现代JavaScript框架：调查研究

摘要：随着Web的日益普及，一些新的Web技术应运而生，与HTML相比，作为静态编程语言引入Web应用程序。JavaScript是提供动态网站的语言，该网站主动与用户通信。JavaScript在今天的Web应用程序中作为客户端脚本语言以及在服务器端使用。JavaScript语言支持模型-视图-控制器（MVC）体系结构。该体系结构维护可读代码并清晰地分离程序代码的各个部分。本研究的主题是对比流行的JavaScript框架： AngularJS，Ember，Knockout，Backbone。所有四个框架都是基于MVC或类似的架构。在本文中，将介绍每个框架的优缺点、对应用程序速度的影响、测试这些JS应用程序的方法以及提升代码安全性的方式。

关键词：客户端 服务器 应用程序 MVC架构 JavaScript AngularJS Ember Knockout Backbone

1. 绪论

已经实现为桌面独立的应用程序越来越多地转向Web和移动平台。出于这种需要，已经开发了许多新的客户端-服务器、移动技术和框架，并且设计模式也积极地被使用。用于改进Web应用程序的最流行的体系结构包括：模型视图控制器（MVC），模型视图展示器（MVP），模型视图ViewModel（MVVM）和许多其他结构。

这个研究的目的在于提出四种流行的JavaScript框架——AngularJS、Ember、Knockout和Backbone，以展示他们的优缺点，根据某些标准进行侧面对侧面分析，并总结分析。

第二章简要介绍了每个框架，第三章、第四章和第五章提供了每个框架的比较概述及优缺点。在最后给出结论。

1. 流行的JavaScript框架

本章简要概述了所要分析的框架。

A. AngularJS

AngularJS是一个前端Web应用程序框架，它的第一个版本是由Google在2010年设计的，目前活跃的版本是从2018年开始的1.7版本。AngularJS是MEAN堆栈的一部分，由 MongoDB数据库、Express.js应用程序服务器框架、AngulJS和Node.js组成，作为服务器运行时的环境。

它基于MV\*（Model View Whatever）架构，与MVC架构略有不同。在MVC中，我们区分负责数据管理的模型组件，负责向用户显示数据的视图组件，以及控制模型和视图之间的交互并表示应用程序的业务逻辑的控制器组件。而在MV\*体系结构中，连接模型和视图的内容并不相关，但重要的是每个模型更改都会影响视图，反之亦然。在MVC中，控制器控制情况并由属性和功能组成。它使用ng-controller指令定义。每个控制器都接受$ scope对象作为参数，或接受引用应用程序的对象。 Scope对象表示控制器和视图之间的介质，它是我们添加变量和函数的对象。该对象演示了一种称为依赖注入的技术。这样，函数不以任何方式依赖于变量。其他重要的AngularJS指令是：ng-app，ng-init，ng-include，ng-model，ng-bind，ng-repeat，ng-show，ng-hide，ng-click。

B．BackboneJS

BackboneJS是一个JavaScript库，它提供了一种灵活简约的解决方案，用于将业务逻辑和用户界面分离。它基于类似MVC的体系结构，使用模型、DOM视图抽象数据并使用事件绑定模型和试图。Backbone依赖于JavaScript文件underscore.js，它必须与backbone.js文件一起包含在内。Backbone架构包含以下内容：

1）HTTP请求——用于从客户端向服务器发送请求；

2）路由器——用于使用URL将应用程序连接到操作和事件；

3）视图——负责表示层；

4）事件——触发绑定到事件发生对象的适当函数；

5）模型——包含应用程序数据和数据逻辑；

6）集合——模型列表；

7）数据源——与数据库的连接。

C．EmberJS

EmberJS是一个用于开发Web应用程序的JavaScript框架。 它使用MVC架构并具有以下部分：

1）模型——每条路线都有一个相关的模型，并显示模型中的数据是路线的分配；

2）视图——由于Handlebar模板的存在，它们很少被创建；

3）模板——一个标准模板，它有一个页眉、页脚、至少一个{{outlet}}和一个表达式，它将根据当前的URL加载指定的内容；

4）控制器——管理显示逻辑并控制路径、模型和视图之间的操作;

