哈 尔 滨 工 业 大 学

国 家 示 范 性 软 件 学 院

本科毕业设计(论文)开题报告

题 目： 基于知识库的海量异构数据集成系统的设计与实现

专　　业	物联网工程
学 生 姓 名	李天宝
学　　号	1133730206
联 系 方 式	15704600640
年　　级	2013 级
实 习 基 地	上海骇咕赛信息科技有限公司
基地指导教师	丁盛豪
联 系 方 式	13761793694
校内指导教师	王宏志
联 系 方 式	13069887146
开 题 日 期	2016.11.19

哈尔滨工业大学软件学院

目　录

1. 项目来源及开发目的和意义

1.1 项目来源

本项目来源于上海骇咕赛信息科技有限公司大数据平台数据集成子系统的最新需求。公司长期专注于企业级高并发高可用性技术解决方案和数据分析、数据挖掘领域，主要产品有高并发高可用性分布式缓存集群系统，该项目设计为大型计算集群、为分布式计算服务提供热缓存数据交换。在此系统之上，针对于海量数据整合、处理、分析、学习的需求，公司力主架构于该分布式系统，实现简便、高效的海量数据加工的功能，为该系统设计海量数据快速处理的 API。作为数据处理平台，数据集成便是其中不可缺少的一项，本人的项目即基于此内容进行一些创新性的研发。

1.2 项目开发目的和意义

数据集成涉及将不同来源中的数据整合，并统一地向用户展示这些。数据集成在很多领域上非常重要，包括商业（当两个相似的公司需要合并它们的数据库）、科学（组合来自不同生物信息学数据库的研究结果）。 随着数据量和数据分享需求爆炸式的增长，数据集成出现的频率越来越高。它已成为大范围理论工作的焦点，然而许多问题仍然没有解决。[1]

上文中，维基百科的介绍已经暗示了数据集成的内容与重要性。简言之，数据集成是一个将具有不同概念、上下文、逻辑关系的数据文本进行合并，形成一个具有统一模式的数据集。作为数据分析和运用的基础，计算机领域内数据集成具有重要的意义，包括数据清洗、模式识别、生物信息等等。然而，随着大数据时代的到来，数据每天都在被生成、分析并且大量的应用着，数据驱动的决策也成为了社会中不可或缺的一部分。现如今，互联网上分散存在着海量的数据，为了充分利用这些数据中的信息以及蕴含的价值，将这些数据有效的集成在一起成为了一个显著的需求。

大数据时代下的数据集成，与传统数据集成的考核点（数量、速度、多样性、真实性）有着一定程度上的不同。首先，单一数据源无法为单一目标

[1] https://en.wikipedia.org/wiki/Data_integration

储存如此大量的数据。其次，很多数据源是动态的，总会有很多数据源在生成、消失，或者某些数据源改变了储存方式。第三，在诸多数据源之间，我们很难保证他们是完全统一的，这就造成了异构数据源的产生，并且相同条目下的数据也可能重复或者存在精度、修改时间等方面的差异。

为了将海量的异构、异源数据进行集成，首先需要做的就是将不同数据库的模式进行集成，生成一个全局的数据库模式，进而方便数据库记录的填充和数据库的融合。对于数据集成问题，传统的方法往往是预先指定一个全局的数据库模式，然而针对于海量数据的背景，人们难以在大量的数据中捕捉全局的信息来设计预定的全局模式，并且建立全局模式和每一个数据库模式之间的匹配关系也是耗时耗力的。

因此，在上述背景下，通过设计一些合理匹配关系和高效的算法，省时、准确的生成一个全局数据库模式成为了迫切的需求。

2. 国内外相关领域开发及应用现状分析

自从人们开始整合有关联的数据、从中获取有用的信息开始，数据集成便走上了历史的舞台。早在计算机科学家研究这个领域之前，统计学家已经做了很多的工作，试图分析获取的数据集之间的关联。然而数据集成在很多方面都是具有挑战性的，为了解决这个问题，在过去的数十年内许多计算机科学家提出了诸如匹配连接、数据融合、数据结构化等基础性议题及可行的解决方案。近年来，无论是科学、通讯、多媒体，还是经济、医疗等方面，人们开始能够获取现实生活过程中每一件事、每一个事物的电子数据。因此，我们急切需要能够在大数据中分析、提取有效信息的能力，将现有的生活向数据驱动转变。然而最根基的一步便是数据集成。

数据集成的过程大体可以分为以下三个步骤：

图片 1 数据集成流程

从图中我们可以看出，数据集成的基础工作就是模式集成。模式集成的目标是将不同数据库中相同或者相似的属性合并成一个属性。相同或者相似这一概念对于人们来说是可以理解的，然而对于计算机来说这个是很模糊的概念。究竟什么样的属性可以判定成相同或者相似，这是实现数据集成一个关键的问题。相关研究者大体在以下两方面进行了研究。

