Assignment 2

Team 10

Akshat Meena (CS19B052) Siddharth Singh (CS19B072)

1. **Inverted index**
   a. Herbivores → S1
   b. typically → S1, S2
   c. plant → S1, S2
   d. eaters → S1, S2
   e. meat → S1, S2
   f. Carnivores → S2
   g. Deers → S3
   h. eat → S3
   i. grass → S3
   j. leaves → S3
2. TF-IDF(t) $= n_t * log_2(N/n)$, where n_t = frequency of type 't', N = total number of documents in the collection, and n = the number of documents in which type 't' occurs.

   Let $\vec{a}, \vec{b}, \vec{c}, \vec{d}, \vec{e}, \vec{f}, \vec{g}, \vec{h}, \vec{i}, \vec{j}$ be the corresponding vectors for the terms "Herbivores", "typically", "plant", "eaters", "meat", "Carnivores", "Deers", "eat", "grass", "leaves" respectively. Then, the TF-IDF vector representations for the above documents are:

   - $\vec{S_1} = log_2(3)\vec{a} + log_2(1.5)\vec{b} + log_2(1.5)\vec{c} + 2 * log_2(1.5)\vec{d} + log_2(1.5)\vec{e}$
   - $\vec{S_2} = log_2(1.5)\vec{b} + log_2(1.5)\vec{c} + 2 * log_2(1.5)\vec{d} + log_2(1.5)\vec{e} + log_2(3)\vec{f}$

- $\vec{S_3} = log_2(3)\vec{g} + log_2(3)\vec{h} + log_2(3)\vec{i} + log_2(3)\vec{j}$

3. Based on the inverted index, documents S1 and S2 will be retrieved.
4. The TF-IDF vector representation for the query will be $\vec{q} = log_2(1.5)\vec{c} + log_2(1.5)\vec{d}$.

   CosineSim($\vec{q}$, $\vec{S_1}$) = 0.560156917515788

   CosineSim($\vec{q}$, $\vec{S_2}$) = 0.560156917515788

   CosineSim($\vec{q}$, $\vec{S_3}$) = 0

   Ranking of documents based on the cosine similarity:

   Rank 1 → S1, S2

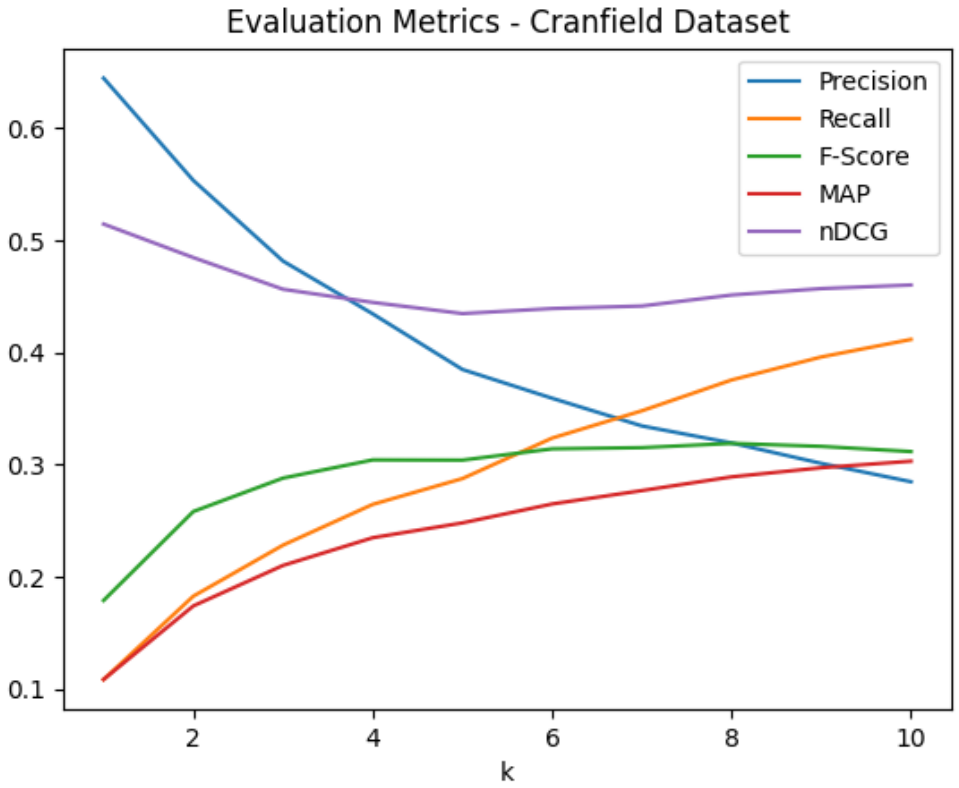
   Rank 2 → S3
5. No, if we consider the context of the query, then S2 should be ranked last as the query is regarding plant eaters or herbivores, but S2 describes carnivores. Thus, S3 should also be considered relevant to the query and rank higher than S2.
6. Done
7. a. 0
   b. Yes, the IDF of a term is always finite because the terms that occur in the document collection are only considered, and hence the value of n in the formula of $IDF = log_2(N/n)$ will always be in the range $[1, N]$.
8. Yes. Other measures include the angle, dot product, and Euclidean distance between the vectors. Cosine similarity is better than these methods because it captures the notion of similarity in terms of direction, which is essential for IR applications. Also, it is computationally more efficient than other metrics.
9. Accuracy is not used as a metric to evaluate information retrieval systems because of the class imbalance in most IR scenarios. The number of retrieved documents is always much smaller than those not retrieved.
10. $F_\alpha = \frac{1}{\alpha\frac{1}{P} + (1-\alpha)\frac{1}{R}}$, where $0 \leq \alpha \leq 1$
    a. $F_\alpha = R$ at $\alpha = 0$
    b. $F_\alpha = P$ at $\alpha = 1$
    c. F_α gives equal weightage to precision and recall at $\alpha = \frac{1}{2}$

