

基于依存关系的自然语言可视化仿真系统

袁雨轩¹, 李放¹, 陈科淇², 韩正¹

(1. 大连东软信息学院 数字艺术与设计学院, 辽宁 大连 116023;

2. 华威大学 华威商学院, 考文垂 CV4 7AL)

摘要:目前的虚拟仿真系统主要围绕物理、化学实验和数学公式展开,对文学作品的仿真案例较少。随着设备计算能力的进步,利用深度学习模型处理中文自然语言输入是目前的研究热点。由于中文是一种意和型的、不执著于文法规范的语言,自然语言处理技术对中文的处理相比英文要复杂许多,这一现状极大限制了该技术在应用层面的发展。而在实际应用上,对自然语言文本中知识的可视化可以极大丰富内容的精彩程度。对此,针对中文的词法和句法展开研究,通过实验分析不同词性和依存关系对句子深层逻辑的影响。对人们使用自然语言输入的句子提炼成以词语为单位的主要成分,并转化为易于计算机理解知识的实体和逻辑关系,同时设计了一套可行的解决方案辅以实体在游戏引擎中生成动画,最终以视频的形式进行可视化呈现。

关键词:虚拟仿真;Unity3D;百度 AI;自然语言处理;依存句法分析

中图分类号:TP391.1

文献标识码:A

文章编号:1673-629X(2021)09-0214-07

doi:10.3969/j.issn.1673-629X.2021.09.036

Visual Simulation System of Natural Language Based on Dependency Grammar

YUAN Yu-xuan¹, LI Fang¹, CHEN Ke-qi², HAN Zheng¹

(1. School of Digital Arts and Design, Dalian Neusoft University of Information, Dalian 116023, China;

2. Warwick Business School, University of Warwick, Coventry CV4 7AL, UK)

Abstract: At present, the virtual simulation system mainly revolves around physical and chemical experiments and mathematical formulas, and there are few simulation cases for literary works. The resolution of Chinese language input methods by using deep learning model on more advanced computing power system becomes prevalent in research field. However, study shows that the Chinese language is much different than English as more of a logogram language with non-regular grammar rules, which creates natural barrier on the application of such technology. In terms of the actual implementation of such technology, it would largely enrich the visualized contents of natural language processing. Therefore, we will analyze the influence on the logical relationship through an experimental analysis to take research on the interdependency and different parts of speech by inspecting the particular grammar and lexical definition of the Chinese language. By extracting the sentences from the users and converting these extractions into a real identity and logical relationship that could be processed by computers, and designing a solution in which would act as an ancillary unit of such identity to help processing images and visualized to a video form in gaming engine.

Key words: virtual simulation; Unity3D; Baidu AI; natural language processing; dependency parsing

0 引言

自然语言处理技术是在程序设计中常用的人工智能算法,也是人工智能的热点研究方向。主要应用场景包括个性化推荐、搜索纠错等。神经网络是自然语言处理以及人工智能领域最重要的方法之一。为了让计算机理解自然世界的知识,可以根据具体研究对象的特征构建不同的神经网络结构,如适用于图像处理

的卷积神经网络(convolutional neural networks, CNN)、适用于文本和序列数据的递归神经网络(recurrent neural networks, RNN)和长短期记忆网络(long short-term memory neural networks, LSTM)等^[1-2]。尽管这些网络结构在处理不同的任务时表现出不同的性能,对于具体任务仍很难确定使用哪种网络结构^[3],所使用模型的训练过程同样需要耗费大量

收稿日期:2020-10-07

修回日期:2021-02-08

基金项目:辽宁省社会科学规划基金项目(L19CXW006)。

作者简介:袁雨轩(1999-),男,CCF会员(D2005G),专业为数字媒体技术;李放,副教授,研究方向为虚拟仿真。

时间。为了覆盖更多的语言场景,本系统依赖百度 AI 开放平台的自然语言处理技术接口。百度 AI 开放平台提供了全面的语言处理基础能力和语言处理应用技术,并以标准化的接口封装。且该平台的训练语料以中文为主,对中文输入的接受能力较强。