首先，我们需要将形式上相近的属性归为一类。在数据库的录入过程中，由于拼写错误导致的数据库质量不高的问题是很常见的。而这一问题对于数据集成来说，也是一个很显然的隐患。如何将拼写错误的属性和原属性判定为相同是一个很关键的，这里经常使用的方式是计算词间距离。我们按照一定方式计算两个属性词之间的距离，该距离在一定阈值之下，我们就可以认定两个属性是相同的。常用的方式是编辑距离[1]，以一个词转变为另一个词所需的增、删、改的操作数作为两者的距离。基于此，相关的 join 算法已

经被广泛提及[2]，基于类似的算法，我们可以对多个属性集进行 join 操作来实现模式的集成。

除此之外，为了让计算机能够判定人类思维中相似的概念，近年来很多科学家致力于挖掘互联网上的信息并提取相关实体、关系，根据人们的思路去建立知识库，来尽可能地去表达人们对事物的认知。现有的相关工作包括 Freebase[3]，Google Knowledge Graph[4]，ProBase[5]，Yago[6]等等，这些知识库可以用来辅助我们的数据集成，使得数据集成过程能够更好地模拟人工集成的方式。如果能根据这些知识来实现模式之间的 join 算法，进而来实现模式集成和数据集成，我们可以预见这一结果将是符合人们预期的。而在此角度上，由于知识库是一个较新的概念，并且知识库还不够完善，相关工作做得并不多。

基于此，我们希望能够结合形式和语义两个方面，来完成一套模式集成的系统。

3. 需求分析及总体设计方案

数据集成是一个十分重要但却复杂的过程，而模式集成是其中最基础也是格外具有挑战的一个步骤。本节中将详细分析模式集成所需要进行的工作，并给出预期的设计方案。

3.1 主要开发内容

模式集成的主要工作是针对异源、异构数据表的模式，在形式和语义的两个维度将它们进行集成，从而得到一个统一的模式，既能将多个数据源中的所有属性全部包含，又能保证产生的数据模式中属性彼此不重复，满足数据库的范式要求。我们在此以航班数据库[7]为例，对主要工作进行分析。我们拥有 5 个数据库的数据，并在下表中展示出它们的数据库模式。

表格 1 航班数据库示例

数据库	表名	模式
Airline1	Schedule	Flight Id, Flight Number, Start Date, End Date, Departure Time, Departure Airport, Arrival Time, Arrival Airport
	Flight	Flight Id, Departure Date, Departure Time, Departure Gate, Arrival Date, Arrival Time, Arrival Gate, Plane Id
Airline2	Flight	Flight Number, Departure Airport, Scheduled Departure Date, Scheduled Departure Time, Actual Departure Time, Arrival Airport, Scheduled Arrival Date, Scheduled Arrival Time, Actual Arrival Time
Airport3	Departures	Air Line, Flight Number, Scheduled, Actual, Gate Time, Takeoff Time, Terminal, Gate, Runway
	Arrivals	Air Line, Flight Number, Scheduled, Actual, Gate Time, Landing Time, Terminal, Gate, Runway
Airfare4	Flight	Flight_Id, Flight Number, Departure Airport, Departure Date, Departure Time, Arrival Airport, Arrival Time
	Fares	Flight_Id, Fare Class, Fare

Airinfo5	AirportCodes	Airport Code, Airport Name
	AirlineCodes	Air Line Code, Air Line Name

在表中我们可以发现，这五个数据库之间是相互独立的，但不同的数据表和属性之间是相等或者是重叠的。在这个例子中，模式集成的目标就是将多个数据源模式归并到一张数据表中。在这个过程中，应将含义类似（如 Start Date，Takeoff Time 和 Scheduled Departure Date）、形式类似（如 Flight_Id 和 Flight Id）等情况的属性合并。很显然，随着数据表数量的增大和单一数据表内属性个数的增多，逐一比较每两个属性是不合适的，我们通过实现类似于数据库范畴内的 join 操作，在形式和语义两个角度上完成集成。

3.2 需求分析

对于一个模式集成平台，它应能接收标准的数据库模式，并尽可能的在多个维度上完成模式集成的功能，保证前文提到的几种情况的类似属性都能够检测出并合理的整合。除了这些基本功能外，为了保证这个系统的可用性和展示效果，应设计友好的用户界面来指导完成模式集成的工作，使这个抽象的操作更容易的进行。

