Background:

According to the our past working experience in newspaper, journalists actually also rely on googling online to get news facts that they need because there no single resource channel allowing them get information faster. However, it’s always time-consuming and exhausting when they do this since multiple web links show up everytime which they have to look through from top to bottom. And also, when a determiner happen to be “hot words” in other news, like "iphone 6 screen” in “producer of iphone 6 screen”, most websites returned are of none value answer our specific question. Therefore, the current situation is, even for a simple topic as “Will Japanese Prime Minister step down this year, according to historical in office period data?”, the data collection part could use whole 3 days for two people, while some information could still be incomplete, which seriously impair the timeliness of news articles.

Problem Overview:

While there’s no one solving this problem for journalists now, we aim to train a QA system allowing them get the accurate facts that they need for articles quickly.

The typical usage scenarios would be: 1. “苹果手机屏幕生产厂家”—“which company produces iPhone screen”; We just want the name of the company here, but “苹果手机屏幕” is such a hot word especially when it comes to customer service, so journalist have to find the right answer from news about complaints of its poor quality. 2. “安徽首例非典确诊时间”—“Time of the first case of SARs diagnosed in Anhui”, we just want the date here, but “first case of SARs” appear in so many news which most of them don’t give any valuable informations of the date we want. In both cases, we will be forced to shatter unrelated information to get the answer we want manually, which costs plenty of time.

state-of-art/Related work:

<http://dspace.mit.edu/bitstream/handle/1721.1/33141/62242506-MIT.pdf?sequence=2>

基于句法分析和答案分类的中文问答系统

technical approaches:

数据集获取:

爬虫

问题处理：

word segmentation(斯坦福), syntactic parsing/POS tagging(抽取关键词), 分类成who/which/where, when的两类（怎么实现？？），新闻本身有tag有分类，统计词频，在问题匹配时反向索引就知道属于哪个tag下面，然后再精确匹配。

答案处理：

我们还需要一个stop-word list(我有一个词性词频的数据集，出现次数最多的可以当做stop words删掉);

passage/段落 retrival(包含关键词的段落/前后两句抽出来)（regular expression based hard matching）(需要一个syntactic regular expression library，或者拿那个问题集里面的事实类出来训练？文章里举的英文library的例子能够帮助区分noun之间的意义，帮助应该很大，可以有没有中文的？)<http://dspace.mit.edu/bitstream/handle/1721.1/33141/62242506-MIT.pdf?sequence=2>

syntactic pattern—我们基本上就是要提取noun phrases, 主要有两种：copulas, noun phrase modifying the question target;

Name entity tagger—针对who/which/what, where, when问题的答案 (这俩是不是有点重复？俩都做是不是效果更好？还是不好？)

answer ranking criteria最后只产出一个答案: （怎么做？）<https://www.researchgate.net/publication/30995011_A_Definitional_Question_Answering_System_Based_on_Phrase_Extraction_Using_Syntactic_Patterns>

Evaluation:

问题的test:

相同问题set的答案的test:

Reference: