Simulating signal and noise queries for score normalization in distributed IR

Arampatzis, A.; Kamps, J.

Publication date
2010

Document Version
Author accepted manuscript

Published in
Proceedings of the 10th Dutch-Belgian Information Retrieval Workshop (DIR 2010)

Citation for published version (APA):
Simulating Signal and Noise Queries for Score Normalization in Distributed IR

Avi Arampatzis$^1$  
Jaap Kamps$^{2,3}$

$^1$ Electrical and Computer Engineering, Democritus University of Thrace, Greece  
$^2$ Archives and Information Studies, University of Amsterdam  
$^3$ ISLA, Informatics Institute, University of Amsterdam  

avi@ee.duth.gr  
kamps@uva.nl

ABSTRACT

Score normalization is indispensable in distributed retrieval and fusion or meta-search where merging of result-lists is required. Distributional approaches to score normalization with reference to relevance, such as binary mixture models like the normal-exponential, suffer from lack of universality and troublesome parameter estimation especially under sparse relevance. We develop a new approach which tackles both problems by using aggregate score distributions without reference to relevance, and is suitable for uncooperative engines. The method is based on the assumption that scores produced by engines consist of a signal and a noise component which can both be approximated by submitting well-defined sets of artificial queries to each engine. We evaluate in a standard distributed retrieval testbed and show that the signal-to-noise approach yields better results than other distributional methods.

1. INTRODUCTION

Modern best-match retrieval models calculate some kind of score per collection item which serves as a measure of the degree of relevance to an input request. Scores are used in ranking retrieved items. Their range and distribution varies wildly across different models making them incomparable across different engines [4], even across different requests on the same engine if they are influenced by non-semantic query characteristics, e.g., length. Even most probabilistic models do not calculate the probability of relevance of items directly, but some order-preserving (monotone or isotone) function of it.

The main aim of this paper is to analyse and further develop score distributional approaches to score normalization. Our underlying assumption is that normalization methods that take the shape of the SD into account will be more effective than methods that ignore it. We want to make no assumptions on the search engines generating the scores to be normalized other than that they produce ranked lists sorted by decreasing score. Thus, we treat each engine as a ‘black-box’ and are interested in approaches based only on observing their input-output characteristics: the queries and resulting score distributions.

2. SINGLE DISTRIBUTION METHODS

Z-score A standard method for score normalization that takes the SD into account is the Z-SCORE. Scores are normalized, per topic and engine, to the number of standard deviations that they are higher (or lower) than the mean score:

$$Z-SCORE: \quad s' = \frac{s - \mu}{\delta}$$

where $\mu$ is the mean score and $\delta$ the standard deviation. Z-SCORE assumes a normal distribution of scores, where the mean would be a meaningful ‘neutral’ score. As it is well-known, actual SDs are highly skewed.

Aggregate Historical CDF Simplified A recent attempt models aggregate SDs of many requests, on per-engine basis, with single distributions [3] using the historical CDF:

$$HIS: \quad s' = P(S_{HIS} \leq s)$$

where $P(S_{HIS} \leq s)$ is the cumulative density function (CDF) of the probability distribution of all scores, and HIS refers to the fact that historical queries are used for aggregating the SD that the random variable $S_{HIS}$ follows. HIS normalizes input scores $s$ to the probability of a historical query scoring at or below $s$.

3. SIGNAL-TO-NOISE METHODS

We investigate the use of dual aggregate SDs. Assuming that scores produced by an engine consist of two components, signal and noise, the score random variable $S$ can be decomposed as:

$$S = S_{signal} + S_{noise}$$

The probability densities of the components are given respectively by $P_{signal}$ and $P_{noise}$ defined across the engine’s output score range. Furthermore, we assume ‘stable’ system characteristics for the engine in the sense that the signal and noise levels at a score depend only on the score. We can define a function which normalizes input scores $s$ into the fraction of the signal at $s$:

$$S/N: \quad s' = \frac{P_{signal}(s)}{P_{signal}(s) + P_{noise}(s)}$$

Since engines are expected to produce increasing signal-to-noise ratios as score increases, this may be an interesting normalization.

However, the magnitude of the original score is not taken into account. An obvious improvement would be to multiply $S/N$ with a calibrated score $s$, for which we could use the HIS normalization:

$$S/N*HIS: \quad s' = \frac{P_{signal}(s)}{P_{signal}(s) + P_{noise}(s)} P(S_{HIS} \leq s)$$


Copyright is held by the author(s).  
DIR-2010 January 25, 2010, Nijmegen, the Netherlands.  
Copyright 2010 by the author(s).
The resulting scores would be comparable across engines, however, the distribution of the variable \( S_{\text{his}} \) depends on the availability of historical queries. Using historical queries, although very feasible and no cooperation is required, may lead to instabilities and biases. To deal with this, we can instead use the variable \( S_{\text{signal}} \):

\[
s' = \frac{p_{\text{signal}}(s)}{p_{\text{signal}}(s) + p_{\text{noise}}(s)} P(S_{\text{signal}} \leq s)
\]

This calibrates \( s \) to the probability of having signal at or below \( s \).

The question is how long those fragments should be. When incorporating dependencies, retrieval models are becoming practically intractable, which led in the past to the infamous independence assumption. Instead of trying to model term probabilities of occurrence and dependencies, we can rather tackle both features at once by picking real text fragments out of a corpus. The remaining question is how long those fragments should be.

Arampatzis and Kamps [1] arrive at a truncated Poisson/Pareto law model of query length. The bulk of queries concentrates at short lengths where a power-law does not fit at all given the current query languages, therefore it makes practical sense to use a truncated mix of Poisson-Zipf to generate query lengths. In such a practical model, the lengths are Poisson-distributed for \( k < k_0 \) while they are Zipf-distributed for \( k \geq k_0 \). The choice of \( k_0 \) depends on the specific domain (i.e., a combination of features of the document collection, query/indexing language, and pattern of use of the system). As a rule of thumb, \( k_0 \) seems to be just above the mean observed query length.

5. EVALUATION: DIR TESTBEDS

Standard score normalization methods like the MinMax ignore the score distribution: \( s' = \frac{s - \min}{\max - \min} \), with \( \min \) (\( \max \)) the minimal (maximal) score per query and engine. That is, MinMax forces all scores in \([0,1]\), resulting in a maximal score per topic and engine. While they are Zipf-distributed for \( k \geq k_0 \), the consistent improvements in \( S_{\text{signal}} \) is also significant better than \( \text{ROUNDROBIN} \), and at least as good as \( Z\)-SCORE. We compare \( H\) against the new signal-to-noise methods \( S/N \), \( S/N+H\), and \( S/N+\text{SIG} \). Table 2 presents the distributed retrieval results without resource selection. Overall, the \( S/N+H\) and \( S/N+\text{SIG} \) runs show significant improvements over the strong baseline of \( H\), while the consistent improvements in \( S/N \) are mostly non-significant.

Acknowledgments This research was supported by the Netherlands Organization for Scientific Research (NWO), CATCH programme, under project number 640.001.501.

REFERENCES