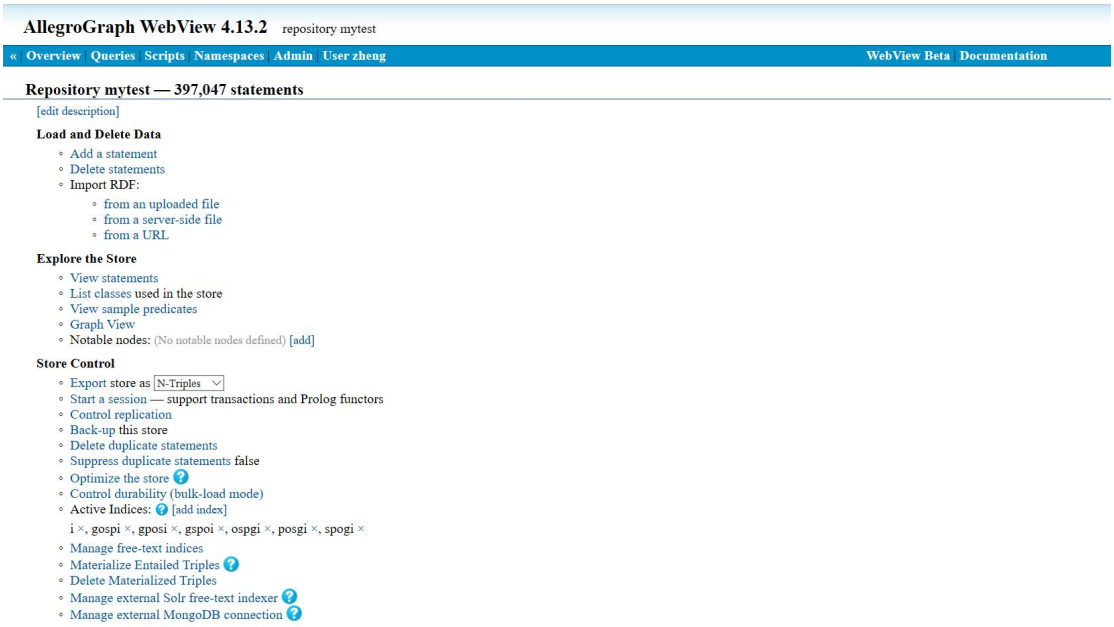
SPARQL 本身支持插入 INSERT 操作和查询 SELECT 操作，插入操作类似于常见的数据库系统，比较好理解。因为 Agraph 本身支持了大规模数据上传至数据库

的功能，直接插入用的不是很多，一般是使用 Agraph 自带的 Import RDF 功能上传三元组文件。

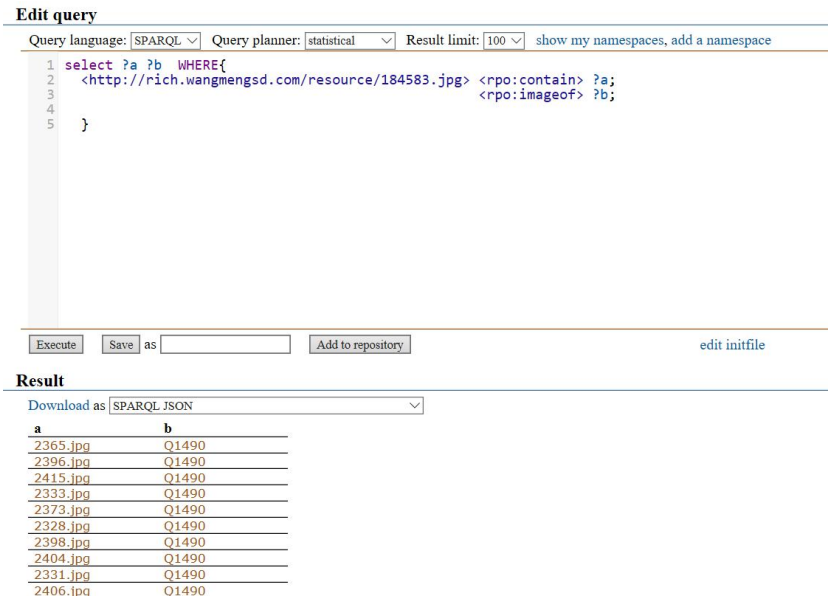
SPARQL 的查询操作则表现出了三元组的关系性，实现了一种基于关系的查询。直接的查询语句一般格式是 select ?x where {条件}，条件中可以指明主谓宾中的任意部分将一部分作为查询对象，数据库会返回满足以给条件的数据项，比如 richpedia 中可以指定图片编号和 contain 谓词来得到其包含的对象的编号。此外，查询机制本身还提供了诸如 OPTIONAL(可选条件)和 FILTER(过滤条件)的关键字，从而使得查询的结果更加灵活。

