“智慧政务”中的文本挖掘应用

摘要：

本文利用自然语言处理和文本挖掘的方法，提出了本赛题中三个题目的解决方法。

针对问题一，我们使用一种自监督学习方法，首先使用标签数据提炼词典，然后在无标签数据（待分类留言）中进行反馈更新词典，最终确定留言分类。

具体来说，该过程是基于词典的迭代。首先，我们使用带标签数据，我们将留言进行分割得到词汇列表，然后计算词汇权重，得到词典，词典记录了词汇及其权重，共有 13 种一级分类，每一类对应一个词典。词典用于对未分类留言进行分类，然后从分类的留言中找到更多词汇，并更新词典，新的词典有助于对更多留言进行分类，通过该迭代过程，逐步更新（通常是放大）已经得到的词典。在迭代过程中，为了防止得到的词典使得结果偏向于特定的某一类，提出一种“类别控制”方法，该方法对留言进行排名，保证每次迭代中保持相同数量的排名靠前的不同类别的留言。而那些由于“类别控制”产生的未分类留言，我们直接使用词典进行判定，不需要进行迭代。

针对问题二，我们首先使用相关的命名实体识别算法，将留言中的地点、人物进行识别，将同一地点、同一类人发表的留言划分到同一簇中，将留言进行初步归类。然后我们以单独的一个簇作为研究对象，使用 TF-IDF 算法计算语句间的相似度，利用密度聚类将留言进一步分类，得到热点问题。最后，我们考虑单位时间内同类留言发表频率、留言发表时间、留言的点赞数、反对数等方面，提出了热度评价公式。

针对问题三，我们通过语言学规律和实地验证，总结了一条前提和两条规律。前提 1:用户的留言具有很好的可解释性；规律 1:词汇的出现顺序表征了答复意见的可解释性；规律 2:根据前提 1，用户留言和官方答复的相似词汇越多，相关性和完整性越高。基于上述两条规律和一条前提，我们首先使用 TextRank 算法提取关键词作为用户留言和答复意见的特征项，然后使用 TF-IDF 计算词汇权重，并根据关键词在留言中的出现顺序将关键词进行排序，得到一个关于词汇的序列。针对序列，我们使用 cosine 根据关键词的 TF-IDF 权重得到用户留言和官方回复的相似度，并以此作为答复意见的评价指标，相似度越高，答复意见的相关性、完整性、可解释性更高。

关键词：自监督，TF-IDF，命名实体识别，TextRank，密度聚类

目录

1.问题分析.....3

2.群众留言分类.....3

1.1 步骤一：词典初始化.....4
1.2 步骤二：判定留言分类.....4
1.3 步骤三：更新词典.....4
1.4 类别控制.....4
1.5 迭代控制.....5

3.群众留言分类.....5

3.1 留言初步分类.....5
3.2 热点问题识别.....6
3.3 热度评价公式.....6

$S_0 = \frac{C}{T}$.....6

$S_{user} = \log_2(z+1)$.....6

$z = \begin{cases} 0, x < 0 \\ 1, x = 0 \\ |x|, x > 0 \end{cases}$.....7

4.答复意见的评价.....7

5.实验结果.....7

6.参考文献.....9

1.问题分析

问题一属于多分类问题。在自然语言处理中，针对二分类问题，有一类方法叫做基于词典的方法，它的根本思想就是某个词汇更偏向于某一类。而在留言中也是如此，某个词汇并不单独属于某个类别，但是词汇肯定会更偏向于某一类。基于此，我们通过标签数据提取词汇对于每个类别的影响权重，并根据这些词汇及其影响权重确定了留言的类别。

问题二，我们需要找出留言的热点问题，给出热度评价指标，并提取地点/人群。提取地点/人群，就用到了命名实体识别相关的算法。其次，同一地点/人群可能对应着不同的事件。基于上述，所以我们首先将同一地点/人群的留言划分为同一类别。然后针对同一类别的留言，使用密度聚类算法进行聚类，最终得到热点问题。密度聚类不需要指定聚类的个数，非常适合本问题。

问题三，我们需要从相关性、完整性、可解释性等角度给出答复意见的评价。我们的出发点依旧在词汇上，我们通过总结规律得出，词汇的主题影响着答复意见的相关性和完整性，词汇的顺序影响着答复意见的可解释性。而为了解决这个问题，我们还需要一个“对照”语句，于是我们默认用户的留言具有很好的可解释性，并将答复意见与用户留言进行对比，最终给出答复意见的评价。