3.2.1 系统功能需求

本系统主要包括预处理、形式整合、语义整合、生成全局属性、前端界面展示的 5 个模块。本节中将以模块为单位详细介绍每个模块的功能需求。

预处理模块主要功能点如表 2 所示。

表格 2 预处理模块需求分析表

功能点	需求描述
模式预处理	1. 将输入的属性进行预处理，转化成词组的形式，包括以下几种情况： ● 原词满足要求不需要改动 ● 存在缩写，根据模式说明提供的缩写规则，或是平常普遍接受的规则进行展开 ● 去除词之间的分割符号，以空格代替 ● 所有字母转换成小写 2. 为每个词组选取一个关键词代替整个词组，进而与知识库中的实体进行匹配时采用这个关键词，这里采取 tf-idf 算法

知识库预处理	1. 将现有的知识库进行预处理，提取实体之间的相关关系，并储存在硬盘 2. 采用合适的方式进行索引，加快磁盘中知识的访问速率，可采用的方式有： ● hash 表 ● B 树

形式整合模块主要功能点如表 3 所示。

表格 3 形式整合模块需求分析表

功能点	需求描述
生成倒排表	1. 用户给定形式上差异允许的阈值 ε_t 2. 将用户给定的词拆分成长度为 q 的 gram，并依据 q 和 ε_t 计算判定相似时所应具有相同 gram 的个数 3. 生成倒排表： ● 为属性集合中的元素生成倒排表 ● 为知识库中提取出的每一个概念生成倒排表
寻找形式相近属性	1. 对于给定属性，通过倒排表筛选出形式上相近的属性 2. 将相似的所有属性构成一个集合

语义整合模块主要功能点如表 4 所示。

表格 4 语义整合模块需求分析表

功能点	需求描述
寻找语义相近属性	1. 用户给定语义上差异允许的阈值 γ 2. 对给定的属性在知识库范围上进行 γ 次 join 操作
后处理形式相近属性集合	1. 将判定为相似的属性按组构成集合 2. 在每组内对结果进行去重操作 3. 试图判定每个组内属性之间的语义联系，保证每个组内的属性在语义近似（γ 范围内）是一个闭包

生成全局属性模块主要功能点如表 5 所示。

表格 5 生成全局属性模块需求分析表

功能点	需求描述
整合形式近似和语义近似两个模块	1. 获取形式近似和语义近似两个模块的输出结果 2. 在两类结果中匹配相同属性所在的集合，并将包含同一属性的集合进行融合 3. 验证并保证新生成集合是一个形式、语义近似的闭包

生成全局模式	1. 在每个集合内，选出一个主属性，作为组内相似属性（次属性）的代表 2. 将组内其他次属性匹配至该主属性 3. 将所有输入的属性取并，并删除所有次属性 4. 建立新的全局属性到原有各个表中每个属性的关联

前端界面模块主要功能点如表 5 所示。

表格 6 前端界面需求分析表

功能点	需求描述
输入	1. 指定输入的模式集合 2. 指定所使用的知识库
输出	1. 显示生成的全局模式 2. 显示输入模式中，每一属性与全局模式中属性的对应关系