SPARQL 的一个重要功能是可以在已有的元组基础上通过查询语句直接构建新的关系和新的元组，使得数据库本身变得更加完整。比如数据库中只有父母关系和兄弟关系，但是实际上可以把构造和查询结合起来构建具有 uncle 关系的新元组。这为使用数据库的人提供了很大的便利性。

除此之外，与传统数据库类似，agraph 也提供了一些普遍的操作，比如集合操作，排序操作，限制操作等常用操作。



基于 Richpedia 构建的数据库



Richpedia 中对同一个对象进行多条件限制的 SPARQL 查询操作

将原始的三元组关系 json 通过文本处理转换为了一种 n 元组文件，并使用 AGraph 自带的 RDF 上传功能将数据存储至数据库并在其基础上实验了 SPQRQL 语言的使用。统计了部分单边查询的响应时间，大概为 100ms。

json 本身是一种网络中的数据传输格式，相比 xml 语言，其有着很好的结构性，使得其比起杂乱的 xml 语言更好处理，加上其通过处理可以成为任何编程语言可以理解的数据，其有着广泛的应用。

```
<http://rich.wangmengsd.com/resource/184583.jpg> <rpo:contain> <http://rich.wangmengsd.com/resource/2325.jpg>
<http://rich.wangmengsd.com/resource/184583.jpg> <rpo:contain> <http://rich.wangmengsd.com/resource/2326.jpg>
<http://rich.wangmengsd.com/resource/184583.jpg> <rpo:contain> <http://rich.wangmengsd.com/resource/2327.jpg>
<http://rich.wangmengsd.com/resource/184583.jpg> <rpo:contain> <http://rich.wangmengsd.com/resource/2328.jpg>
<http://rich.wangmengsd.com/resource/184583.jpg> <rpo:contain> <http://rich.wangmengsd.com/resource/2329.jpg>
<http://rich.wangmengsd.com/resource/184583.jpg> <rpo:contain> <http://rich.wangmengsd.com/resource/2330.jpg>
<http://rich.wangmengsd.com/resource/184583.jpg> <rpo:contain> <http://rich.wangmengsd.com/resource/2331.jpg>
<http://rich.wangmengsd.com/resource/184583.jpg> <rpo:contain> <http://rich.wangmengsd.com/resource/2332.jpg>
<http://rich.wangmengsd.com/resource/184583.jpg> <rpo:contain> <http://rich.wangmengsd.com/resource/2333.jpg>
<http://rich.wangmengsd.com/resource/184583.jpg> <rpo:contain> <http://rich.wangmengsd.com/resource/2334.jpg>
<http://rich.wangmengsd.com/resource/184583.jpg> <rpo:contain> <http://rich.wangmengsd.com/resource/2335.jpg>
<http://rich.wangmengsd.com/resource/184583.jpg> <rpo:contain> <http://rich.wangmengsd.com/resource/2336.jpg>
<http://rich.wangmengsd.com/resource/184583.jpg> <rpo:contain> <http://rich.wangmengsd.com/resource/2337.jpg>
<http://rich.wangmengsd.com/resource/184583.jpg> <rpo:contain> <http://rich.wangmengsd.com/resource/2338.jpg>
<http://rich.wangmengsd.com/resource/184583.jpg> <rpo:contain> <http://rich.wangmengsd.com/resource/2339.jpg>
<http://rich.wangmengsd.com/resource/184583.jpg> <rpo:contain> <http://rich.wangmengsd.com/resource/2340.jpg>
<http://rich.wangmengsd.com/resource/184583.jpg> <rpo:contain> <http://rich.wangmengsd.com/resource/2341.jpg>
<http://rich.wangmengsd.com/resource/184583.jpg> <rpo:contain> <http://rich.wangmengsd.com/resource/2342.jpg>
<http://rich.wangmengsd.com/resource/184583.jpg> <rpo:contain> <http://rich.wangmengsd.com/resource/2343.jpg>
<http://rich.wangmengsd.com/resource/184583.jpg> <rpo:contain> <http://rich.wangmengsd.com/resource/2344.jpg>
<http://rich.wangmengsd.com/resource/184583.jpg> <rpo:contain> <http://rich.wangmengsd.com/resource/2345.jpg>
<http://rich.wangmengsd.com/resource/184583.jpg> <rpo:contain> <http://rich.wangmengsd.com/resource/2346.jpg>
<http://rich.wangmengsd.com/resource/184583.jpg> <rpo:contain> <http://rich.wangmengsd.com/resource/2347.jpg>
```

