ISE 540
Text Analytics

Mayank Kejriwal

Research Assistant Professor/Research Lead

Department of Industrial and Systems Engineering

Information Sciences Institute

USC Viterbi School of Engineering

kejriwal@isi.edu

Evaluation of ranked retrieval results

- Important to introduce some notation:
  - Q is a set of queries
  - D is a set of documents ('corpus)
  - For query q_j in Q, let the set of 'relevant' documents (this has to be human-annotated!) be $G_j = \{d_1,\ldots,d_{m\_j}\}$
  - We refer to G_j as the 'ground truth' for q_j, with the set of ground truths for all queries typically constituting a 'gold standard' for evaluation purposes
  - Sometimes, when we don't completely trust the gold standard, we refer to it as a 'silver' or even 'bronze' standard to reflect its quality/our faith in in its correctness

Example

- **Query:** why was the stock market so volatile recently
- Let's try this query in Google
- Let's try this query in Bing
- Try to answer the following questions:
  - How do you decide whether a retrieved webpage is relevant or not? Hint: do not overthink…
  - What is the highest ranked relevant entry? (note: a rank of 1 is higher than a rank of 2)
  - Which is better: Google or Bing?

Simplest metric

- Mean Reciprocal Rank (MRR)
- Assumes there is only one relevant document per query
- Suppose the rank of this document in the ranked list is k
- MRR is just 1/k (can be averaged if there is more than one query)
- What is the highest MRR? What is the second-highest…?
  - What does this tell you about MRR?

Normalized Discounted Cumulative Gain (NDCG)

- Stands for Normalized Discounted Cumulative Gain (again, 'normalized' because it lies between 0 and 1)
- Has the advantage that it can work with real-valued 'relevance' scores (when would this arise?)
- Let R(j,d) be the relevance score assessors gave to document d for query q_j

$$\mathrm{NDCG}(Q,k) = \frac{1}{|Q|}\sum_{j=1}^{|Q|} Z_{kj} \sum_{m=1}^{k} \frac{2^{R(j,m)}-1}{\log_2(1+m)}$$

- k is the rank at which the **last** relevant document occurs (same result is obtained if you put k=|D|)
- Z_{kj} is the 'normalization factor' to ensure that a 'perfect ranking' (which is what?) would yield NDCG of 1
  - Requires a separate calculation, but is necessary

This is a complicated formula…

- …So we'll do it in steps
- For each query q_j, we need to compute:

$$Z_{kj} \sum_{m=1}^{k} \frac{2^{R(j,m)} - 1}{\log_2(1+m)}$$

- Currently, we assume that R=1 if document at rank m is relevant for query q_j (i.e. is in the ground truth) and 0 otherwise
- The expression inside the sum evaluates to ____ for irrelevant entries in the ranked list and _____ for relevant entries in the ranked list

Abstract example

- Suppose we have the following list for query q and the ground-truth contains entries d7 and d5
- What does the ‘sum’ in the previous slide evaluate to?

d1
d2
d3
d4
d5
d6
d7
d8

Abstract example

- Suppose we have the following list for query q and the ground-truth contains entries d7 and d5
- What is an ideal ranking?

d1
d2
d3
d4
d5
d6
d7
d8

Abstract example

- Suppose we have the following list for query q and the ground-truth contains entries d7 and d5
- What is an ideal ranking?
- Now what does the 'sum' or *Discounted Cumulative Gain (DCG)* evaluate to?

d5
d7
d3
d4
d1
d6
d2
d8

Abstract example

- Suppose we have the following list for query q and the ground-truth contains entries d7 and d5
- What is an ideal ranking?
- Now what does the 'sum' evaluate to?
- Recall that we want the 'ideal' NDCG or IDCG to be 1…hence, we need the normalization factor Z

d5
d7
d3
d4
d1
d6
d2
d8

This is a complicated formula…

- …So we'll do it in steps
- For each query q_j, we need to compute:

$$Z_{kj} \sum_{m=1}^{k} \frac{2^{R(j,m)} - 1}{\log_2(1+m)}$$

- Currently, we assume that R=1 if document at rank m is relevant for query q_j (i.e. is in the ground truth) and 0 otherwise
- For this expression to be 1 for the ideal ranking, what should Z be?

