## 项目的最终目标

综合性的开源文本处理平台，采用GPU加速，其中包含若干文本处理算法，这种软件包目前还没有。

这个软件包包含四个方面，

1. 根据一个很大的文档集计算单词之间的相似度。从若干G的文本中发掘词与词之间的相似度从而可以得到某些单词的近义词或者同义词；比如文档集合是5GB的学术论文，那么我计算的相似度就是在学术方面单词之间的相似度。
2. 第二个方面是对较大的文本数据进行聚类，得到聚类的结果；聚类是很多其它算法的基础，用处不用多说。
3. 第三个方面是实现重复文档的检测，例如给定一个文档集合（如100w篇毕业论文）计算其中的重复文档，文档不必完全相同，只要几乎相同（相似程度的百分比很高）的就可以检测出来；
4. 给定一个文档集合以及一个查询集，能够比较快速的对查询集中的文档进行knn查找。

项目最后以开源软件的形式提供下载，用户可以使用我们的软件根据相应的文档在自己的机器上面跑相应的程序并得到结果；另外我们还准备做一个简单的网站并且提供一些demo，用户只要提供数据就可以在我们的服务器上跑出结果并且将结果返回给用户。

## 项目的阶段性目标

Duplicate detection和knn已经找到了相应的CPU或者GPU论文，现在暂时放下了。

1. 放假之前（8月16日）作出一个k-means的文本聚类，CPU和GPU两个实现版本；
2. 回来之后开始对程序进行调优并且开始做文本流的聚类，10月之前将文本流在GPU上面的聚类做出来；
3. 接下来做文档重复检测，期望实现时间一个月；
4. 最后做knn，期望实现时间一个月。

## 调研现状：

1. **聚类和分类的深入调研**
   1. 文本聚类的主要方法

* K-means
* Bisecting k-means
* BIRCH算法（层次聚类）

调研结论：上述这些方法都是比较经典的算法，其中BIRCH算法依赖比较复杂的数据结构因此不太适合在GPU上面实现，k-means算法比较简单高效、易于并行化，因此决定使用k-means算法作为基础算法，借助之前调研的STREAM数据流聚类的权重模型来修改k-means算法，希望能够处理文本流的聚类问题。

* 1. 文本分类的调研（如果有时间再考虑做）
* svm

调研结论：在分类效果上面，svm是最佳的，而使用GPU对svm进行加速也由Berkeley的研究人员做过相应的研究，期间也分析过一份基于GPU的svm代码，但最后发现效果并不是很好，这方面还没有决定怎样实现甚至实现与否。

1. **文档重复检测**

由于document duplicate detection并没有像文档的knn以及clustering那样主流，因此使用GPU加速这方面的研究很少，我们找到了一篇使用fuzzy hashing来进行Document duplicate detection的论文，初步决定在此基础之上使用GPU进行加速，也就是做一个基于GPU的fuzzy hashing。

1. **KNN的调研**

* 低维数据点的knn与高维、文本数据的knn的差别
* 基于LSH的knn

调研结论：对于低维数据使用kd树进行knn查找能取得较好的效果，但是kd树在高维数据上的性能会变得非常差，甚至不如brute-force的性能；对于高维的数据一般都是使用LSH进行近似的knn查找，利用GPU加速基于LSH的高维数据的knn查找已经由UNC（北卡罗来纳大学）的研究人员研究过并且给出了源代码，可以从其网站上下载，但是他们并不是基于文本的；我们的目标是使用GPU加速基于LSH的文本的knn算法，初步决定借鉴UNC的代码并且实现文本的knn。

1. **流数据处理的深入调研**
   1. 文本流处理的现状：文本流聚类

调研结果：现在对于文本流处理主要集中在文本聚类上面，有一些关于文本流分类的研究，其他的流处理形式更少

* 1. 文本流聚类的两种类型

调研结果：对于文本流聚类有大致有两种处理模型，一类称为单遍扫描的数据流聚类，这类算法计算整个数据流中界标或者滑动窗口中的数据的聚类结果； 另一类则侧重于数据流中簇进化分析，这类算法认为数据流中的簇是不断演化的，通过对数据流中的数据进行聚类处理，获取数据流中不断进化的簇信息。前者的结果一般是滑动窗口中数据点的聚类结果，后者的结果一般是数据流中对于聚类的某种描述（micro-cluster），前者以聚类算法STREAM（我们已经研究过其详细代码）为代表，而后者以CluStream为代表，各有不同。

* 1. 文本流处理与GPU的结合

调研结果：现在将文本流处理与GPU结合起来的研究很少，现在找到的唯一做过类似工作的是复旦大学的一个博士于2004年做的，由于时间较早所以并不是基于现行的CUDA而是使用的处理图形的OpenGL，要做很多的跟图形方面相关的研究才能做。

1. **LSH的调研**

寿老师最近审到一篇VLDB的文章，是使用基于集群的LSH来处理上十亿篇的tweets并且取得了比较好的效果，因此在LSH方面想让我们再深挖一下，看能不能将LSH与聚类也联系起来。

## 调研后项目方向与创新点

由于文档的重复检测与单词相似度计算没有在GPU上面实现过，因此这算是一个创新点； 而对于文本聚类和相似文本的knn查询，现在将流处理与GPU结合起来的研究还很少，目前只查到一篇复旦大学发的论文，但是比较早，当时GPU还没有用来做通用计算，因此他们是基于OpenGL并不是基于CUDA的。

我觉得这个项目并不是纯粹意义上的科研项目，而是要做一个文本处理方面的开源软件包，其中绝大部分还是工程性质的东西。综合起来，项目的创新点一共有两个：

1. 使用GPU对文档的重复检测以及单词相似度的计算进行加速；
2. 使用GPU对大量文本流的处理（包括聚类和knn查询）加速，一起取得更短的响应时间。

为什么做的人比较少？

太难？太简单？正在做还没有发表？

一方面是因为GPU的通用计算刚刚起步，研究GPU通用计算的人越来越多，但是大多都集中的非流式的处理上面；二是因为流式计算涉及面并不是很宽，因此研究的人也相对比较有限，因此将两者结合起来的研究现在还很少。

## 项目的意义

经过一段时间的调研，发现用GPU加速文本处理的文章并不多（可能很多的大数据处理都交给了集群去做，mapreduce之类的）。我觉得我们的项目可以试试这方面的可行性，在单机上面提高文本处理的效率。做出一个文本处理的框架，对于实验室今后在文本处理方面的持续研究还是有帮助的。