CS6370 Assignment 2

Subhankar Chakraborty and S Sivasubramaniyan

March 2020

Question 1

The inverted index representation in the context of NLP is a mapping from words (terms) to its locations or frequencies in a given set of documents (corpus). For the example given in question-1, the inverted indices for the words are as follows:

$$\textbf{cat} = \begin{pmatrix} 4 \\ 2 \\ 1 \end{pmatrix},\ \textbf{dog} = \begin{pmatrix} 3 \\ 0 \\ 3 \end{pmatrix},\ \textbf{animal} = \begin{pmatrix} 1 \\ 3 \\ 3 \end{pmatrix}$$

Each word is represented as a 3-dimensional vector where each dimension is indicative of the frequency of the word in Doc A, Doc B and Doc C respectively.

Question 2

The Term-Frequency (TF) values are calculated as follows.

$$TF_{i,j} = \text{Number of times word } j \text{ occurs in document } i$$

It can be interpreted as a vector representation of documents in the word space where the weight of each word is the frequency of it's occurrence in the document.

TF Values

word →	**cat**	**dog**	**animal**
Doc A	4	3	1
Doc B	2	0	3
Doc C	1	3	3

The Inverse-Document-Frequency (IDF) is calculated for each word as follows. This value is indicative of the discriminating ability of a word. A higher value of IDF is indicative of a lower presence of the word across documents and thus a higher discriminating ability for the word.

$$IDF_i = log_{10}(\frac{\text{Total number of documents in the collection(N)}}{\text{Number of documents in which word } i \text{ occurs(n)}})$$

Different references use a base of 2 and 10 interchangeably. As the values of either are just scaled versions of each other, the choice of base doesn't affect our final results. We have used a base of 10 for our implementation.

IDF Values

word ↓	**IDF Value**
cat	0
dog	0.1761
animal	0

The final TF-IDF representation of documents in the word space is a scaled TF matrix where the weight of each word is scaled by their corresponding IDF values. The required TF-IDF matrix and documents represented in the TF-IDF word space are as follows:

TF-IDF Matrix

word →	**cat**	**dog**	**animal**
Doc A	0	0.5283	0
Doc B	0	0	0
Doc C	0	0.5283	0

$$\textbf{Doc A} = \begin{pmatrix} 0 \\ 0.5283 \\ 0 \end{pmatrix}, \ \textbf{Doc B} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}, \ \textbf{Doc C} = \begin{pmatrix} 0 \\ 0.5283 \\ 0 \end{pmatrix}$$

Question 3

Based on the inverted index constructed before, the documents retrieved will be a union of all the non-zero dimensions of the words contained in the query. Since the query is a single word containing only the word "dog", the documents retrieved will be **Doc A** and **Doc C** as those are the only documents containing the word "dog". It can be interpreted as the non-zero dimensions in the inverted index representation of "dog".

Question 4

The TF-IDF representation of the query "dog" is given by

$$\textbf{Query} = \begin{pmatrix} 0 \\ 0.1761 \\ 0 \end{pmatrix}$$

The TF-IDF representations of the retrieved documents are given by

$$\textbf{Doc A} = \begin{pmatrix} 0 \\ 0.5283 \\ 0 \end{pmatrix}, \ \textbf{Doc C} = \begin{pmatrix} 0 \\ 0.5283 \\ 0 \end{pmatrix}$$

Using the formula of the cosine of the angle between vectors $\mathbf{u}$ and $\mathbf{v}$:

$$\text{Cos}(\theta) = \frac{|<\mathbf{u},\mathbf{v}>|}{\|\mathbf{u}\|*\|\mathbf{v}\|}$$

The cosine similarities between the TF-IDF representations of Doc A and Doc C with the TF-IDF representation of the query are calculated as 1 and 1 respectively. Since both the retrieved documents have the same value of cosine similarity, both are assigned the same rank.

Question 5

Implementation of an Information Retrieval System for the Cranfield Dataset using the Vector Space Model has been attached.

Question 6

- The IDF of a term that occurs in every document is **0**.
- No. The IDF of every term is **not necessarily finite**. A word in the dictionary which **does not occur in any document** has infinite (∞) IDF. The IDF can be made finite by adding a smoothing term and slightly modifying the formula as follows:

  $$\text{IDF} = log_{10}(\frac{N+\lambda}{n+\lambda})$$

  N and n are as used earlier. $\lambda << N$ is a hyperparameter. Though, if our corpus of words is formed from the documents themselves, we can be rest assured that all the words occur at least in one document and therefore the smoothing parameter won't be required.

Question 7

A comparison of cosine similarity with some other distance measures that can be used to compare vectors are as follows.