Abstract example

- Suppose we have the following list for query q and the ground-truth contains entries d7 and d5
- Given the value of Z you computed earlier, what is the NDCG for the original ranking?

d1
d2
d3
d4
d5
d6
d7
d8

Z is NOT equal to Sum_log(1+m)
1/log(x)+1/log(y) = DCG
1/log(x)+1/log(y)…=log(y)(z)…/log(x)lo
g(y)log(z)…+log(x)log(z)…/
log(x)log(y)log(z)
1/DCG =
log(x)log(y)log(z)…/log(y)log(z) +
1/DCG=1/(1/log(x)+1/log(y)…)

Now let's return to the original formula

- Stands for Normalized Discounted Cumulative Gain (again, 'normalized' because it lies between 0 and 1)
- Has the advantage that it can work with real-valued 'relevance' scores (when would this arise?)
- Let R(j,d) be the relevance score assessors gave to document d for query q_j

$$\mathrm{NDCG}(Q,k) = \frac{1}{|Q|}\sum_{j=1}^{|Q|} Z_{kj} \sum_{m=1}^{k} \frac{2^{R(j,m)}-1}{\log_2(1+m)}$$

- k is the rank at which the **last** relevant document occurs (same result is obtained if you put k=|D|)
- Z_{kj} is the 'normalization factor' to ensure that a 'perfect ranking' (which is what?) would yield NDCG of 1
  - Requires a separate calculation, but is necessary

Question: why is Z ‘inside’ the outer sum?

Mean average precision (MAP)

- Just like NDCG, formula looks complicated when you first look at it, make sure you understand it completely!
- Let R_{jk} be the set of ranked retrieval results from the top result until we get to document d_k
  - Notice the index j, this tells you the ranked list was in response to query q_j
  - m_j is the number of relevant documents in the ground truth of query q_j
  - You don't need to remember any of this by heart, but you must understand what it all means

$$\text{MAP}(Q) = \frac{1}{|Q|} \sum_{j=1}^{|Q|} \frac{1}{m_j} \sum_{k=1}^{m_j} \text{Precision}(R_{jk})$$

- The 'average' refers to averaging over 'precisions' (notice what 'k' ranges over, this is not trivial!)
- The 'mean' is easier to understand, simply the average over all queries
- Each AP (and hence, MAP) is always between 0 and 1

Mean average precision (MAP)

- Once again, let's try to do this in steps

$$\mathrm{MAP}(Q) = \frac{1}{|Q|} \sum_{j=1}^{|Q|} \frac{1}{m_j} \sum_{k=1}^{m_j} \mathrm{Precision}(R_{jk})$$

- We'll assume a single query q, allowing us to ignore the outer sum for the moment
- Let's return to our previous or 'running' example, but first, don't forget the definition(s) of precision

$$P = tp/(tp + fp)$$

$$\mathrm{Precision} = \frac{\#(\text{relevant items retrieved})}{\#(\text{retrieved items})} = P(\text{relevant}|\text{retrieved})$$

Abstract example

- Suppose we have the following list for query q and the ground-truth contains entries d7 and d5
- What is Precision(R_1)? Note we can ignore sub-index j since we only have one query for the moment

d1
d2
d3
d4
d5
d6
d7
d8

Abstract example

- Suppose we have the following list for query q and the ground-truth contains entries d7 and d5
- What is Precision(R_1)? Note we can ignore sub-index j since we only have one query for the moment
- What is the first rank r at which Precision(R_r) becomes non-zero? What is the precision at this rank?

d1
d2
d3
d4
d5
d6
d7
d8

Abstract example

- Suppose we have the following list for query q and the ground-truth contains entries d7 and d5
- What is Precision(R_1)? Note we can ignore sub-index j since we only have one query for the moment
- What is the first rank r at which Precision(R_r) becomes non-zero? What is the precision at this rank?
- What are the other ranks at which Precision is non-zero? Is the precision@r=6 zero?