对自然语言处理技术来说,其最细的划分粒度是词语。而在计算机仿真领域中可以使用三维模型、图片等来模拟名词,使用动画、声音等交互系统模拟动词。由于中文的能指极其复杂,在自然语言处理技术难以解决的词语多义性问题上更是雪上加霜。文献[4]提出了中文句式的三个层级:(1)表层:形式结构,语言中语法项的线性配列式。如主语+谓语构成 SV 句式等。(2)中介层:表现法,语言中线性配列的特定样式。如抒情、描写手法等。(3)深层思维方式:反映说话者心理的主体意识。并提出深层思维方式的表达注重于主观感受的抒发,即为了满足句子表层结构的语法在表达中的正确性,其深层思维方式可以不严格满足语法规则。由于深层思维方式不执著于逻辑和形式结构规范,其表达与所指的联系也更紧密。

基于以上分析,文中将深入讨论句子中不同实体之间的依存关系,排除中介层和表层结构中的修饰成分,从而得到易于计算机理解句子深层逻辑的输入。并提出了一种可视化的虚拟仿真解决方案,用实体所指代的对象代替中文复杂的能指,并以视频的形式可视化呈现。

1 系统开发中的关键技术

1.1 中文依存关系的研究

聚类(cluster)是一组数据对象的集合,在文中对应着人们使用自然语言输入、不利于计算机理解的句子。文献[5]在引用中总结到,聚类的基本划分方法存在需要事先给出样例的缺陷。而人工定义的语法规则也是难以穷举、无法事先给出的。同时,中文的依存句法分析区别于英文,需要对句子进行分词^[6]。对过于复杂而无法进行预处理的数据可以考虑使用基于深度学习算法的模型。百度 AI 依存句法分析接口提供了使用百度日常搜索内容做训练数据的 Query 模型,该模型擅长对口语 Query 句式的处理。以及来源于全网网页数据的 Web 模型,该模型偏向表达规整的书面语法。以上模型可以解析出共 34 种依存关系。对此可以根据集合学习的思想,构建多个对不同数据的处理效果有明显偏好的模型,整合出一个更强大的模型做最后决策。

为了研究解析出的不同依存关系对句子深层逻辑的影响程度,实验 1.2 将使用百度 AI 的短文本相似度接口,采用 CNN 模型验证去除不同依存关系后对句子

结构的影响,相比于该接口提供的词袋模型(bag-of-words model, BOW)和 RNN 模型, CNN 模型的显著特点是对序列输入敏感,但是语义泛化能力较弱。本实验更关注实体之间的逻辑,在对比实验的过程中仅去掉某一个依存关系对应的实体,因此对词语泛化能力的要求较低,且 CNN 模型对句子保留的局部特征更敏感,因此选择 CNN 模型可以在实验中获得更理想的结果。

1.2 实验

中文的语言特点决定了多数的语言场景需要以复句的形式完成。处理复句的难点在于各个分句间语义关系的准确识别。文献[7]提出了一种基于句内注意力机制的多路 CNN 网络结构,可以识别分句间的关联特征,以及正确识别复句的逻辑关系。文中虚拟仿真工作产生的效果主要来自各分句中的名词模型,其中的逻辑关系由该分句中的谓语提供,并且中文的表达习惯可以保证句子输入的语序与视频播放的时顺相对应,因此各分句之间可独立看待。为了研究使用自然语言输入的单个句子中词语之间的依存关系,以“袁明吃蛋挞”为例,表 1 将展示该例句在去掉不同依存关系所对应的词语后与原句子的短文本相似度,该数值介于[0,1]之间,反映了被去掉的依存关系对句子逻辑的影响程度。首先排除由复句产生,或无实际意义的虚词产生的依存关系,如标点符号、关联词等共计 17 种。

