11-791: 11-791/693: Design and Engineering of Intelligent Information System Fall 2013

Team1: Pallavi Baljekar, Ran Chen, Jung In Lee, Jerry Sun, Wei Zhang

November 4, 2013

Abstract

The main goal of this project is to improve the given pipeline, using statistical methods and machine learning models.

1 GitHub repository

https://github.com/chenran818/hw5-team01

2 Initial Divisions of Work Among Team Members

Pallavi Baljekar: Presentation.
Ran Chen: leader, implementation.
Jung In Lee: Report write-up.
Jerry Sun: Implementation.
Wei Zhang: Brain-storming of the design, implementation.

3 Initial Pipeline & Initial Type System Design

Our initial pipeline will very much resemble the given pipeline. For the first part of this project, we will be mainly focusing on improving the AnswerChoiceScoring strategy. Instead of using the Point-wise Mutual Information (PMI), we will explore some other methods, namely NPMI, MI, NMI, and some combinations of such, to better optimize the task.

Since we are maintaining most structures in the pipeline as it is given, no changes in the Type System design will be necessary.

4 Baseline Methods to Implement

Improving Point-wise Mutual Information (PMI):
PMI is a measure that captures the correlation between two vectors x and y. This, however, becomes infinitely large when x or y is rare. In reaction to this problem, we examine the following

methods, as proposed in (Gerlof 2009).

1. **NPMI**

$$NPMI(X,Y) = \frac{\ln \frac{p(x,y)}{p(x)p(y)}}{-\ln p(x,y)}$$

Normalizing PMI can be seen as reducing the impact of low frequency on ranking.
Also, unlike PMI, NPMI has nice lower and upper bounds. When two words only occur together, $PMI = 1$; when two words only occur separetly, $PMI = -1$; when they are independent, then $PMI = 0$. This is useful because, we want to compute a cumulative PMI for each answer choice by aggregating the PMIs between word pairs across the query and candidate sentences, and adding measures with bounds are much more practical than those that can reach $\pm\infty$.

2. **MI & NMI**

$$MI(X,Y) = \sum_{x,y} p(x,y) * PMI(x,y)$$

Likewise, we normalize MI by the joint entropy of X and Y.

$$NMI(X,Y) = \frac{\sum_{x,y} p(x,y) * PMI(x,y)}{-\sum_{x,y} p(x,y) \ln p(x,y)}$$

MI is largely different from PMI in that it is better at informing whether X and Y are unrelated or not, but not so good at informing how much they are related. This, however, can still be useful for determining whether two words are correlated or not, and in particular, can be used in accordance with NMI, which is reported to behave more like NPMI or PMI, and completely distinct from MI (Gerlof 2009). It is often recommended that MI and NMI be used together.

3. **Plan for Baseline:** Modify the PMI formula and re-evaluate the pipeline, using the methods proposed above. Examine whether these methods improve the performance.

4. **Future Plans:** Find some other distinguished features, e.g. ML features, overlap of name entities, cosine similarity, semantic role labeling, etc.
Using Machine Learning, learn the coefficients for the best combination of the proposed features that would optimize the task.

References

[1] Evert, S. (2004/2005) The Statistics of Word Cooccurrences: Word Pairs and Collocations. PhD thesis, IMS Stuttgart.

[2] Gerlof, B. (2009) Normalized (Pointwise) Mutual Information in Collocation Extraction. *In: Chiarcos, Eckart de Castilho Stede (eds), Von der Form zur Bedeutung: Texte automatisch verarbeiten / From Form to Meaning: Processing Texts Automatically, Proceedings of the Biennial GSCL Conference 2009*, pp3140, Tbingen, Gunter Narr Verlag.