d1
d2
d3
d4
d5
d6
d7
d8

Abstract example

- Suppose we have the following list for query q and the ground-truth contains entries d7 and d5
- What is Precision(R_1)? Note we can ignore sub-index j since we only have one query for the moment
- What is the first rank r at which Precision(R_r) becomes non-zero? What is the precision at this rank?
- What is the only other rank at which Precision is non-zero? What is the precision at this rank (careful…)?

d1
d2
d3
d4
d5
d6
d7
d8

What does this tell you about Precision? Do we need to calculate at every single rank?
--you will get a 'zig-zag' curve or sawtooth curve (why?)
--we **do** need to calculate precision at the ranks **between** the ranks of two relevant documents

Mean average precision (MAP)

- Once again, let's try to do this in steps

$$\mathrm{MAP}(Q) = \frac{1}{|Q|}\sum_{j=1}^{|Q|}\frac{1}{m_j}\sum_{k=1}^{m_j}\mathrm{Precision}(R_{jk})$$

Also known as precision@k

- We'll assume a single query q, allowing us to ignore the outer sum for the moment
- Let's return to our previous or 'running' example, but first, don't forget the definition(s) of precision

$$P = tp/(tp+fp)$$

$$\mathrm{Precision} = \frac{\#(\text{relevant items retrieved})}{\#(\text{retrieved items})} = P(\text{relevant}|\text{retrieved})$$

MAP: now you can do this for every query in Q and average…

- Just like NDCG, formula looks complicated when you first look at it, make sure you understand it completely!
- Let R_{jk} be the set of ranked retrieval results from the top result until we get to document d_k
  - Notice the index j, this tells you the ranked list was in response to query q_j
  - m_j is the number of relevant documents in the ground truth of query q_j
  - You don't need to remember any of this by heart, but you must understand what it all means

$$\text{MAP}(Q) = \frac{1}{|Q|} \sum_{j=1}^{|Q|} \frac{1}{m_j} \sum_{k=1}^{m_j} \text{Precision}(R_{jk})$$

- The 'average' refers to averaging over 'precisions' (notice what 'k' ranges over, this is not trivial!)
- The 'mean' is easier to understand, simply the average over all queries
- Each AP (and hence, MAP) is always between 0 and 1

Abstract example

- Suppose we have the following list for query q and the ground-truth contains entries d7 and d5
- What is Precision(R_1)? Note we can ignore sub-index j since we only have one query for the moment
- What is the first rank r at which Precision(R_r) becomes non-zero? What is the precision at this rank?
- What is the only other rank at which Precision is non-zero? What is the precision at this rank (careful…)?

d1
d2
d3
d4
d5
d6
d7
d8

What does this tell you about Precision? Do we need to calculate at every single rank?

--you will get a 'zig-zag' curve or sawtooth curve (why?)

--we **do** need to calculate precision at the ranks **between** the ranks of two relevant documents

--averaging must be done over Precision@r=1…7, even though Precision@1, 2…4 = 0

Addendum

Mean average precision (MAP)

- Just like NDCG, formula looks complicated when you first look at it, make sure you understand it completely!
- Let R_{jk} be the set of ranked retrieval results from the top result until we get to document d_k
  - Notice the index j, this tells you the ranked list was in response to query q_j
- You don't need to remember any of this by heart, but you must understand what it all means

$$\mathrm{MAP}(Q) = \frac{1}{|Q|} \sum_{j=1}^{|Q|} \frac{1}{m_j} \sum_{k=1}^{m_j} \mathrm{Precision}(R_{jk})$$