最终生成的元组文件的一部分数据

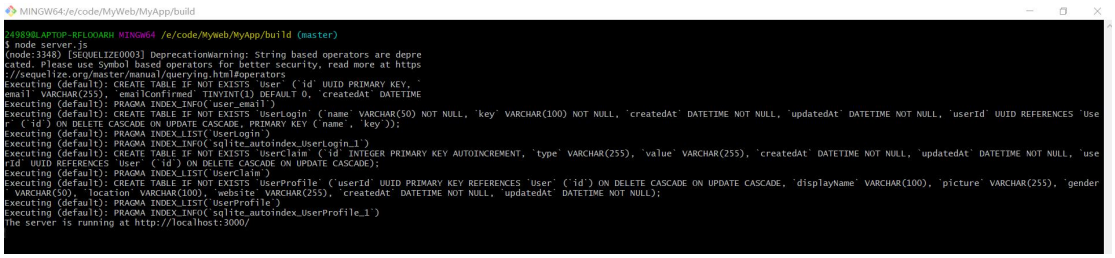
（3）学习前端语言以及 React 框架和 node.js，并根据已有的框架搭建了一个初步的本地可访问网站

学习了 html，css 和 js 语言，对网页开发有了一个初步的理解。html 是标记网页内容的语言，其决定了网页本身的结构和内容，是一种标签语言。css 与 html 有着紧密的联系，其起到修饰网页内容的作用，比如决定字号，颜色以及其他效果。因为以上的部分都是静态语言，所以如果希望和用户交互，就需要一套动态的语言。js 负责写脚本，使得控件对于人的互动有所反应，比如给按钮写一个反应函数或者一个计时器等。现在因为有了 react 框架，使得三种部分进一步融合，可以直接在 js 中使用 react 的方法返回 html 的内容，使得开发变得方便。