表 1 语义相似度对比实验

序号	例句	代称	缺少类型	相似度
1	袁明(吃)蛋挞	HED	核心	0.915 52
2	袁明吃蛋挞		相同内容	1
3	袁明吃(得)蛋挞	DEI	得字结构	0.967 623
4	袁明爱吃蛋挞(吗)	MT	语态结构	0.958 391
5	袁明(把)蛋挞吃	BA	把字结构	0.957 993
6	袁明(昨天)吃蛋挞	TMP	时间关系	0.957 062
7	袁明疯狂(地)吃蛋挞	DI	地字结构	0.950 521
8	我们(大家)吃蛋挞	APP	同位关系	0.950 041
9	蛋挞袁明(所)爱	SUO	所字结构	0.928 912
10	袁明吃(美味的)蛋挞	DE	的字结构	0.926 912
11	袁明爱吃(美味)蛋挞	ATT	定中关系	0.926 912
12	袁明(以)吃蛋挞	ADV	状中结构	0.918 116
13	蛋挞(被)袁明吃	BEI	被字结构	0.894 248
14	袁明吃(两个)蛋挞	QUN	数量关系	0.883 784
15	袁明(在上海)吃蛋挞	LOC	处所关系	0.831 824
16	袁明吃蛋挞(和披萨)	COO	并列关系	0.789 764
17	(袁明)吃蛋挞	SBV	主谓关系	0.663 492
18	袁明吃()	VOB	动宾关系	0.619 409
19	袁明关于(蛋挞)	POB	介宾关系	0.356 925
20	哈哈(袁明吃蛋挞)		无关内容	0.205 001
21	Hello world(袁明吃蛋挞)		无关内容	0.134 363

1.2.1 讨论

相似度是一个十分复杂的概念^[8]。根据实验序号2可知,两个完全相同的句子相似度为1,序号20和21是两个与原文无关的句子,其相似度在本实验中的样例均小于0.3。序号19介宾关系是另一个相似度接近0.3的样本,原因是丢失掉宾语后与原句本身差异过大。对以上结果可暂时作如下假设:短文本相似度接近0.3甚至更小的句子与原文的出入较大,故应取大于相似度0.3的结果做讨论。

文献[9]在文本相似度计算中提到了句子的关键成分与修饰成分的概念,而人们往往从关键成分就可以了解一个句子的大概意思。在介于(0.9,1)区间内存在十种依存关系,除核心关系外,其他的依存关系均是对句子逻辑影响较小的结构成分或修饰成分。以上两种成分的存在保证了句子表层语法规则的正确。以“地”字结构为例,其语法解析树如图1所示。该依存关系仅修饰动词“吃”。去掉修饰成分对句子要表达的深层逻辑影响很小,此句子的动画依然可以由“袁明吃蛋挞”来表示。对此结果不妨假设在(0.9,1)区间内的修饰成分在生成动画时可以忽略。

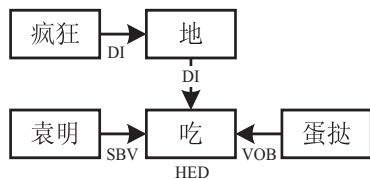


图1 例句“袁明疯狂地吃蛋挞”

文献[10]中的例句验证了,在以中文复句输入所返回的语法生成树中,关系词都是动词。关系词在复句中常作为分句的核心节点出现,因此本实验的单句中核心节点的作用类同于复句中的关系词。

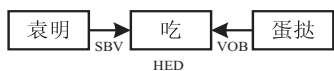


图2 例句“袁明吃蛋挞”



图3 例句“袁明蛋挞”

对于图2所示的核心关系需要特别讨论,因为动词“吃”是SVO句式的关键成分而不是修饰成分,显然与上文的假设相矛盾。这是因为本实验使用的CNN模型虽然对局部特征敏感,但是“袁明”和“蛋挞”之间没有任何成分解释两个词之间的逻辑关系,于是该模型默认把“袁明”作为修饰成分,原句的意思被泛化成图3描述的“袁明牌蛋挞”了。CNN模型在卷积层内部状况为词向量的拼接。词向量是计算机理解词汇的媒介,这种表示方式是将词语映射到一个高维且稠密的向量空间中,用空间距离的形式反映两个