下述内容中，我们在第二节给出了问题一的解决方案，第三节给出了问题二的解决方案，第四节给出了问题三的解决方案，第五节展示了实验结果，第六节列出了参考文献。

2.群众留言分类

针对群众留言分类，我们使用一种自监督[1,2]学习方法，首先使用标签数据提炼词典，然后在无标签数据（待分类留言）中进行反馈更新词典，最终确定留言分类。

具体来说，该过程是基于词典的迭代。首先，我们使用带标签数据，我们将留言进行分割得到词汇列表，然后计算词汇权重，得到词典，词典记录了词汇及其权重，共有 13 种一级分类，每一类对应一个词典。词典用于对未分类留言进行分类，然后从分类的留言中找到更多词汇，并更新词典，新的词典有助于对更多留言进行分类，通过该迭代过程，逐步更新（通常是放大）已经得到的词典。在迭代过程中，为了防止得到的词典使得结果偏向于特定的某一类，提出一种“类别控制”方法，该方法对留言进行排名，保证每次迭代中保持相同数量的排名靠前的不同类别的留言。而那些由于“类别控制”产生的未分类留言，我们直接使用词典进行判定，不需要进行迭代。

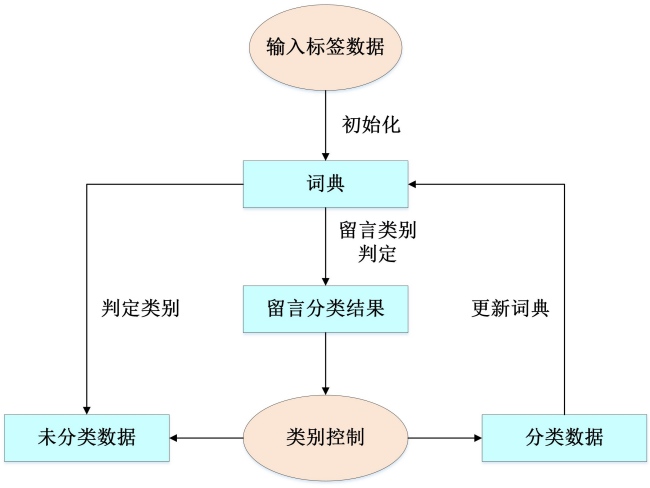

图 1. 问题一的总体结构图

1.1 步骤一：词典初始化

这一节中，我们将留言分割为一个个词汇，并将词汇添加到词典中，并为词汇设置权重。这里一级分类有 13 种，那么通过这一节，我们可以得到 13 个词典，在编写代码时重复声明 13 个相同结构的词典明显不符合编码规范，所以我们在编码实现时创建一个列表对象，称为词典列表 $List_{Dic}$，而 $List_{Dic}$ 中包含了 13 个词典。步骤如下：

(1).关键词提取。操作对象为一条留言 *Mess*，将 *Mess* 使用 TextRank 算法提取关键词得到词汇列表 *Vocs*，并去除停用词。

(2).更新词典。如果留言 *Mess* 属于类别 *A*，将词汇列表 *Vocs* 中每条词汇加入到词典 D_A 中，并将 D_A 中该词汇的权重+1，如果 D_A 中没有该词汇，则将词汇加入词典并设置权重为 1。

(3).同比缩减、统一补偿。为了防止某个词汇权重过大，影响结果，更新词典过程中，对词典 D_A 中每个词汇的权重乘以一个比率因子β(0<β<1，经验值在 0.5 左右)，然后词典中每个词汇的权重加上$\frac{1-\beta}{len(D_A)} \times sum(D_A)$，其中 $len(D_A)$表示词典D_A的长度，$sum(D_A)$表示D_A中未进行缩减时的所有词汇的权重之和。

(4).循环迭代。循环迭代直到遍历标签数据（即附件 2）完成。

在这里，我们需要保证标签数据中不同类别的留言数量相等，这是为了防止我们得到的词典更加偏向于某一类。比如“城乡建设”类具有 50 条留言，“党务政务”类具有 10 条留言，那么我们得到的词典中，“城乡建设”类对应的词典中词汇的权重会明显高于“党务政务”类对应的词典中词汇的权重，这对我们的结果产生了很大的影响。