- **Euclidean Distance**
  - Distance between two vectors defined as the L_2 norm between them.
  - Cosine similarity is scale invariant as it is a measure of angle between the vectors while euclidean distance is not scale invariant.
  - Documents being of much higher magnitude than queries, the euclidean distance between the query and documents is bound to be very high. Therefore, setting a threshold for retrieval becomes tough.
- **Manhattan Distance**
  - Distance between two vectors defined as the L_1 norm between them.
  - Once again worser than cosine similarity because of the same reasons as euclidean distance.
- **Jaccard Similarity Index**
  - The Jaccard similarity index can be defined as the ratio of intersection of non-zero dimensions between vectors with the union of non-zero dimensions between vectors. Higher the number of shared dimensions over distinct ones implies a higher similarity.
  - This takes us back to the simple old model where the frequency and IDF of terms in the documents are essentially ignored. Therefore, it is expected to perform worser than cosine similarity.

Therefore, Cosine similarity looks like the most relevant and appropriate for our relatively simple Information Retrieval task.

Question 8

Accuracy is defined as

$$\text{Accuracy} = \frac{\text{True Positives + True Negatives}}{\text{Total Number of Documents}}$$

For our Information Retrieval(IR) task, the number of true negatives is expected to be almost equal to the total number of documents in most applications. So the accuracy values are bound to be close to 1 even for very bad IR systems, say with zero true positives. Hence, it is not a very good effectiveness measure for the performance of an IR system.

Question 9

The F_α measure is given by

$$F_\alpha = \frac{1}{\frac{\alpha}{\text{precision}} + \frac{1-\alpha}{\text{recall}}}$$

- For $\alpha \approx 1$, $F_\alpha \approx$ precision
- For $\alpha \approx 0$, $F_\alpha \approx$ recall
- For $\alpha \in$ [0,0.5) the F_α score gives more weightage to recall than precision.
- For $\alpha \in$ (0.5,1] the F_α score gives more weightage to precision than recall.

Question 10

Precision@k (P@k) counts the number of relevant results in the top k retrieved documents. One issue with this metric is that it fails to capture the positions of the relevant documents among the top k. For example, say the documents retrieved by two IR systems (rank goes from high to low) are [1,1,1,1,1,0,0,0,0,0] and [0,0,0,0,0,1,1,1,1,1] (1 denotes a relevant document and 0 denotes a document which is not relevant) respectively. Both the IR systems will get the same Precision@10 score when the first one is doing much better than the second.

Average Precision@k is defined as

$$\text{AveP@k} = \sum_{j=1}^{j=k}(\text{Recall@(j)-Recall@(j-1)})*\text{P@}j$$

or equivalently,

$$\text{AveP@k} = \frac{\sum_{j=1}^{j=k}(P@j * rel(j))}{\text{Number of relevant documents}}$$

$$rel(j) = \begin{cases} 1, & \text{if } j^{th} \text{ document is relevant} \\ 0, & \text{otherwise} \end{cases}$$

It is therefore much better at capturing the **positions** of relevant documents. It may be noted that AveP@10 will score the first IR system much higher than the second IR system in our example.

Question 11

The Mean Average Precision@k (MAP@k) over Q queries is defined as

$$\text{MAP@k} = \frac{\sum_{q=1}^{q=Q} \text{AveP@k}_q}{\text{Q}}$$

Mean average precision for a set of queries is the mean of the average precision scores for each query. Unlike AveP@k which is defined for a single query, MAP@k is defined for a set of queries, therefore better indicative of the performance of an IR system.

Question 12

nDCG is a better metric than AveP for the Cranfield dataset for the following reasons.

- The human relevance judgements in the Cranfield dataset, apart from mentioning whether a document is relevant to a query, also ranks how relevant a document is to a query. (Higher position indicating higher relevance)
- AveP is designed for binary (relevant/non-relevant) ratings and doesn't take into account fine-grained numerical ratings for relevance.
- In the calculation of AveP, we might want to threshold the fine-grained ratings to make binary relevance predictions, therefore introducing bias in the evaluation metric because of the manually set threshold. Besides, we are also throwing away the fine-grained information.

Question 13

Implementation of the above mentioned evaluation metrics for our IR system has been attached.