词之间的系。文献[11]提到,将词语在计算机中向量化表示的常用方法有one-hot、词向量和Word2vec模型等。虽然百度AI平台没有描述短文本相似度接口使用的CNN模型中的词向量在卷积层内部的具体情况,但该平台语言处理基础能力的SDK中也提供了词向量表示接口。词向量模型训练过程的初始状态下,对语料库内包含的所有词向量随机赋初值,按照词语在样本句子中的位置相应地调整词向量在空间中的位置,以“袁明吃蛋挞”为例句训练词向量,该句子由人名+动作+宾语组成。如图4所示,当再次输入“韩明”和“蛋挞”时,“吃”字在空间中的优先级会小于其他的动词如“买”。输入“李明”和“吃”之后“蛋挞”的优先级同样会高于其他名词。类似的,使用训练集中的句子不断调整词向量的位置,多次训练后的词向量可以作为特征值用于在计算机中表示该词语。

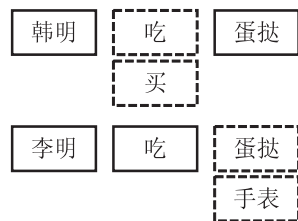


图4 例句“袁明吃蛋挞”的词向量训练

基于以上结论,对例句词语分别调用百度AI的词向量表示接口,返回类型为1024维空间大小的词向量,其中对“袁明”的返回值为 $[-0.074\ 207\ 4, -0.458\ 546, -0.010\ 252\ 2, 0.010\ 083\ 9, \dots]$,对“吃”的返回值为 $[0.015\ 029\ 4, 1.488\ 3, -0.482\ 325, -0.740\ 492, \dots]$,对“蛋挞”的返回值为 $[0.015\ 029\ 4, 1.488\ 3, -0.482\ 325, -0.740\ 492, \dots]$ 。CNN模型检测文本相似度方法的内部情况是将以上三个词向量拼接成矩阵表示,在池化层完成对数据降维的工作^[12]。文献[13]详细描述了目前主流的池化方法:

(1)平均池,选择池窗口中的平均值表示池区信息。

(2)最大池,选择池窗口中的最大值,该方法可以获得池区的最优值,也是最常用的方法。

(3)K-max池,选择池窗口中的最大K值,该方法是最大池方法的扩展,可以更好地保持文本特征。

(4)多池,该方法是针对文本任务的特点,获得文本的一维向量表示,并在一维向量上使用其他池化方法分段。

该方法的优点是能保留一些词序信息,可以视为最大池的一种变体。本项目对于文本相似度接口返回的数值为离散数据,且难以描述实际意义,可以考虑使用聚类方法处理。参考多池的分段方法,研究返回的三个词向量之间的关系可以通过计算欧氏距离表示词

向量在二维空间下的位置关系,其中 D 为绝对距离, X, Y 为实体, x, y 为特征值,计算方法如下:

$$D(X, Y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}$$

最终计算“袁明”到“蛋挞”的距离为 17.433 217 41, 句子“袁明吃蛋挞”的距离之和为 39.262 644 02。另一个例子,“袁明跳蛋挞”是一句显然不符合表达习惯的句子,此句子与“袁明蛋挞”的相似度为 0.767 38,且符合上一节中介值于(0.3,0.9)之间的假设,这个句子的距离之和为 46.659 304 67。将以上距离关系映射到二维空间下的表示如图 5 所示,由于词向量的特征值仅代表在向量空间下的绝对位置,因此该坐标系下的单位仅作运算,无实际意义。

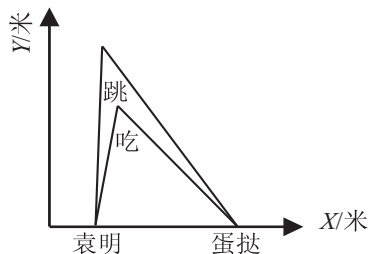


图5 词向量在二维空间下的映射

由数据降维后的结果可见,“吃”字对卷积运算的结果影响较小,以上结果是因为 CNN 模型运算词向量时对特征敏感而本例句过于简单所导致,所以为了保持句子的完整结构,核心关系应作为特例保留。