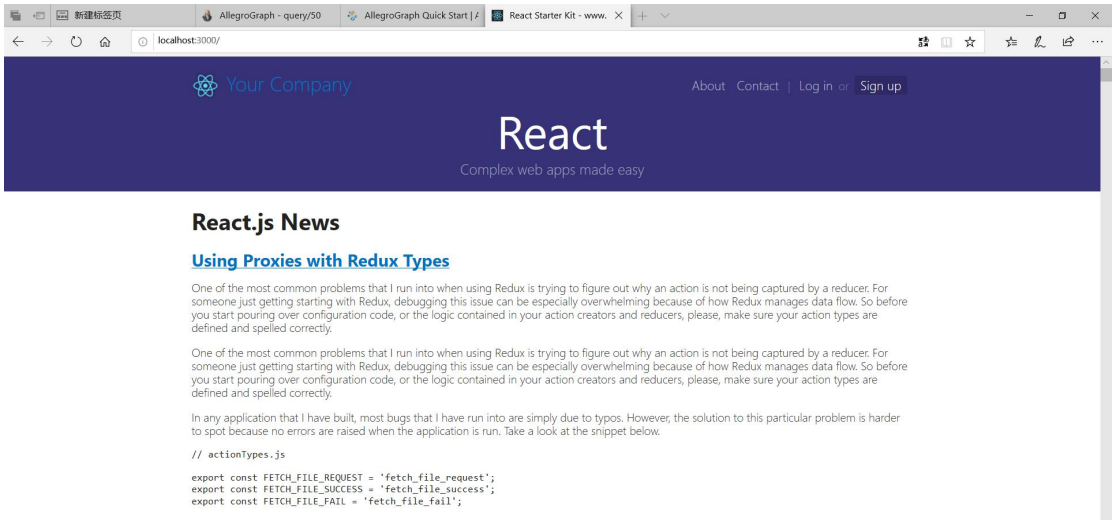
js 只是一种很基础的语言，就像 python 如果不引入库的话，其能做的事情很有限。react 就是 js 常用的一个框架或者说一个库，其封装了很多 js 的操作和比较常用的功能，使得开发得到进一步的简化，它的思想是尽可能让上层模块控制其子模块，子模块最好不要保存其自己的状态，从而实现更好的控制功能。

除此之外，node.js 的运行环境也进一步降低了网站开发的负担。该环境本身是运行在浏览器上的一个 js 的运行环境，其价值在于服务器端也可以理解 js 代码，同时因为统一性，js 比起传统的后端语言有着更好的容错性。所以正在学习如何利用 js 开发出一整个网页。

现在的进度是已经根据 react 的代码和教程使用 node.js 构建起了一个本地可以访问的服务器。



使用 node 环境运行服务器



本地访问到的网站模板

3. 视觉实体链接论文转投

参考 reviewer 和 meta-reviewer 的评审意见，对视觉实体链接论文进行修改，转投其他会议。

4. 代码实践（数据挖掘竞赛）

在读论文的同时，每天也注意代码实践能力的提升，寻找往年数据挖掘竞赛经典例题，自主实现一些机器学习算法来处理竞赛题目，提升理论能力的同时加强自己的编码能力。

主要实现的数据挖掘例题有两个，第一个是基于用户近期数据的音乐推荐题目，第二个是基于短视频用户信息的预测题目。

第一题的主要 dataset 是用户听歌记录和歌曲知识库信息，主要格式如下所示，编码用户 id 和歌曲 id。第一种思路是利用歌曲的相似度进行推荐，利用 Jaccard 相似系数，Jaccard 相似系数，矩阵中[i,j]的含义就是用户听过的第 i 首歌曲这些歌曲被哪些人听过，如果两个歌曲很相似，那其受众应当是一致的，交集/并集的比例应该比较大，如果两个歌曲没啥相关性，其值应当就比较小了。 代码中计算了矩阵[66,4879]中每一个位置的值应当是多少，在最后推荐的时候还应当注意一件事对于数据集中每一个待推荐的歌曲都需要跟该用户所有听过的歌曲计算其 Jaccard 值，例如歌曲 j 需要跟用户听过的 66 个歌曲计算其值，最终是否推