3.2.2 系统非功能需求

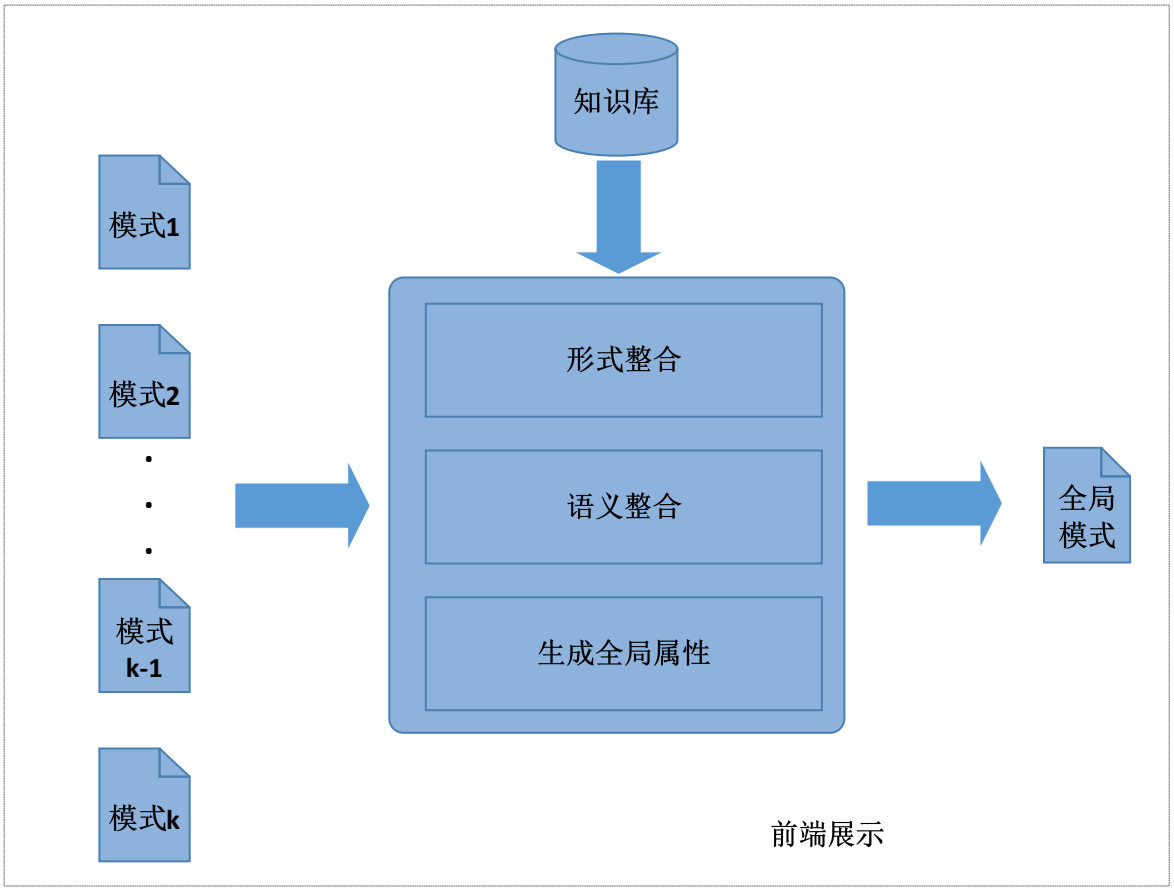
本节中主要包含系统的非功能性需求，包括空间、时间限制、数据完整性等方面的需求，其具体内容如下。

1) 空间需求：本系统所储存的数据量巨大，知识库的实体数目应在百万条以上，数据库属性的个数随使用者需求而定，一般不低于一万条。程序运行过程中不应将数据成批次的读入至内存中，整个系统和相关算法应该是基于外存的。
2) 时间需求：系统运行时间应随输入模式大小而变。如遇大批量的数据进行处理时，可根据先行测试得知某一数据量处理所需大概时间，并对用户予以提示。
3) 完整性：程序运行的结果应储存在磁盘内。程序运行的过程中，应保证输入数据、所用知识库、预处理数据不被更改。
4) 扩展性：系统应可根据后续需要进行扩展，例如增加知识库处理方式、近似度判定方式等等。
5) 健壮性：系统在运行过程中，应对可能出现的情况进行预测并对差错进行防范，不会因为数据量的提升导致系统崩溃或是内存泄漏。同时前端界面应尽可能的简洁易懂，同时对用户的误操作有容忍度。