1.2 步骤二：判定留言分类

在这一节中，我们会使用上一节中得到的词典确定留言的类别。

首先，对于一条未分类留言 $Mess_{un}$，将其分割为一个词汇列表 $Vocs_{un}$，去除停用词。

然后，对于 $Vocs_{un}$ 中的一个词汇 *Voc*。如果词汇 *Voc* 属于词典 D_A，则留言 $Mess_{un}$ 对类别 *A* 的得分 $S_A=S_A+D_A[Voc]$，其中 S_A 默认为 0，$D_A[Voc]$是词典 D_A 中 *Voc* 对应的权重。注意，词汇 *Voc* 可能属于多个类别，被记录在多个词典中，这时，需要将每个对应类别的得分均加上对应的权重。如果词汇 *Voc* 不属于任何词典，则跳过不予处理。

经过上述操作，得到了留言 $Mess_{un}$ 关于 13 个类别的 13 个得分，我们将 $Mess_{un}$ 归类为最高得分对应的类别，并记录下最高得分。

1.3 步骤三：更新词典

这一节，我们要通过上一节确定的留言分类来更新词典，以达到反馈更新的目的。

在这里，我们定义一个指标用来评价留言是否可用来更新词典。在步骤二中，我们得到了留言关于 13 个类别的 13 个得分，我们将其中最高得分命名为S_{max}，次高得分命名为S_{next}，那么判定指标为：

$$F\_norm = \frac{S_{max} - S_{next}}{S_{next}}$$

只有当S_{max}和S_{next}差异特别大时，即 F_norm 值特别大时，我们才将该留言用来更新词典。在实验中，我们设置 F_norm 为 1.

1.4 类别控制

在步骤二中，判定得到的不同类别的留言数量通常会不同，就像在步骤一种说的那样，这可能会导致在步骤三更新词典后使得我们得到的词典更加偏向于某一类，所以，我们在这

里提出一种比率控制的算法，保证进行步骤三更新词典时使用的不同类别的留言数量相同。

算法 1：类别控制

输入：留言分类结果 *Result*

输出：留言及其类别（不同类别留言数量相同）

主要步骤：

1. C_{min} = *Result* 中数量最少的留言类型对应的留言数量
2. 分别将 *Result* 中每一类按照步骤二中记录下的最高得分按降序对所有留言进行排序
3. 每一类选取排名前 C_{min} 的留言，得到 $Result_{filter}$，其他留言保持未分类
4. Return $Result_{filter}$

1.5 迭代控制

在步骤二和步骤三之间迭代，设置一个参数 N 作为迭代次数，经验值通常在 5 左右。

一般来说，因为迭代过程中存在的误差会随着迭代进行不断的向后传播，导致误差越来越大，所以说当标记的数据越多，标记数据出现错误的个数也会越多，那么随着迭代进行，这个误差会不断扩大。因此，迭代应该在迭代的尽量的早期阶段完成。然而，迭代也不能太早完成，这会导致词典没有得到足够的反馈更新。

当迭代结束后，关于算法 1 中产生的未分类留言，我们使用迭代完成得到的词典，使用步骤二直接判定留言分类。

3.群众留言分类

针对热点问题挖掘，我们首先使用相关的命名实体识别算法，将留言中的地点、人物进行识别，将同一地点、同一类人发表的留言划分到同一簇中，将留言进行初步归类。然后我们以单独的一个簇作为研究对象，使用 TF-IDF[3, 4, 5]算法计算语句间的相似度，利用密度聚类将留言进一步分类，得到热点问题。最后，我们考虑单位时间内同类留言发表频率、留言发表时间、留言的点赞数、反对数等方面，提出了热度评价公式。