荐的得分值还得进行处理，即把这 66 个值加在一起，最终求一个平均值，来代表该歌曲的推荐得分。第二种思路是基于 SVD 分解进行推荐，相似度计算的方法看起来比较简单就是实现出来，但是当数据较大的时候计算的时间消耗实在太大了，对每一个用户都需要多次遍历整个数据集来进行计算。奇异值分解(Singular Value Decomposition，SVD)是矩阵分解中一个经典方法，在 SVD 中我们所需的数据是用户对商品的打分，但是我们现在的数据集中只有用户播放歌曲的情况并没有实际的打分值，所以我们还得自己来定义一下用户对每个歌曲的评分值。如果一个用户喜欢某个歌曲，那应该经常播放这个歌曲，相反如果不喜欢某个歌曲，那播放次数肯定就比较少了。用户对歌曲的打分值，定义为：用户播放该歌曲数量/该用户播放总量。

```
In [5]: triplet_dataset.head(n=10)
```

	user	song	play_count
0	b80344d063b5ccb3212f76538f3d9e43d87dca9e	SOAKIMP12A8C130995	1
1	b80344d063b5ccb3212f76538f3d9e43d87dca9e	SOAPDEY12A81C210A9	1
2	b80344d063b5ccb3212f76538f3d9e43d87dca9e	SOBBMDR12A8C13253B	2
3	b80344d063b5ccb3212f76538f3d9e43d87dca9e	SOBFNSP12AF72A0E22	1
4	b80344d063b5ccb3212f76538f3d9e43d87dca9e	SOBFOVM12A58A7D494	1
5	b80344d063b5ccb3212f76538f3d9e43d87dca9e	SOBNZDC12A6D4FC103	1
6	b80344d063b5ccb3212f76538f3d9e43d87dca9e	SOBSUJE12A6D4F8CF5	2
7	b80344d063b5ccb3212f76538f3d9e43d87dca9e	SOBVFZR12A6D4F8AE3	1
8	b80344d063b5ccb3212f76538f3d9e43d87dca9e	SOBXALG12A8C13C108	1
9	b80344d063b5ccb3212f76538f3d9e43d87dca9e	SOBXHDL12A81C204C0	1

```
In [26]: track_metadata_df_sub.head()
```

	track_id	title	song_id	release	artist_id	artist_mbid	artist_name	duration	artist_far
0	TRMMGCB128E079651D	Get Along (Feat: Pace Won) (Instrumental)	SOHNWIM12A67ADF7D9	Charango	ARU3C671187FB3F71B	067102ea-9519-4622-9077-57ca4164cfbb	Morcheeba	227.47383	0.819087
1	TRMMGTX128F92FB4D9	Viejo	SOECFIW12A8C144546	Caraluna	ARPAAPH1187FB3601B	f69d655c-ffd6-4bee-8c2a-3086b2be2fc6	Bacilos	307.51302	0.595554
2	TRMMGDP128F933E59A	I Say A Little Prayer	SOGWEOB12AB018A4D0	The Legendary Hi Records Albums_ Volume 3: Ful...	ARNNRN31187B9AE7B7	fb7272ba-f130-4f0a-934d-6eeea4c18c9a	Al Green	133.58975	0.779490

基于 Jaccard 相似系数的实验结果如下所示：

```
In [44]: #执行推荐
         is_model.recommend(user_id)

No. of unique songs for the user: 66
no. of unique songs in the training set: 4879
Non zero values in cooccurence_matrix :290327
```

	user_id	song	score	rank
0	a974fc428825ed071281302d6976f59bfa95fe7e	Put Your Head On My Shoulder (Album Version)	0.026334	1
1	a974fc428825ed071281302d6976f59bfa95fe7e	The Strength To Go On	0.025176	2
2	a974fc428825ed071281302d6976f59bfa95fe7e	Come Fly With Me (Album Version)	0.024447	3
3	a974fc428825ed071281302d6976f59bfa95fe7e	Moondance (Album Version)	0.024118	4
4	a974fc428825ed071281302d6976f59bfa95fe7e	Kotov Syndrome	0.023311	5
5	a974fc428825ed071281302d6976f59bfa95fe7e	Use Somebody	0.023104	6
6	a974fc428825ed071281302d6976f59bfa95fe7e	Lucky (Album Version)	0.022930	7
7	a974fc428825ed071281302d6976f59bfa95fe7e	Secrets	0.022889	8
8	a974fc428825ed071281302d6976f59bfa95fe7e	Clocks	0.022562	9
9	a974fc428825ed071281302d6976f59bfa95fe7e	Sway (Album Version)	0.022359	10