5）路由——定义应用程序如何表示给用户，加载模板和必要的数据，而路由器根据给定的URL确定应加载哪些路由。

Ember的相关特征是：计算属性，观察者，绑定。

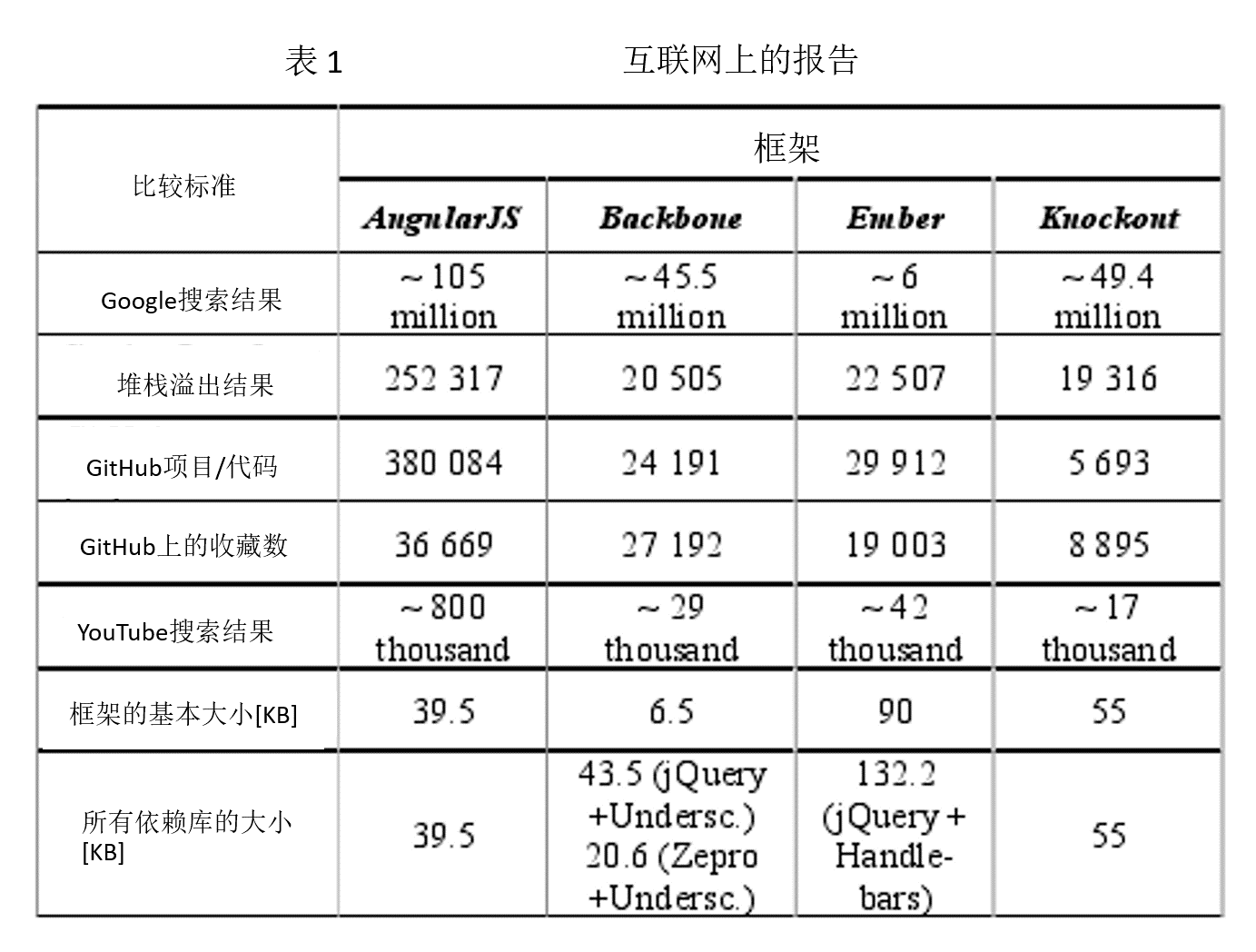
D．KnockoutJS

KnockoutJS是一个基于MVVM架构的JavaScript库，源自MVC模式。它为我们提供了构建丰富应用程序的支持，我们可以与之交互。View表示使用HTML构建的用户界面。