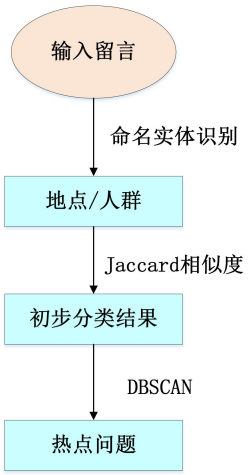

图 2. 问题二的总体结构图

3.1 留言初步分类

在这一步中，我们根据留言中的地点、人物将留言进行初步分类，缩小研究范围。具体步骤如下：

(1).命名实体识别[6]。jieba 库是一个广泛被大家认可的第三方库，它的效果被大家认可

的。我们利用 jieba 库的词性标注函数进行命名实体识别。jieba 库的词性方法是基于正则表达式匹配和隐马尔科夫模型[7, 8, 9]的。

(2).词汇拼接。我们在第一步中通过命名实体识别方法得到了地点和人物的词汇，我们需要将这些词汇拼接为短语。我们遵循“先出现，先拼接”的原则。例如“A5 区劳动东路魅力之城小区一楼的夜宵摊严重污染附近的空气”，我们可以得到三个地点词汇“A5 区”、“劳动东路”“魅力之城小区”，按照这些地点词汇的出现顺序，“A5 区”“劳动东路”“魅力之城小区”，所以我们得到短语“A5 区劳动东路魅力之城小区”。

(3).留言初步分类。我们通过词汇拼接得到了地点和人群，那么我们需要将同一地点、同一人群划分到同一簇中，而相同实体的不同表达方式对我们的结果造成了很大的影响。例如“魅力之城”和“魅力之城小区”属于同一实体，但是表达方式却是不同。所以，我们使用 Jaccard 系数[10]来解决这个问题:

$$Jaccard(X,Y) = \frac{X \cap Y}{X \cup Y}$$

其中，X 和 Y 分别为字的集合，当 $Jaccard(X,Y)$的值大于某个阈值时，我们判定为同一个实体，在实验中，阈值的经验值通常在 0.5 左右。对于上述例子，$X = \{$魅,力,之,城$\}$，$Y = \{$魅,力,之,城,小,区$\}$，$Jaccard(X,Y) = 2/3$。

3.2 热点问题识别

在 2.1 节中，我们将同一地点、同一人群的留言划分到同一簇中。在本节中，我们以单独的一个簇作为研究对象，使用 TF-IDF 算法计算留言的相似度，利用密度聚类算法 DBSCAN 将留言簇进一步细分。密度聚类中，留言间的距离即为 1-sim，sim 为留言间的相似度。DBSCAN 算法需要输入两个参数，一个是含义为ε-邻域的参数ε，另一个是含义为满足要求的核心对象的ε-邻域包含的最小对象数目 MinPts。在实验中，ε的经验值为 0.7，MinPts 的经验值为 3。

3.3 热度评价公式

针对热度评价公式，我们考虑二个方面的因素。一是留言的初始热度(S0)，初始热度可以大致概括为在单位时间内针对同一问题用户发表的留言数。留言数越多，热度越高。二是用户对留言的评价(主要表现在留言的点赞数、反对数)，评价越高，留言的影响和可信度应当越高，我们将其总结为用户交互热度(S_{user})。详细描述如下。

(1).初始热度(S_0)。热点问题是群众集中反映的某一问题，因此集中反映的问题热度应该比其他事件高，我们采用单位时间内该问题出现的留言条数来表示，确定初始热度值，即：

$$S_0 = \frac{C}{T}$$

其中：

- C 为热点问题对应的留言总条数。
- T 为热点问题中所有留言的时间跨度，以月为单位。

(2).用户交互热度(S_{user})。根据留言数据，用户交互即为点赞数和反对数，我们考虑到留言为群众反映问题的途径，用户反对留言可能是留言用户提出问题不符合实际情况或是提出不符合群众期望的建议，因此反对数可当做降低问题热度值的因素，同时考虑数据量大，用户数量多，点赞数和反对数可从 0 到数千不等，因此我们可适当弱化点赞数和反对数对热度值的影响。参考 *Reddit* 排名算法得到用户交互热度值计算方法为：

$$S_{user} = \log_2(z+1)$$

x 代表点赞数和反对数之差，则 z 为：

$$z = \begin{cases} 0, x < 0 \\ 1, x = 0 \\ |x|, x > 0 \end{cases}$$

如此设置保证了当更多的用户反对该条留言时，该条留言的用户交互热度为 0。对 z 取对数弱化了其对热度的影响。

(3).热度评价公式。初始热度考虑的是单位时间内用户发表的留言条数，而用户交互热度考虑的是大众对于该条留言的认可程度，所以我们可以用户交互热度作为初始热度的加权，所以我们可以得到该条留言的总热度：

$$S_i = S_{user} * S_0$$

若 n 代表热点问题中包含的留言条数，那么热点问题的评价公式 *Score* 为：

$$Score = \frac{\sum_{i=1}^{n} S_i}{n}$$

4.答复意见的评价

针对答复意见的相关性和完整性，首先，我们需要提取关键词来作为留言的特征项，我们选择 TextRank[11]算法提取用户留言和官方回复的关键词，并使用 TF-IDF 计算词汇的权重。我们通过语言学规律和实践验证，总结了一条前提和两条规律。