1.2.2 结论

基于上文的实验,文本相似度处于(0.3,0.9)区间的依存关系有主谓关系、介宾关系和动宾关系以及做特例的核心四种依存关系作为句子的关键成分,相似度同样处于(0.3,0.9)区间的处所关系、并列关系、数量关系和“被”字结构是对句子逻辑影响比较大的修饰成分或结构成分。利用已经确定好的八种重要的依存关系,可以重新组合出更接近句子深层含义的 SVO 句式。在不改变句子本身逻辑的前提下,去掉尽可能多无需以动画展示的修饰成分,对第二节中生成动画的工作显然是更友好的。

2 系统的设计与实现

2.1 系统架构设计

基于依存关系的自然语言可视化仿真系统使用 C/S 分离式架构设计,其时序图如图 6 所示。使用 .net 框架中的 TCP 协议通信,分离式架构的优点是可以方便本系统在应用层面开发时的扩展^[14],例如将模型资源保存在服务器上的数据库中以减少客户端资源的使用,以及在不修改客户端的条件下更新数据。本系统的核心问题是对以上依存关系合理性的验证,故

暂时不考虑在架构上做性能优化。

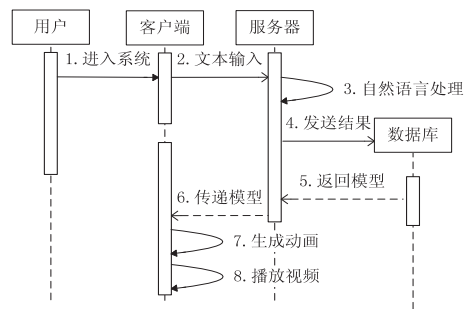


图6 UML 序列图

2.2 服务器的系统开发

服务器的开发需要调用百度 AI 开放平台的自然语言处理接口。使用前需要在官方网站申请 API_KEY、SECRET_KEY 和 APPID 并下载 C# 平台的 SDK,在 Visual Studio 中新建项目,添加对 SDK 中类库的引用,并创建 InputHandle 类和 NetWork 类。调用接口时的请求文本应采用 GBK 编码,而此编码在 .NET Core 平台上不可使用。在程序启动时注册 RegisterProvider 能提供平台不可使用的编码格式,具体代码如下:

```
System.Text.Encoding.RegisterProvider(System.Text.CodePagesEncodingProvider.Instance);
```

```
System.Text.Encoding.GetEncoding("GB2312");
```

2.2.1 InputHandle 类

InputHandle 类包含了调用自然语言处理接口和声明 Json 解析的方法。首先,将输入的句子传入到依存句法分析方法中,使用正则表达式解析返回的 Json 结果,并放入到如下格式的结构体数组中。

```
public struct Word {
    public int head; // 头节点
    public string word; // 词
    public int id; // 节点编号
    public string deprel; // 关系
};
```

根据此结构体数组,可以获得一棵有序的语法解析树,使用 Equals() 方法判断结构体中 deprel 的取值。关键代码如下:

```
string Subject, Verb, Object;
foreach (Word w in Words) {
    if (Equals(w.deprel, "SBV") Subject = w.word;
    if (Equals(w.deprel, "HED") Verb = w.word;
    if (Equals(w.deprel, "QUN") Verb = w.word;
    if (Equals(w.deprel, "BEI") Verb = w.word;
    if (Equals(w.deprel, "COO") Object = w.word;
    if (Equals(w.deprel, "VOB") Object = w.word;
    if (Equals(w.deprel, "POB") Object = w.word;
    if (Equals(w.deprel, "LOC") Object = w.word;
}
```

将以上代码赋值后的 Subject、Verb、Object 变量有序地存入字符串数组,此方法封装到 DepPraseHandle 类中,并将字符串数组作为参数传递给 NetWork 类。