Question 14

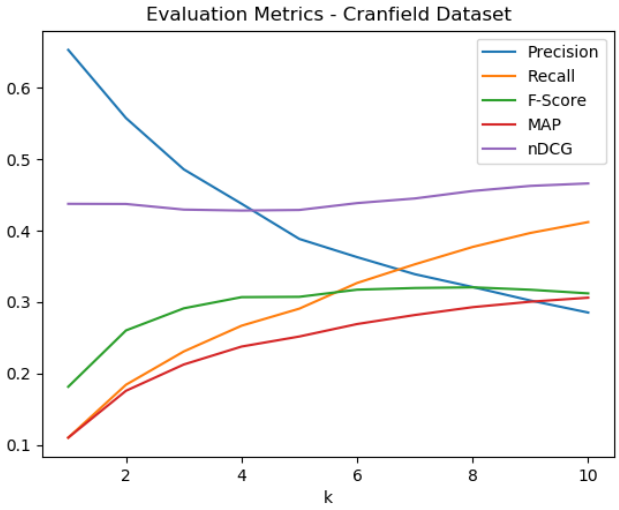

Figure 1: Evaluation Metrics

The plot for different evaluation metrics as a function of k is shown in Figure 1. The following observations can be made from the plot:

- Precision decreases monotonically with k. This is as expected for a well functioning IR system as the density of relevant documents is expected to decrease with increase in retrieved documents. It is important to note that monotonicity is not a necessity for precision, rather indicative of our IR system's performance.
- Recall monotonically increases with k as expected. As the total number of relevant documents is a constant, the number of relevant documents retrieved can only increase with k.
- F-score initially increases with k and then stabilizes. F-score is generally used to compare various models to break the precision-recall trade-off. In this case, the initial increase and the stabilization later can probably be seen as the reason for why a good number of documents are retrieved and shown to the end-user to choose from.
- Average Precision is also expected to increase monotonically with k from definition, as it is observed from the plot.
- The nDCG values are relatively constant with a minor increase for larger values of k. This may be indicative of an overall stable model, with a good correlation to human relevance judgements.

Question 15

Some of the queries for which the performance is not as expected are mentioned below. The possible explanation is also mentioned along with it.

- "papers on internal /slip flow/ heat transfer studies ."
  - Each of the words in the query other than '/slip' and 'flow/' after inflection reduction occur in more than 50 documents in the Cranfield corpus and are hence, not discriminatory.
  - The words '/slip' and 'flow/' after inflection reduction stay '/slip' and 'flow/' respectively. '/slip' never occurs in any of the Cranfield documents and is an out of dictionary word.
  - In the human labeled documents relevant to the query mentioned, the words occur either as "slip-flow" or "slip" and "flow" separately. The same words are present in the reduced documents.
  - Hence there is no way for the matching algorithm to get the correct results and Precision@k is zero for k>5.

  - This is an instance where a weakness of the Penn Treebank tokenizer is apparent.

- "papers on shock-sound wave interaction ."
  - The matching algorithm performs poorly for this query for the same reasons mentioned above.

- "what are the details of the rigorous kinetic theory of gases . (chapman-enskog theory) ."
  - The query and the most relevant document to it (position labeled as 1) do not have a single term in common after inflection reduction which makes the cosine similarity between them to be zero.
  - The rest three documents labeled relevant to the query have position values 3 or 4 meaning they are weakly relevant to the query.
  - The aforementioned three documents also happen to have no terms in common with the query in their reduced forms. It is now impossible for the matching algorithm to identify them as relevant documents as the cosine similrity values are all zero.
  - This is an example of a scenario in which the vector space model fails.

A few other queries for which the matching algorithm failed to obtain any relevant documents in in its top 10 places are listed below.

- "does there exist a good basic treatment of the dynamics of re-entry combining consideration of realistic effects with relative simplicity of results ."
- "did anyone else discover that the turbulent skin friction is not over sensitive to the nature of the variation of the viscosity with temperature ."
- "what application has the linear theory design of curved wings ."
- "are there any papers dealing with acoustic wave propagation in reacting gases ."
- "how far around a cylinder and under what conditions of flow, if any, is the velocity just outside of the boundary layer a linear function of the distance around the cylinder ."
- "where can i find pressure data on surfaces of swept cylinders ."
- "do the discrepancies among current analyses of the vorticity effect on stagnation-point heat transfer result primarily from the differences in the viscosity-temperature law assumed ."
- "what parameters can seriously influence natural transition from laminar to turbulent flow on a model in a wind tunnel ."

- "is the problem of similarity for representative investigations of aeroelastic effects in heated flow as intractable as previous investigations imply ."
- "what is the magnitude and distribution of lift over the cone and the cylindrical portion of a cone-cylinder configuration ."
- "is there any information on how the addition of a /boat-tail/ affects the normal force on the body of various angles of incidence ."
- "what is the best theoretical method for calculating pressure on the surface of a wing alone ."
- "how can the effect of the boundary-layer on wing pressure be calculated, and what is its magnitude ."
- "exact solution methods for calculating the ablative mass loss of a material ablating at high temperatures in a hypersonic flight environment ."
- "how do large changes in new mass ratio quantitatively affect wing-flutter boundaries ."
- "what investigations have been made of the wave system created by a static pressure distribution over a liquid surface ."