View绑定到ViewModel，因此ViewModel中发生的更改将使用双向数据绑定的概念自动传播到View。 ViewModel是一个绑定Model和View的类，即将实际数据与其用户表示绑定。 该框架所基于的三个重要概念是：可观察量、数据绑定和模板。组件构成应用程序并提供代码可重用性。

1. 总的信息分析

在本研究的第一步中，我们比较了各框架在互联网上的表述、框架的大小以及它们对应用程序速度的影响，如表1所示。图一显示了在2010年1月至2018年3月期间的兴趣对比图。AngularJS是互联网上最常见的框架。它提供了大量的书面教材和视频教程，使学习更加容易，缩短了熟悉框架所需的时间。Backbone在2012年5月至2013年8月期间引起了用户的极大兴趣，但此后对此框架的兴趣正在下降。AngularJS从2013年开始经历了突然的扩张，直到今天，这个框架还落后于其他三个。



比较这些框架的另一个重要因素是它们的大小。由于用户在应用程序加载方面没有太多耐心，因此在Web应用程序中这一因素非常重要。最小化的版本大小比较也展示在表1中。

Backbone更像是一个库而不是一个框架，研究表明，它可以使快速应用程序快速响应用户操作。Backbone有许多插件可以与它集成以获得所需的功能。与AngularJS相比，具有更大库的Ember和Knockout使我们更容易通过自己做这些事情来做事，例如在更改模型时自动刷新视图组件，Backbone不是这样，并且使用它产生更多代码。更少的代码意味着更容易测试和更少的错误空间。 Ember拥有最大的库，但它可以是一个优势，因为它提供了大量的内置支持。Ember对大型项目可能是的正确选择。

1. 测试和安全分析

所有四个框架都支持不同类型的测试，包括功能（黑盒）测试和结构（白盒）测试。这些技术支持最着名的JavaScript测试模块，例如用于AngularJS的Karma和Jasmine，用于Backbone的Mocha、Jasmine和Sinon等等。在AngularJS中，完全支持依赖注入，这在软件测试方面是一个很大的优势。Web浏览器中有不同的扩展，例如谷歌浏览器，以帮助调试，由开发分析框架的团队开发。使用Angular Batarang可以实现调试AngularJS。加载Batarang时，可以选择我们感兴趣的范围和具体要素。除了读取元素的值之外，Batarang还提供了测试性能，可视化依赖图等的功能。与Ember一起使用的插件是Ember Inspector。该工具提供了Ember应用程序各个部分的视图，模型处理以及路径的详细视图。Knockout Context Debugger添加了包含显示任何所选元素的当前绑定的元素的部分。Backbone Debugger实时显示应用程序中的所有模型，集合，视图和角色。

A.表现

除了大小之外，应用程序的速度还受其他因素的影响，因此我们无法明确判定哪个框架速度更快。应用程序的速度还取决于我们应用程序源代码的结构。为了进行比较，附上了一个简短测试的结果，该测试打印了500个数字，在所有四种分析技术中发布[6]。测试显示每种技术中每秒的操作次数。在这个测试实例中，如图2所示，AngularJS已经显示出了最佳性能。

B.安全性

Retire.JS工具有助于发现我们的JS应用程序的漏洞。Retire.JS建议根据发现的漏洞改进五个不同版本的AngularJS，一个版本的Backbone和十五个版本的Ember。AngularJS和Backbone都没有常见的漏洞和暴露（CVE），但Ember有五个公共CVE文档。应用程序中最容易受到攻击的地方是模板。它们是DOM最常被攻击的地方。Backbone和Ember不使用自己的模板操作库，而AngularJS使用自己的模板系统。Knockout允许执行注入HTML5数据属性的任意JavaScript，因此这是受攻击最多的部分。当调用ko.applyBindings函数时，位于databind属性中的每个JS程序代码都将执行一次。正如2013年的项目Mustache-security所研究的那样，所有四个库都在不同程度上容易受到这个因素的影响[7]。用于确定模板系统安全性的排名系统具有以下特征：