2.2.2 NetWork 类

对于服务器和客户端之间的通信问题,本系统采用 .net 基于套接字为 TCP 协议扩展的 TcpClient 类和 TcpListener 类共同实现。TcpListener 类可以侦听来自 TCP 网络客户端的连接。由于 TCP 协议是以稳定字节流的形式将数据在网络间传输^[15],TcpClient 类中的 GetStream() 方法可以返回用于发送和接收的网络流。对于字节类型可以使用 BinaryWriter 类和 BinaryReader 类共同实现将基元类型读取和写入的二进制值的操作,以上提及的两个类位于命名空间 System. IO 下。创建 ReceiveMessage() 方法,包含使用 BinaryReader 读取到网络流中客户端发送的句子,并作为参数按上文所述的顺序调用 InputHandle 类中的方法,最终根据返回的消息找到模型在服务器上保存的位置,并使用 FileStream 类的方法写入流数据,关键代码如下:

```
public byte[] FileReader(string path)
{
    FileStream fileStream = File.Open(path, FileMode.Open);
    Console.WriteLine(path);
    byte[] array = new byte[fileStream.Length];
    fileStream.Read(array, 0, array.Length);
    fileStream.Close();
    return array;
}
```

创建 SendMessage() 方法,包含将返回的字节数组使用 BinaryWriter. Write() 方法写入到网络流 NetWorkStream 中。其中 NetWorkStream 位于命名空间 System. Net. Sockets 下。将上述方法封装于 NetWork 类,最终的服务器 UML 类图如图 7 所示。

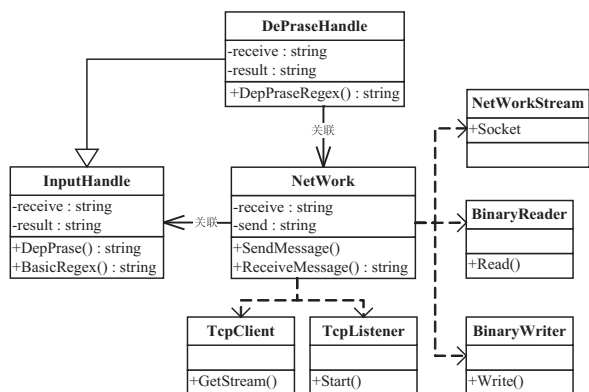


图 7 服务器 UML 类图

2.3 客户端的系统开发

2.3.1 命名实体处理

在 Unity3D 中将每个实体模型制作为 GameObject

类型的预制体,并以实体的含义命名后保存在数据库中。

(1) 名词类。

由于需要表示的动画不仅是由模型组成,在 Unity3D 中实现名词还可以意味着动画、天空盒子、粒子系统、声音、视频等功能,故需要根据词语创建模型,并对非模型类的命名实体挂载一个继承自 MonoBehaviour 类的脚本,并在 Start() 方法中调用不同组件的 Play() 方法,以便在生命周期开始时执行。

(2) 动词类。

动词类在中文的单句中作为语法树的根节点或度大于一的节点控制前后名词类的逻辑,不仅需要访问名词类,还要根据动词本身的意思让名词类执行不同的操作如的开始、结束、改变位置、复制多个等,在动词类中创建 FunSubject()、FunVerb()、FunObject() 三个方法,分别对应着与名词的主动关系、动作和被动关系。通常情况下,数量关系、处所关系等修饰成分充当谓语成分时对应着 FunObject() 方法的实体,表示宾语被动产生的动作。而核心 HED 往往由动词产生,对应着 FunSubject() 方法,表示主动发起的动作。以 FunSubject() 方法为例,获取名词所对应命名实体的代码如下:

```
public Inventory Bag;
public GameObject Mod;
public string Subject;
foreach (GameObject b in Bag.itemList) {
    if (Equals(b.name, Subject) == true) Mod = b;
}
```

获取到模型后,首先要分析该动词的含义,在 FunVerb() 方法中以低代码化的方式添加简单的代码片段。例如下面的代码为动词“害怕”添加了音频并向后移动了主语模型的位置。

```
public void FunVerb() {
    Instantiate(audioSource);
    Mod.transform.position += new Vector3(0, 0, -1);
}
```

2.3.2 背包系统实现

根据系统架构设计,客户端需要接收服务器发送的字节数组。并按照字节数组中的顺序在客户端写入文件并序列化地生成模型,可以考虑使用 GameObject 类型的泛型列表保存模型,这种方法的优点是易于根据名称随机读取模型。