前提 1：用户的留言具有很好的可解释性。根据这条前提，用户的留言和官方回复的相似度越高，代表官方回复的可解释性越高。

规律 1：词汇的出现顺序表征了答复意见的可解释性。按照中文语言特征来说，比如“为了完善住宅使用功能”这句话，先出现“住宅”关键词，再出现“功能”关键词，这是符合语言特征的。

规律 2：根据前提 1，用户留言和官方答复的相似词汇越多，相关性和完整性越高。

所以，根据上述两条规律和一条前提，针对某条留言或答复意见，我们根据关键词在留言中的出现顺序将关键词进行排序，得到一个关于词汇的序列。最终我们使用 cosine 相似度，根据关键词的 TF-IDF 权重得到序列的相似度 $sim(sim \in [0,1])$，并以此作为答复意见的评价指标，相似度越高，答复意见的相关性、完整性、可解释性更高。

为了方便评价，我们将答复意见的评价指标映射到特定区间内，所以最终的评价指标为：

$$S = e^{sim}$$

可以看到 $sim \in [0,1]$，所以 $S \in [1,e]$。

5.实验结果

我们在问题一的解决方案中，存在一个参数β($\beta \in [0,1]$)，下面给出了β对分类结果的影响，我们的评价指标是赛题中要求的 F-score。

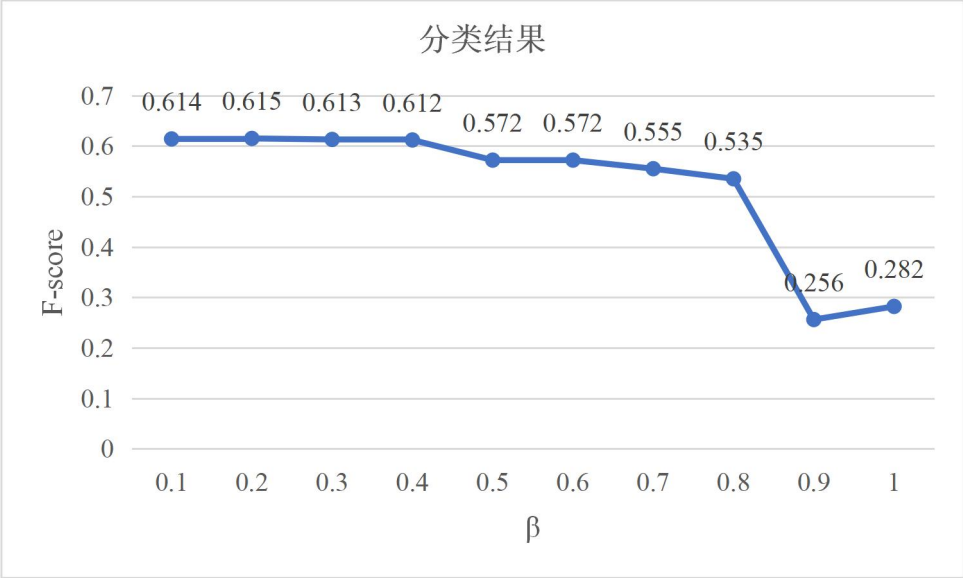

图 3. 问题一实验结果

可以看到，当β较小时，可以取得相对较好的结果。当$\beta = 0.9$ 和$\beta = 1$ 时，其结果是远远差于其他结果，这也验证了我们提出的“同比缩减，统一补偿”的有效性。

下面展示的是问题二的实验结果，我们设置ε=0.7，MinPts=3。我们给出了 Top-5 的热点问题。

表 1.问题二实验结果

热度排名	问题 ID	热度指数	时间范围	地点/人群	问题描述
1	1	2.32	2019/08/09 至 2019/08/16	A 市经开区泉塘	A 市经开区泉星公园项目规划需优化
2	2	2.0	2019/05/30 至 2019/07/16	楚郡未来实验学校	楚郡未来实验学校针对四年级学生上调学费是否违规
3	3	1.73	2019/03/01 至 2019/01/08	A 市	请 A 市加快轨道交通建设力度
4	4	1.58	2019-05-28 至 2019/05/26	A 市	咨询 A 市办理户口迁出可以在网上申请办理吗？
5	5	1.58	2019/07/22 至 2019/07/18	A7 县	A7 县新国道 107 距我家仅 3 米，相关政府部门为何不同意拆迁？