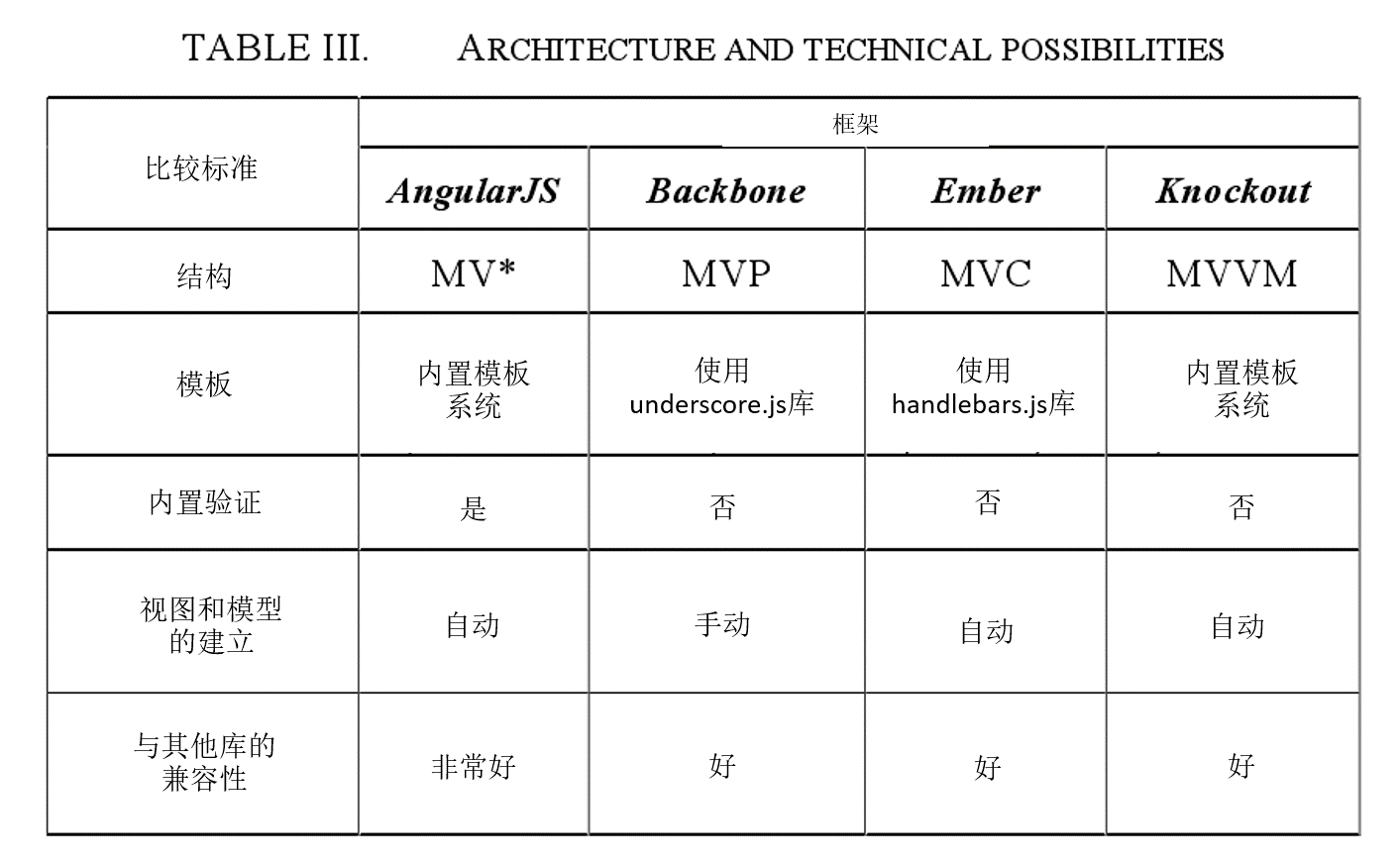
* {} SEC-A是否在不使用eval或Function的情况下执行模板表达式？ （是=通过）
* {} SEC-B执行范围是否隔离或沙盒？ （是=通过）
* {} SEC-C只能将脚本元素用作模板容器吗？ （是=通过）
* {} SEC-D框架是否允许，鼓励甚至强制分离代码和内容？ （是=通过）
* {} SEC-E框架维护者是否有安全响应程序？ （是=通过）
* {} SEC-F框架是否允许或鼓励使用安全的内容安全策略（CSP）规则（是=通过）