3.3 总体设计方案

3.3.1 系统功能结构图

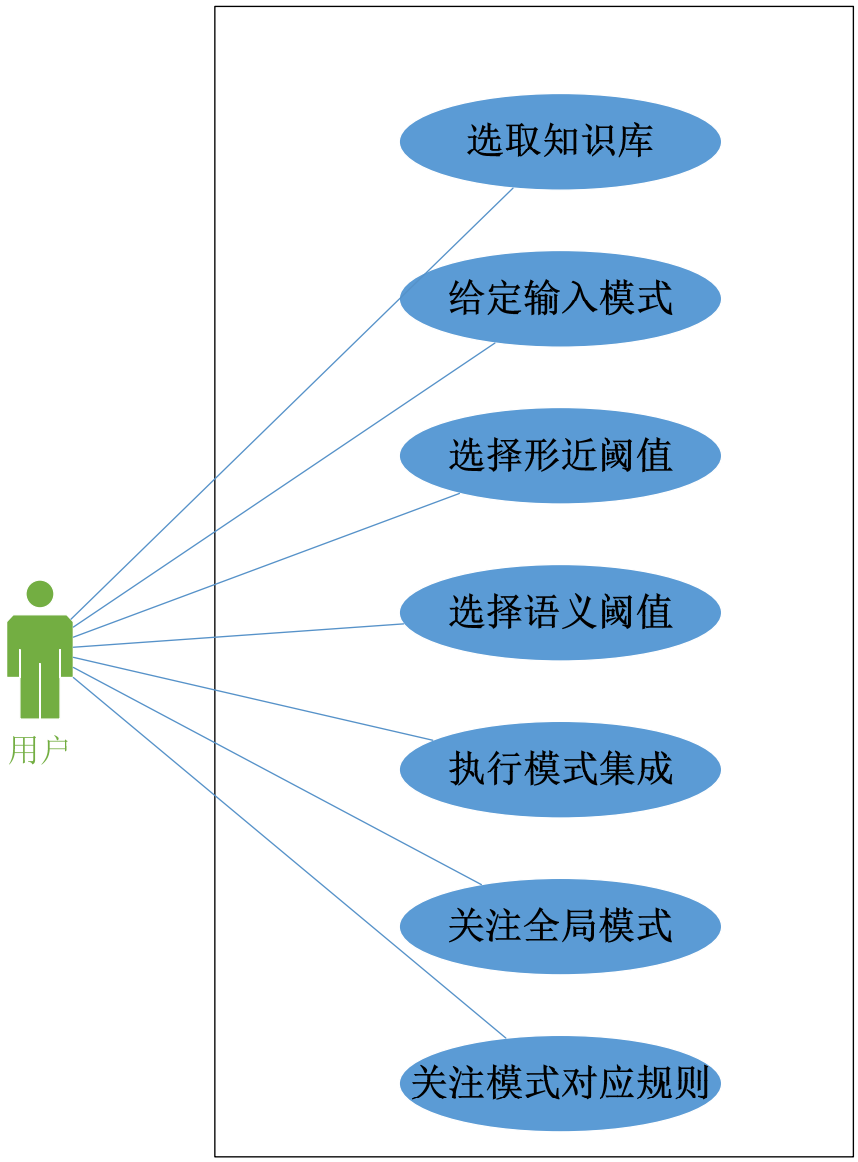
整个系统以多个数据库的模式为输入，以现有知识库为依托，经过形式整和语义整合，生成全局属性。系统通过前端界面对以上过程的结果及必要的中间输出进行体现。该系统的功能结构图如图 2 所示。



图片 2 系统功能结构图

3.3.2 系统主要流程用例图

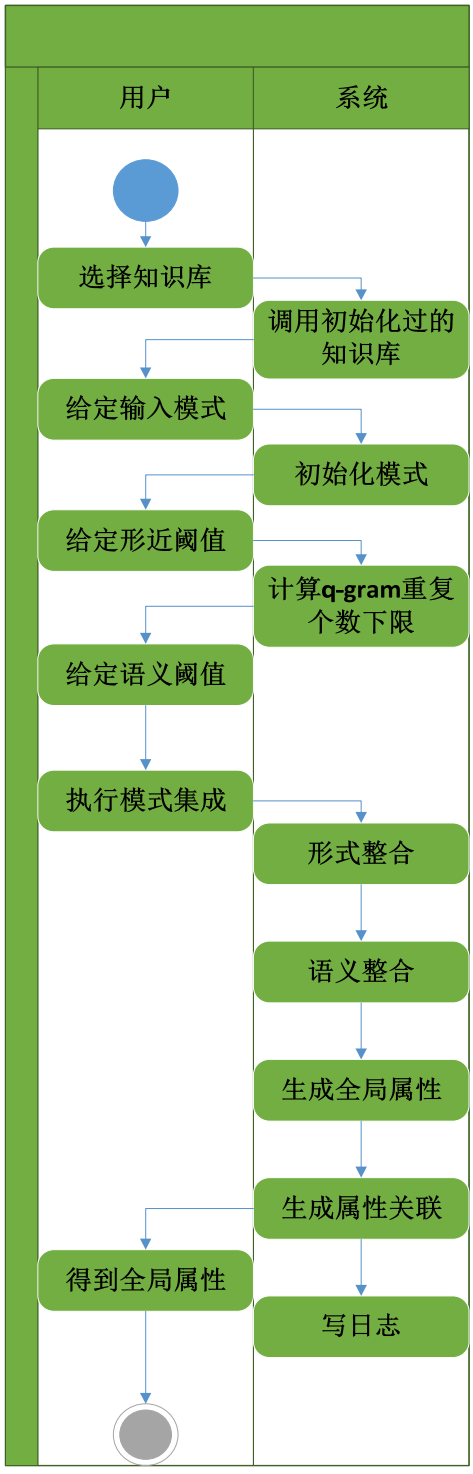
图 3 所示的是用户通过该系统执行模式继承的用例图，包含了 3.1 节中所需用户与系统进行交互的全部操作。



图片 3 系统用例图

3.3.2 系统主要流程活动图

图 4 展示了执行整个模式集成过程中，用户与系统之间交互活动的过程。


用户
系统
选择知识库
调用初始化过的知识库
给定输入模式
初始化模式
给定形近阈值
计算q-gram重复个数下限
给定语义阈值
执行模式集成
形式整合
语义整合
生成全局属性
生成属性关联
得到全局属性
写日志