创建背包的方法如下:

```
public class Inventory : ScriptableObject {
    List<GameObject>itemBag = new List<GameObject>();
}
```

2.3.3 控制器设计

首先创建 Receive() 方法接收到服务器以字节流

格式发送的命名实体,其关键代码如下:

```
public string Receive()
{
    byte[] recvBuf = new byte[102400];
    networkStream.ReadTimeout = 2000;
    int bytesRead = networkStream.Read(recvBuf, 0, 102400);
    if (bytesRead > 0)
    {
        string message = Encoding.UTF8.GetString(recvBuf);
        return message;
    }
}
```

创建 `FileWrite()` 方法,用于将字节数组写入文件。在写入时需要注意判断服务器写入文件的顺序,可以使用标签或分批发送的方式区分好不同实体,下面以动词为例,写入方法如下:

```
public void FileWrite(string message)
{
    int a = message.IndexOf("%YAML 1.1");
    //在 a 下标的 100 位之后继续寻找下一个文件的报头
    int b = message.IndexOf("%YAML 1.1", a + 100);
    string verb = message.Substring(a, b - a);
    FileStream vFile = new FileStream(@"Assets\Resources\
Verb.prefab", FileMode.Create, FileAccess.ReadWrite);
    StreamWriter Vwriter = new StreamWriter(vFile);
    Vwriter.WriteLine(verb);
    Vwriter.Close();
}
```

若制作的实体仅由 Unity3D 自带的模型组成,可以直接使用 UTF-8 格式编码后使用 `StreamWriter` 以字符串的形式写入文件,这种方法相比直接操作字节而言更易于计算。否则需要把制作的命名实体及相关资源以 `unitypackage` 包的形式导出,使用 `BinaryWriter` 将二进制值写入文件后导入项目。最终将生成的实体放入背包中,方法如下:

```
Bag.itemList.Add(Resources.Load("Verb") as GameObject);
```

在执行了以上部分的代码后,此时已获取到句子中涉及到的所有模型和对应关系,名词类模型包含了具体的属性,动词类模型可以获取到名词的属性并执行相应的事件。要实现动画的播放,则只需要依次从背包中取出对应的实体并实例化在场景中即可,关键代码如下:

```
public Inventory Bag;
void FindVerbsInBag() {
    foreach (GameObject go in verbBag.itemList) {
        if (go.name == Subject) Instantiate(go);
        if (go.name == Verb) Instantiate(go);
        if (go.name == Object) Instantiate(go);
    }
}
```

在控制器脚本的生命周期结束时,还应添加资源

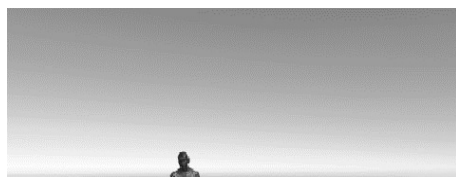
销毁机制并清空背包,避免该场景在下一次激活时执行写入或查找操作出现异常,可以参考如下代码:

```
private void OnDestroy() {
    File.Delete("Assets/Resources/Verb.prefab");
    File.Delete("Assets/Resources/Subject.prefab");
    File.Delete("Assets/Resources/Object.prefab");
    bag.itemList.Clear();
}
```

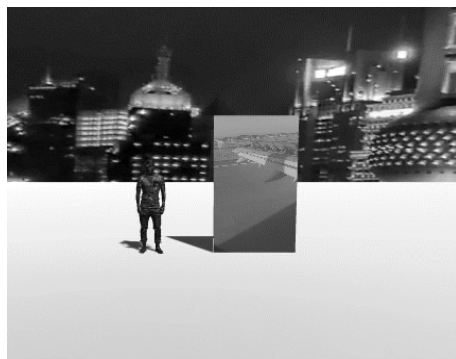
该方法在添加时需要慎重考虑,因为在复句或较复杂句式的处理中,该模型与下一个句子可能仍然存在联系。应考虑具体需求灵活变动,例如在销毁前增加条件判断或创建缓存机制等。