分析框架的结果在表2中给出。使用underscore.js而不是Backbone，是因为Backbone使用的是它自己的模板。

5、分析JavaScript框架的优缺点

Backbone仅代表应用程序的核心，或代码编辑的方式。 我们可以用各种插件来丰富Backbone；没有这些插件，它就是一个小型库，不会带来很多新东西。AngularJS是增长最快，变化最快的，因此IT公司仍然没有在很大程度上选择该技术。Ember在它提供的功能方面类似于AngularJS，但是互联网上有很多关于它的内容在实践中不起作用。Knockout没有集成路由，这在AngularJS中可用。 AngularJS还包括依赖注入概念，提供许多服务（$ http，$ log，...）、模块、过滤、表单验证等。例如，这些概念未在Knockout中实现。Knockout可以用作不太复杂的应用程序中的框架，不需要很好的用户界面显示控制。

表3显示了系统架构方面的比较和技术可能性。作为比较的第一个标准，分析了框架的体系结构。 所有这四种技术都基于MVC架构的变体。 Backbone被分配给MVP体系结构，其中HTML和DOM表示View组件，Backbone View表示Presenter组件。



分析的第二个标准是使用模板的方法。Backbone使用underscore.js来处理模板，Ember使用handlebars.js，而Knockout有自己的集成系统用于模板处理。然后，分析了视图和模型绑定以及刷新网页的方法。AngularJS，Ember和Knockout在模型更改时提供自动刷新视图，这与Backbone恰恰相反，每次我们想要刷新View时都必须调用render函数，这会给程序员留下很多控制权。

研究中遇到的一个问题是使用Backbone时内存泄漏的问题。因此，如果用户对JavaScript不够熟悉，最好选择其余三个框架中的一个。

表4包含以下比较标准：技术文档和Web浏览器支持，或与移动平台上某些Web浏览器的兼容性。技术文档以1（最差）到5（最佳）的比例显示，其中评估包括这些框架的官方网站的文档，具有w3schools等官方教程的站点，以及活跃用户支持的官方论坛。

对本文中学习技术所需的时间进行了分析，结果如表5所示。熟悉该框架的最短时间是使用Knockout，最长的是Backbone。

AngularJS是分析框架中最复杂的，但它有大量优秀的教程，使学习更容易。 此外，还分析了这些JavaScript框架的语法。 AngularJS语法是最简单的，因为它需要最少的额外代码，Knockout稍微困难一些，而Backbone和Ember需要大量额外的代码才能编写。评级如表V所示，从1（最差）到5（最佳）。 表五还显示了一些使用这些框架的知名客户公司，根据这一标准，Backbone最为突出。

1. 结论

在本研究论文中，给出了基于JavaScript技术分析的当前顶级框架。 影响框架选择的因素很多，主要是需要知道Web应用程序结构以及应该从框架中获得什么样的帮助。分析显示Backbone与AngularJS，Ember和Knockout有很大不同，因为它给开发人员留下了很多决策。剩下的三个框架提供了更多的完成功能，并且自己做了很多决策。AngularJS，Ember和Knockout提供的功能包括自动内容更新，通过程序进行数据管理，基于特定标准的相对简单的内容过滤，列表排序等。所有四个框架都将程序结构分成不同的部分，因此创建了一个更易读的代码，这对于大型Web系统的开发非常重要。作业划分为明确定义的部分由其架构提供，该架构基于MVC的变化。每个框架也有一些不好的方面，这些方面与应用程序安全性，测试，执行速度和技术文档有关。