针对第三题，留言的答复意见的篇幅过长，在本文中难以做一个完整的展示，所以我们只将实验得到的评分 top-2 展示出来。

表 2.问题三实验结果

排名：1　　**评分**：2.32 **留言**：请问：”2017 年西地省人力资源和社会保障厅关于将 36 种药品纳入西地省基本医疗保险、工伤保险和生育保险药品目录乙类范围的通知”是否对 A 市医保人员不起作用，难道该文件还有等级之分？ **答复意见**：网友“UU0082385”您好！您的留言已收悉。现将有关情况回复如下：《关于将

36 种药品纳入西地省基本医疗保险、工伤保险和生育保险药品目录乙类范围的通知》（[政府发文]57 号）将利拉鲁肽注射剂等 36 种药品纳入西地省基本医疗保险、工伤保险和生育保险药品目录乙类范围，从 2017 年 9 月 1 日起执行。A 市医疗保险执行的是西地省的药品目录，这 36 个药同样也纳入了 A 市医保的报销范围。只要是 A 市医疗保险正常参保人，在 A 市基本医疗保险协议医疗机构住院治疗，而该院有这些药品，根据病情需要使用该药品的都可以正常报销，没有任何等级之分。感谢您对我们工作的支持、理解与监督！2018 年 10 月 25 日
排名：2　　**评分**：2.27 **留言**：领导您好，　　关于（关于实施差别化购房措施的通知）中，签订拆迁安置协议一年内的属于刚需群体，是否签订拆迁协议一年后还未购房的就不属于刚需群体了。请领导解答，谢谢。 **答复意见**：网友“UU0081211”您好！您的留言已收悉。现将有关情况回复如下：根据《关于实施差别化购房措施的通知》，签订拆迁安置协议一年内的属于刚需群体，签订拆迁协议一年后还未购房的不属于刚需群体。感谢您对我们工作的支持、理解与监督！2018 年 4 月 2 日

6.参考文献

[1]. Qiu, Likun, et al. "Selc: a self-supervised model for sentiment classification." Proceedings of the 18th ACM conference on Information and knowledge management. 2009.

[2]. Zhang, Weishi, et al. "SESS: A self-supervised and syntax-based method for sentiment classification." Proceedings of the 23rd Pacific Asia Conference on Language, Information and Computation, Volume 2. 2009.

[3]. Ramos, Juan. "Using tf-idf to determine word relevance in document queries." Proceedings of the first instructional conference on machine learning. Vol. 242. 2003.

[4]. Aizawa, Akiko. "An information-theoretic perspective of tf–idf measures." Information Processing & Management 39.1 (2003): 45-65.

[5]. Wu, Ho Chung, et al. "Interpreting tf-idf term weights as making relevance decisions." ACM Transactions on Information Systems (TOIS) 26.3 (2008): 1-37.

[6]. Nadeau, David, and Satoshi Sekine. "A survey of named entity recognition and classification." Lingvisticae Investigationes 30.1 (2007): 3-26.

[7]. Krogh, Anders, et al. "Predicting transmembrane protein topology with a hidden Markov model: application to complete genomes." Journal of molecular biology 305.3 (2001): 567-580.

[8]. Beal, Matthew J., Zoubin Ghahramani, and Carl E. Rasmussen. "The infinite hidden Markov model." Advances in neural information processing systems. 2002.

[9]. Fine, Shai, Yoram Singer, and Naftali Tishby. "The hierarchical hidden Markov model: Analysis and applications." Machine learning 32.1 (1998): 41-62.

[10]. Niwattanakul, Suphakit, et al. "Using of Jaccard coefficient for keywords similarity." Proceedings of the international multiconference of engineers and computer scientists. Vol. 1. No. 6. 2013.

[11]. Mihalcea, Rada, and Paul Tarau. "Textrank: Bringing order into text." Proceedings of the 2004 conference on empirical methods in natural language processing. 2004.