3 系统测试

以“袁明害怕小狗”,“袁明坐飞机去上海”为例,为这句话中出现的命名实体制作相应的模型,这两句话在本系统中的运行效果如下:图 8(a)按顺序出现了模型,并播放了声音表示害怕,同时人物的模型向后移动了一段距离。图 8(b)按顺序出现了模型,并在视频播放完成后更换了天空。



(a) 例句“袁明害怕小狗”



(b) 例句“袁明坐飞机去上海”

图 8 例句展示

在不销毁以上模型的前提下,输入一个重新组合的句子“袁明到上海,小狗挨着香蕉”,再次制作命名实体“香蕉”对应的模型,添加动词“到”和“挨着”,并识别逗号用做分隔符以两个分句的形式连续调用上述方法。本系统将以两个单句的效果播放动画,最终效果如图 9 所示。



图 9 例句展示

4 结束语

在现实生活中,人们通常更加熟悉该词语本身所指代的对象,只是不熟悉这一对象在某种语言下的表示中所使用的语言符号。文中为能指与所指之间提供了一种可视化的解决方案,在处理 SVO 结构单句时可以表现出理想的效果,适合应用于汉英字典例句的可视化翻译、同一语料库内的内容的可视化。对于复句或结构更复杂的句式可以在输入时根据需求切分并逐条处理,以满足大多数应用场景。但是对于不符合语法规则的口语句式或输入错误时表现出的处理结果并不理想。除此之外,标注大量模型的工作是另一个难题。笔者将在日后继续研究此系统的实用性扩展。

参考文献:

- [1] 黄江珊,高 娃,宿 静,等. 基于卷积神经网络与长短期记忆网络的医学影像数据管理方法研究[J]. 医学与社会, 2020,33(6):84-89.
- [2] 吴建生,谢永盛,金 龙. 基于均生函数的长短期记忆神经网络径流建模研究与应用[J]. 建模与仿真,2020,9(2): 163-177.
- [3] 周飞燕,金林鹏,董 军. 卷积神经网络研究综述[J]. 计算机学报,2017,40(6):1229-1251.
- [4] 刘宓庆. 思维方式、表现法和翻译问题[J]. 现代外语,1993 (1):12-15.
- [5] 徐晓旻,肖仰华. KBAC:一种基于 K-means 的自适应聚类 [J]. 小型微型计算机系统,2012,33(10):2268-2272.
- [6] 马金山. 基于统计方法的汉语依存句法分析研究[D]. 哈尔滨:哈尔滨工业大学,2008.
- [7] 孙凯丽,邓沌华,李 源,等. 基于句内注意力机制多路 CNN 的汉语复句关系识别方法[J]. 中文信息学报,2020, 34(6):9-17.
- [8] AFZALI M, KUMAR S. An extensive study of similarity and dissimilarity measures used for text document clustering using k-means algorithm[J]. International Journal of Information Technology and Computer Science,2018,10(9):64-73.
- [9] 李 彬,刘 挺,秦 兵,等. 基于语义依存的汉语句子相似度计算[J]. 计算机应用研究,2003,20(12):15-17.
- [10] 杨进才,涂馨丹,沈显君,等. 基于依存关系规则的汉语复句关系词自动识别[J]. 计算机应用研究,2018,35(6): 1756-1760.
- [11] 唐 明,朱 磊,邹显春. 基于 Word2Vec 的一种文档向量表示[J]. 计算机科学,2016,43(6):214-217.
- [12] ZHAO Bendong, LU Huanzhang, CHEN Shangfeng, et al. Convolutional neural networks for time series classification [J]. 系统工程与电子技术:英文版,2017,28(1):162-169.
- [13] ZENG Shuifei, MA Yan, ZHANG Xiaoyan, et al. Term-based pooling in convolutional neural networks for text classification[J]. 中国通信:英文版,2020,17(4):109-124.
- [14] WANG Haoming, LIN Baowei, TAMAKI T, et al. Development of a client-server system for 3D scene change detection [J]. Journal of Software Engineering and Applications, 2013,6(7):17-21.
- [15] SINGH R, PATEL B. Performance improvement of TCP over wireless network [J]. International Journal of Engineering Science and Technology,2011,3(11):8106-8113.