Question 16

Some of the shortcomings in using a vector space model for IR are as follows:

- It implicitly assumes that the terms are orthogonal to each other. This is often not the case.
- Therefore, documents with similar content but different vocabularies will get a small similarity measure.
- High latency, i.e., it is very calculation intensive and takes a lot of processing time.
- Every time we add a new term to the term space, we need to recalculate all the vectors.
- It is not the best at modeling the sequence in which the terms occur in a document (Especially with unigrams).
- Very long documents lead to difficulties in calculating the similarity measure. i.e., the issue of small dot products and high dimensionality.

Question 17

When the dataset in consideration is similar to the Cranfield dataset, we can use the top down knowledge that the documents are not very long. In such a case, the contribution from the title can be included by adding the TF-IDF representation of the title to the TF-IDF representation of the document. If we want the title to have three times the contribution of the text in the body, we can add 3 times the TF-IDF representation of the title to the representation of the document, therefore giving more weight to the title.
The documents not being very long is vital because if that is not the case, the magnitudes of the representations of the documents will be much greater than the magnitudes of the representations of the titles. Directly adding the weighted representation of the title to the representation of the document will mean the title makes very little contribution. Higher weights might put over-emphasis on the titles. Hence, striking a proper balance becomes tough. It becomes even tougher for a dataset with varying lengths of documents. We have to look for more novel techniques in such a case.

Question 18

The advantages and disadvantages of using a bigram model over a monogram model are listed below.

- **Advantages:**
  - Vector space representations made using bigrams are much better at modeling sequence, ie, the order in which terms occur in a document.
  - Because of the above mentioned reason, they are also better at modeling context.
  - As bigrams model context and sequencing better, the precision is higher than that obtained while using monograms.
- **Disadvantages:**
  - The recall is lower than that obtained while using monograms.
  - The bigram dimensions for the vector space model would have been formed based on the bigrams present in the document corpus. In case the query does not contain any of those bigrams, which is quite likely as queries are often not grammatically correct sentences, we will have an out of vocabulary issue. Also, it is generally the tendency of users to type in a query as bag of words, rather than making grammatical sense out of it. Although out of vocabulary issues occur in monograms as well, with bigrams they are much more common. In such a scenario we may have to resort to smoothing or assign a vector representation to out of vocabulary words.

- The number of bigrams formed would be in general much higher than the number of terms. This would mean that our vector space is very high dimensional. Given that each document will contain very few of these bigrams, we will have sparse vectors in a very high dimensional space. Also, latency increases multifold from our unigram model.
- There is also the issue of orthogonality. Our simple vector space model assumes that dimensions are orthogonal to each other. Even though this issue exists even with unigrams, it is sort of easier to interpret unigrams as orthogonal building blocks that make up a sentence. The issue blows up multifold with our bigram model as multiple bigrams contain the same term (unigram) but would still be regarded as orthogonal in our vector space model. Therefore, interpretability becomes very hard.

Question 19

Some implicit ways to get relevance feedback are mentioned below.

- **Clickthrough data**
  When the searcher engages with a search result (e.g., by clicking on it), the search engine treats the engagement as implicit positive feedback. Conversely, when the searcher doesn't engage with a search result — either by clicking on a lower-ranked result or by not clicking on any results — the search engine treats the lack of engagement as implicit negative feedback.

- **User Query History**
  Observations of the user's history of query submissions. This includes reformulations, or query rewrites, which can be used to infer the user's dissatisfaction with the results returned for the original formulation. An examination of the queries that immediately preceded a query can also be an indication of the user's interest, which can be used to disambiguate queries that have meanings in multiple domains. The canonical example of such a query is "Java", which could refer to coffee, the Indonesian island, or the programming language; knowing that one of the previous queries was "C++", another programming language, for instance, would help pinpoint the meaning of this query.

- **User's Entire History**
  Observations of all information created, copied, or viewed by the user. This could include everything from webpages viewed, to emails, calendar items, and documents in the user's filesystem.

- **Reading Time**
  Observation of the amount of time the user spends on each result. It seems reasonable that a user would spend more time on more relevant results.

- **Eye Tracking**
  Observation of features such as eye fixation and pupil dilation as the user observes the results. The hypothesis is that features such as duration of fixation and diameter of the pupil vary in a systematic way between relevant and non-relevant results; for example, larger pupil diameter might be an indication of relevance.

References

1. Stanford NLP article on Inverted Index
2. Wikipedia article on Evaluation Metrics
3. Cornell University CS6740 lecture notes
4. Medium article on Relevance Feedback
5. Medium article on Mean Average Precision
6. Medium article on Similarity Metrics