图片 4 系统活动图

4. 开发环境和开发工具

4.1 开发语言

本系统采用 C++语言开发并实现算法，前端展示采用 web 相关技术。

4.2 开发工具

如表 7 所示，本系统的开发使用到了这些开发工具。

表格 7 开发工具表

工具类别	工具名称	作用
集成开发环境	Visual Studio 2013	程序最主要的开发、调试平台
脚本开发工具	Atom	数据预处理、后处理脚本开发平台
界面设计工具	Photoshop CC2015	图片素材设计
前端开发工具	PyCharm	前端界面的实现
版本控制软件	Github	版本控制

4.3 开发环境

操作系统：Windows10

处理器：Intel Core i7，2.40GHz 主频或更高

内存：8GB 1333 MHz DDR3

程序运行环境：Windows 7 或更高版本的 Windows 操作系统

注：以上开发环境均为前期开发要求，当集成到公司系统时，应使用更高性能的服务器，相关运行环境可能发生变化。

5. 项目进度安排、预期达到的目标

5.1 进度安排

项目进度及毕业设计（论文）工作安排见表 8。

表格 8 项目进度及毕业设计（论文）工作计划表

起始时间	完成时间	计划工作内容	备注
2016.07.18	2016.07.31	配置公司要求的环境，熟悉集群	已完成
2016.08.01	2016.08.28	学习所需基础知识	已完成
2016.08.29	2016.09.11	分析现有系统，寻找创新点	已完成
2016.09.12	2016.09.18	选择研究方向	已完成
2016.09.19	2016.10.09	验证选题的可行性	已完成
2016.10.10	2016.11.06	研读相关工作现有论文，进行比较	已完成
2016.11.07	2016.11.20	撰写开题报告，准备开题答辩	在进行
2016.11.21	2016.12.04	寻找知识库，根据需要进行处理	未完成
2016.12.05	2017.01.01	完成形近整合部分算法的设计	未完成
2017.01.02	2017.02.05	完成语义整合部分算法的设计	未完成
2017.02.06	2017.03.05	使用少量数据进行试验，验证准确度	未完成
2017.03.06	2017.03.18	准备中期答辩	未完成
2017.03.19	2017.04.01	用大量数据进行试验，验证效率	未完成
2017.04.02	2017.04.22	根据实验结果调整系统	未完成
2017.04.23	2017.05.13	设计、实现前端界面	未完成
2017.05.14	2017.05.27	进行最后的衍生，完善系统	未完成
2017.05.28	2017.06.20	撰写论文，参加毕业答辩	未完成

5.2 预期达到的目标

项目完成后，应达到以下目标：

1) 前端界面不应出现任何 bug，无明显卡顿
2) 能够正常载入并预处理输入模式
3) 能够正常载入并读取知识库
4) 可以根据形近和语义规则找出相近的属性，并进行聚集
5) 对于近似属性，能够保证较低的误判率
6) 能够快速响应并向用户给出反馈；对于数据量很大的输入给与友好的提示，不应让用户无谓的等待

6. 完成项目所需的条件和经费

6.1 已具备的条件

1) 开发用个人笔记本电脑（Windows10 系统）
2) C++（Visual studio 2013），Python，Web 开发环境
3) 较熟练掌握 C++语言进行程序的主体开发和算法实现
4) 基本掌握 python 语言进行数据的预处理、后处理
5) 较熟练掌握相关的数据结构与算法的知识
6) 对于数据库算法、外存算法有着一定的了解
7) 参与过一些研究工作，对于创新性研究有一点经验
8) 较熟练使用 GitHub 进行版本控制
9) 有较好的英文水平，阅读相关英文论文不存在障碍

6.2 需要的条件和经费

1) 较全面的包含同义、近似、相关关系的知识库
2) 高效处理现有数据（数据表和知识库）的服务器
3) 数据库算法、外存算法的相关文献

本项目所有书籍经费以及设备均由本人和公司承担。

7. 预见的困难及应对措施

本项目开发过程中，可以预见的困难及应对措施如下：

1) 不熟悉骇咕赛公司的平台及相关开发流程

解决办法：研读公司相关的手册及文档，大体了解工作相关部分的架构与工作流程，咨询 mentor

2) 现有知识库不易表达相似这一概念

解决办法：一方面上网去寻找合适的知识库，另一方面针对现有的知识库设计转换规则，从中提取出我们所希望得到的关系

3) 数据表中的属性命名不规范

解决办法：寻找可靠的数据源进行测试，对于期望处理的数据表索取描述文档；其次学习并实现相关 NLP 技术对自言语言描述的属性名进行解析，提取关键信息

4) 所需预处理数据量过大

解决办法：使用公司的服务器进行处理，将相关的数据储存在集群中，使用时通过内网访问

5) 相关科研创新性工作有难度

解决办法：及时与校内外导师取得联系，寻求指导

参考文献

[1] Levenshtein V I. Binary codes capable of correcting deletions, insertions and reversals[C]//Soviet physics doklady. 1966, 10: 707.