基于 SVD 分解的实验结果如下所示：

```
In [55]:
for user in uTest:
    print("当前待推荐用户编号 {}". format(user))
    rank_value = 1
    for i in uTest_recommended_items[user,0:10]:
        song_details = small_set[small_set.so_index_value == i].drop_duplicates('so_index_value')[['title','arti
        print("推荐编号： {} 推荐歌曲： {} 作者： {}".format(rank_value, list(song_details['title'])[0],list(song_d
        rank_value+=1

当前待推荐用户编号 4
推荐编号： 1 推荐歌曲： Fireflies 作者： Charttraxx Karaoke
推荐编号： 2 推荐歌曲： Hey_ Soul Sister 作者： Train
推荐编号： 3 推荐歌曲： OMG 作者： Usher featuring will.i.am
推荐编号： 4 推荐歌曲： Lucky (Album Version) 作者： Jason Mraz & Colbie Caillat
推荐编号： 5 推荐歌曲： Vanilla Twilight 作者： Owl City
推荐编号： 6 推荐歌曲： Crumpshit 作者： Philippe Rochard
推荐编号： 7 推荐歌曲： Billionaire [feat. Bruno Mars] (Explicit Album Version) 作者： Travie McCoy
推荐编号： 8 推荐歌曲： Love Story 作者： Taylor Swift
推荐编号： 9 推荐歌曲： TULENLIEKKI 作者： M.A. Numminen
推荐编号： 10 推荐歌曲： Use Somebody 作者： Kings Of Leon
```

第二个题目是给定一个短视频应用的用户脱敏后的数据，对于用户在下一个周期中可能出现的行为，主要分为播放、关注、点赞、转发、举报和减少此类作品六种行为。部分数据格式如下：

1.注册日志（user_register_log.txt）

列名	类型	说明	示例
user_id	Int	用户唯一标识（脱敏后）	666
register_day	String	日期	01, 02 .. 30
register_type	Int	来源渠道（脱敏后）	0
device type	Int	设备类型（脱敏后）	0

4.行为日志（user_activity_log.txt）

列名	类型	说明	示例
user_id	Int	用户唯一标识（脱敏后）	666
day	String	日期	01, 02 ..30
page	Int	行为发生的页面，每个数字分别对应"关注页"、"个人主页"、"发现页"、"同城页"或"其他页"中的一个	1
video_id	Int	video id（脱敏后）	333
author_id	Int	作者 id（脱敏后）	999
action_type	Int	用户行为类型，每个数字分别对应"播放"、"关注"、"点赞"、"转发"、"举报"和"减少此类作品"中的一个	1

对于这种序列化的数据，我采用基于 RNN 的模型结构去处理序列化数据预测问题，首先将数据进行预处理，然后形成可以被我利用的结构化信息，然后将其输入到 RNN 网络中进行预测，最终得到的结果如下所示：

```
In [131]:
test()

train_loss 0.6144134916860811
test_score: [0.8042493443187537, 0.8034833869239013, 0.8029689608636976, 0.8027878722134838, 0.8030067815998039, 0.8025592650315807]
```

	user_id	prob	label
0	914049	0.551180	1.0
1	833407	0.855232	1.0
2	322751	0.712806	1.0
3	116575	0.311971	1.0
4	611956	0.712806	1.0
5	330729	0.809744	0.0
6	685784	0.671162	1.0