- Two ways to calculat
  - Interpretation 1 (studied in class): m_j is the 'rank' of the 'last' relevant document.
  - Interpretation 2 (found on the Internet and in some books): m_j is the number of relevant documents in the ground truth of query q_j. 'k' ranges only over those ranks where a relevant document is present
- Regardless of the interpretation:
  - Each AP (and hence, MAP) is always between 0 and 1
  - Only the perfect ranking gets an AP of 1 (hence, MAP is only 1 if all queries get perfect rankings)
- The first interpretation is much more aggressive in penalizing sub-optimal rankings compared to the second interpretation (why? *Hint: Think about the 'denominator' m_j when calculating the average precision. For which interpretation will this number* ***always*** *be equal or higher?*)

Mean average precision (MAP)

- Once again, let's try to do this in steps

$$\mathrm{MAP}(Q) = \frac{1}{|Q|} \sum_{j=1}^{|Q|} \frac{1}{m_j} \sum_{k=1}^{m_j} \mathrm{Precision}(R_{jk})$$

- We'll assume a single query q, allowing us to ignore the outer sum for the moment
- Let's return to our previous or 'running' example, but first, don't forget the definition(s) of precision

$$P = tp/(tp + fp)$$

$$\mathrm{Precision} = \frac{\#(\text{relevant items retrieved})}{\#(\text{retrieved items})} = P(\text{relevant}|\text{retrieved})$$

Abstract example

- Suppose we have the following list for query q and the ground-truth contains entries d7 and d5
- What is Precision(R_1)? Note we can ignore sub-index j since we only have one query for the moment

d1
d2
d3
d4
d5
d6
d7
d8

Abstract example

- Suppose we have the following list for query q and the ground-truth contains entries d7 and d5
- What is Precision(R_1)? Note we can ignore sub-index j since we only have one query for the moment
- What is the first rank r at which Precision(R_r) becomes non-zero? What is the precision at this rank?

d1
d2
d3
d4
d5
d6
d7
d8

Abstract example

- Suppose we have the following list for query q and the ground-truth contains entries d7 and d5
- What is Precision(R_1)? Note we can ignore sub-index j since we only have one query for the moment
- What is the first rank r at which Precision(R_r) becomes non-zero? What is the precision at this rank?
- What are the other ranks at which Precision is non-zero? Is the precision@r=6 zero?

d1
d2
d3
d4
d5
d6
d7
d8

Abstract example

- Suppose we have the following list for query q and the ground-truth contains entries d7 and d5
- What is Precision(R_1)? Note we can ignore sub-index j since we only have one query for the moment
- What is the first rank r at which Precision(R_r) becomes non-zero? What is the precision at this rank?
- What is the only other rank at which Precision is non-zero? What is the precision at this rank (careful…)?

d1
d2
d3
d4
d5
d6
d7
d8

What does this tell you about Precision? Do we need to calculate at every single rank?
--you will get a 'zig-zag' curve or sawtooth curve (why?)
--per Interpretation 1 we **do** need to calculate precision at the ranks **between** the ranks of two relevant documents

Abstract example

- Suppose we have the following list for query q and the ground-truth contains entries d7 and d5
- What is Precision(R_1)? Note we can ignore sub-index j since we only have one query for the moment
- What is the first rank r at which Precision(R_r) becomes non-zero? What is the precision at this rank?
- What is the only other rank at which Precision is non-zero? What is the precision at this rank (careful…)?

d1
d2
d3
d4
d5
d6
d7
d8

What does this tell you about Precision? Do we need to calculate at every single rank?

--you will get a 'zig-zag' curve or sawtooth curve (why?)

--per Interpretation 2 we **do not** need to calculate precision at the ranks **between** the ranks of two relevant documents

--According to Interpretation 2, 'AP' is (1/5+2/7)/2

Which interpretation should I use?

- If you care very deeply about getting near-perfect results, which is the case with good IR (if you don't find what you're looking for when you search, how likely are you to 'scroll' down and spend time, vs. just re-doing the search a little differently?), then by all means use the interpretation I used in class (Interpretation 1)
- If you want results that degrade more slowly with sub-optimal ranks, then use interpretation 2 (there are also more examples of this interpretation on the internet)
- As long as you're consistent, I will not penalize either way. It is good for you to be explicit about which interpretation you're using, just to be on the safe side