[2] L. Li, H. Wang, J. Li, and H. Gao. Ed-sjoin; an optimal algorithm for similarity joins with edit distance constraints [j]. Journal of Computer Research and Development, 46:319-325, 2009

[3] Bollacker K, Evans C, Paritosh P, et al. Freebase: a collaboratively created graph database for structuring human knowledge[C]//Proceedings of the 2008 ACM SIGMOD international conference on Management of data. ACM, 2008: 1247-1250.

[4] ACM, 2014: 601-610. Dong, Xin, et al. "Knowledge vault: A web-scale approach to probabilistic knowledge fusion."

[5] Wu W, Li H, Wang H, et al. Probase: a probabilistic taxonomy for text understanding[C]//Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data. ACM, 2012: 481-492.

[6] Weikum G, Theobald M. From information to knowledge: harvesting entities and relationships from web sources[C]//Proceedings of the twenty-ninth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems. ACM, 2010: 65-76.

[7] Dong X L, Srivastava D. Big data integration[J]. Synthesis Lectures on Data Management, 2015, 7(1): 1-198.

[8] A. Arasu, S. Chaudhuri, and R. Kaushik. Learning string transformations from examples. Proceedings of the VLDB Endowment, 2(1):514{525, 2009.

[9] R. Cilibrasi and P. Vitanyi. Automatic meaning discovery using google. In Dagstuhl Seminar Proceedings. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2006.

[10] U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth. From data mining to knowledge discovery in databases. AI magazine, 17(3):37, 1996.

[11] L. Gravano, P. G. Ipeirotis, H. V. Jagadish, N. Koudas, S. Muthukrishnan, L. Pietarinen, and D. Srivastava. Using q-grams in a dbms for approximate string processing. IEEE Data Eng. Bull., 24(4):28{34, 2001.

[12] M. L. Lee, T. W. Ling, and W. L. Low. Intelliclean: a knowledge-based intelligent data cleaner. In Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining, pages 290{294. ACM, 2000.

[13] X.-M. Lin and W. Wang. Set and string similarity queries: A survey. Jisuanji Xuebao(Chinese Journal of Computers), 34(10):1853{1862, 2011.

[14] R. Mihalcea and P. Tarau. Textrank: Bringing order into texts. Association for Computational Linguistics, 2004.

[15] G. Salton, E. A. Fox, and H. Wu. Extended Boolean information retrieval. Communications of the ACM, 26(11):1022{1036, 1983.

[16] K. Sparck Jones. A statistical interpretation of term speci_city and its application in retrieval. Journal of documentation, 28(1):11{21, 1972.

[17] H. Wang, J. Li, and H. Gao. E_cient entity resolution based on subgraph cohesion. Knowledge and Information Systems, 46(2):285{314, 2016.

[18] C. Xiao, W. Wang, and X. Lin. Ed-join: an e_cient algorithm for similarity joins with edit distance constraints. Proceedings of the VLDB Endowment, 1(1):933{944, 2008.

附件 1：哈尔滨工业大学毕业设计（论文）任务书

姓 名：李天宝	院（系）：软件学院
专 业：物联网工程	学 号：1133730206
任务起止日期： 2016 年 07 月 28 日（下基地开始）至 2017 年 05 月 20 日	

毕业设计（论文）题目：

基于知识库的海量异构数据集成系统的设计与实现

立题的目的和意义：

数据集成是一个将具有不同概念、上下文、逻辑关系的数据文本进行合并，形成一个具有统一模式数据集的过程。作为数据收集和分析的基础，计算机领域内数据集成在诸多领域具有重要的意义，包括数据清洗、模式识别、生物信息等等。现如今，互联网上分散存在着海量的数据，为了充分利用这些数据中的信息以及蕴含的价值，将这些数据有效的集成在一起成为了一个显著的需求。为了将海量的异构、异源数据进行集成，首先需要做的就是将不同数据库的模式进行集成，生成一个全局的数据库模式，进而方便数据库记录的填充和数据库的合并。对于数据库集成，传统的方法往往是预先指定一个全局的数据库模式，然而针对于海量数据的背景，人们难以在大量的数据中捕捉全局的信息来得到预定的全局模式，并且建立全局模式和每一个数据库模式之间的匹配关系也是耗时耗力的。因此，在上述背景下，通过设计一些合理匹配关系和高效的算法，省时准确的生成一个全局数据库模式成为了迫切需求。