5. 相关在线讲座的参与和笔记

博文视点讲座 4.18：

1.华先胜老师首先介绍了达摩院人工智能中心的愿景和四大研究方向。然后从四个方面讲述了新冠肺炎 AI 技术服务及应用。介绍达摩院在疫情期间利用视觉 AI、大数据、自然语言处理、基因等核心技术快速研发针对新冠病毒感染诊断、基因分析、蛋白分析、疫情预测、跨语言交流及机器自动问答方面的能力，并快速部署为可规模化服务的经验。

2.美国伊利诺伊大学芝加哥分校杰出教授刘兵老师的新冠病毒全基因组相似

性和进化分析。研究人员通过对 377 个 COVID-19 新冠病毒及相关病毒的全基因序列进行了相似性及进化关系的计算分析，得到了一些潜在有趣的结果，可能会对相关领域专家找到病毒的源头、有效的检测试剂、疫苗及治疗药物的研发等有所帮助。收集了 377 个公开发布的 COVID-19 病毒、先前已知的 4 种引起流感的冠状病毒 HCov-229E、HCov-OC43、HCov-NL63 和 HCov-HKU1 以及致命的致病性 P3/P4 病毒：SARS、MERS、Victoria、Lassa、Yamagata、埃博拉和登革热的全基因组序列。

3.清华大学唐杰教授的基于知识的全球新冠疫情风险评估和复工辅助决策系统

AMiner 知识疫图--唐杰

目标：

1.汇聚冠状病毒的各种数据源。

2.基于大数据的智能预测。

3.构建冠状病毒的知识图谱。

项目内容：溯源、疫苗、复工、传播、隔离。

汇集世界上最全面的知识图谱。

高关注度专家分析，学术成果时间线，惠民惠企政策地图，新闻事件分析日报，用户在线社交行为研究，疫情趋势预测，疫情风险指数、智能预测工具。

4.浙江大学陈为教授的疫情大数据可视化，疫情数据可视化提供面向企业业务场景的一站式大数据分析解决方案，基于大数据、移动互联网、人工智能等先进技术，全面支撑企业业务创新，随时随地透视经营，辅助企业科学决策，加速企业数据化转型升级，助力企业进行精准营销、战略管控、风险预警等。

5.基于数据建模的疫情传播分析，对于流行病毒的传播进行准确、有效的预测分析，是疫情科学防控的重要一环。如何分析病毒传播的内在特征、如何结合重要因素，如检疫的有效性和返工返学人潮流动的影响、如何把感染者个体传播行为与大规模疫情扩散传播有机结合都是精准预测的关键因素。

肖仰华老师在线知识图谱培训 4.24-4.26

传统知识工程的缺陷：处理异常的能力不足，任何人工智能的应用都应该以辅助专家为目的，而不是以取代专家为目的。

大数据时代的机遇：众包机制，高质量 UGC。

知识图谱的研究意义：人类已经进入智能时代，大数据的日益积累、计算能力的快速增长为人类进入智能时代奠定了基础，大数据为人工智能的发展提供了数据红利。

各行业智能化升级与转型，认知智能是人工智能的核心，知识图谱使能认知智能。

知识理解数据的本质：建立从数据到知识库中实体、概念、关系的映射。

机器解释现象的本质：利用知识库中的实体、概念、关系解释现象的过程。

机器语言的理解需要背景知识。

知识引导将成为解决问题的主要方式。

还观看了斯坦福 CS520 知识图谱课程。

2.5 月工作的计划

（1）首先完成论文修改计划，做好下一个会议的转投准备。

（2）网站构建遇到了一些知识瓶颈，需要花费一些时间和精力来完整相关前端知识的学习与掌握，后端的 SPARQL 端口已经测试成功，可以完成相应的 SPARQL 查询工作，下阶段的主要工作是去给本地知识库构建一个前端架构，完成对数据的包装和展示。
（3）继续对论文进行阅读和代码实践，养成日常习惯。
（4）和超宇讨论数据集的工作，寻找一些工作点，辅助超宇的想法做一些实验，看看是否能在 25 号之前形成一篇数据集文章投出去。