技术要求和主要内容：

1) 较熟练掌握 C++语言进行程序的主体开发和算法实现
2) 基本掌握 python 语言进行数据的预处理、后处理
3) 较熟练掌握相关的数据结构与算法的知识
4) 对于数据库算法、外存算法有着一定的了解
5) 对于创新性研究有思路，能根据现有研究工作提出改进
6) 较熟练使用 GitHub 进行版本控制
7) 有较好的英文水平，阅读相关英文论文不存在障碍
8) 能够理解并运用现有的包含同义、近似、相关关系的知识库

进度安排：

2016 年 07 月 18 日—2016 年 07 月 31 日 配置公司要求的环境
2016 年 08 月 01 日—2016 年 08 月 28 日 学习所需基础知识
2016 年 08 月 29 日—2016 年 09 月 11 日 分析现有系统，寻找创新点
2016 年 09 月 12 日—2016 年 09 月 18 日 选择研究方向
2016 年 09 月 19 日—2016 年 10 月 09 日 验证选题的可行性
2016 年 10 月 10 日—2016 年 11 月 06 日 比较相关工作现有论文
2016 年 11 月 07 日—2016 年 11 月 20 日 撰写开题报告，准备开题答辩
2016 年 11 月 21 日—2016 年 12 月 04 日 寻找知识库并进行处理
2016 年 12 月 05 日—2017 年 01 月 01 日 完成形近整合部分算法的设计
2017 年 01 月 02 日—2017 年 02 月 05 日 完成语义整合部分算法的设计
2017 年 02 月 06 日—2017 年 03 月 05 日 少量数据验证准确度
2017 年 03 月 06 日—2017 年 03 月 18 日 准备中期答辩
2017 年 03 月 19 日—2017 年 04 月 01 日 大量数据验证效率
2017 年 04 月 02 日—2017 年 04 月 22 日 撰写研究论文，向会议投稿
2017 年 04 月 23 日—2017 年 05 月 13 日 设计、实现前端界面
2017 年 05 月 14 日—2017 年 05 月 27 日 进行最后的衍生，完善系统
2017 年 05 月 28 日—2017 年 06 月 20 日 撰写论文，参加毕业答辩

同组设计者及分工：

无

指导教师签字_____

年　　月　　日

教研室主任意见：

同　意

教研室主任签字_____

年　　月　　日

附件 2：第 1 次开题检查记录表

<table>
<tr><td>学生姓名</td><td>李天宝</td><td>学　　号</td><td>1133730206</td></tr>
<tr><td>专　　业</td><td>物联网工程</td><td>开题日期</td><td>2016.07.19</td></tr>
<tr><td>指导教师</td><td>王宏志</td><td>职　　称</td><td>教授</td></tr>
<tr><td>毕设/论文题目</td><td colspan="3">基于知识库的海量异构数据集成系统的设计与实现</td></tr>
<tr><td colspan="4">基地指导教师意见（需写具体内容）

签　　字：
年　　月　　日</td></tr>
<tr><td colspan="4">校内指导教师评语：
毕业设计（论文）选题：☐新颖　☐恰当　☐有一定的实用意义　☐不合适；
项目难度：☐高　☐较高　☐适中　☐可以达到要求；
项目预计工作量：☐充分　☐较大　☐适中　☐可以达到要求；
开题报告撰写：内容☐充实　☐较充实　☐一般　☐不够充分；
格式☐规范　☐较规范　☐一般　☐不规范。
☐同意 ☐不同意 ☐报告修改后同意 参加开题检查。

签 字：　　日期：</td></tr>
<tr><td colspan="4">答辩记录：</td></tr>
<tr><td colspan="2">开题检查结论：☐通过　☐警告　☐不通过</td><td colspan="2">成绩：☐A　☐B　☐C　☐D</td></tr>
<tr><td colspan="2">答辩组长签字：</td><td colspan="2" rowspan="2">答辩组成员：</td></tr>
<tr><td colspan="2">答辩秘书签字：</td></tr>
</table>

注：若导师不能亲笔签字，则将邮件整个截图缩小后粘贴到相应的格中，同时将原图附到报告后面（截图内容要求见后面）。

【此处粘贴基地导师/校内导师意见的邮件截图】

（如果基地导师/校内导师意见是通过邮件发送的，请将邮件截图贴于此处。注意：截图中一定包括**邮件时间**、**发件人**信息、**收件人**信息、**意见内容**，同时意见中必需包括“**同意/不同意开题**”字样）