d. At $0 \leq \alpha < \frac{1}{2}$, F_α gives more weightage to recall than to precision

Note that we also have $F_\beta = \frac{(\beta^2+1)PR}{\beta^2 P+R}$, where $\beta \geq 0$

1. $F_\beta = P$ at $\beta = 0$
2. $F_\beta = R$ at $\beta = \infty$
3. F_β gives equal weightage to precision and recall at $\beta = 1$
4. At $\beta > 1$, F_β gives more weightage to recall than to precision

11. A shortcoming of Precision@k that Average Precision@k addresses is that it considers the ranks in which the items are retrieved and penalizes the system if the relevant items are retrieved at lower ranks. On the other hand, Precision@k is agnostic of the ranks of the relevant items.
12. Mean Average Precision (MAP)@k is the mean of Average Precision@k over some queries. Average Precision@k is defined for one query, whereas MAP@k is obtained by taking the mean of Average Precision@k over a finite number of queries. Since it considers the system's effectiveness over different queries, MAP@k is a more reliable metric than AP@k.
13. nDCG is a more appropriate evaluation measure because AP only evaluate a document as relevant or non-relevant. In contrast, nDCG uses human relevance scores which give weightage according to how relevant a document is.
14. Done

15. 

- **Precision and Recall:** Precision and Recall are following the trend of a sound IR system. Precision is monotonically decreasing, and Recall is monotonically increasing and they intersect and pass each other at the crossover point.
- **F-Score:** We can see that F-score increases initially but fail to meet at the crossover point with precision and recall.
- **MAP:** Mean Average Precision can be seen increasing with the rank as it gets more relevant document as we increase the rank.
- **nDCG:** Initially decreases, then start increasing gradually.

16. On analysis of the performance of our search engine on the given dataset, the following points were observed:
    - Some queries retrieve documents based on the presence of terms, but the context they are used for is irrelevant.
    - For some queries, the relevant documents were ranked lower than their actual relevance.

- It is hard to rank the document with cosine similarity = 0 even if it is slightly relevant, as the system matches keywords, not context.

17. A conceptual shortcoming of using a Vector Space Model for IR is that it assumes that the terms are independent of each other. This is a wrong assumption to make as there are a lot of synonymous terms that are certainly not independent of each other. Word-relatedness should be taken into account. The technique of similarity propagation followed by relevance propagation is an improvement over the naive Vector Space Model.
    Also, one more issue with the Vector Space Model model is that it doesn't consider the order of the words in the queries/documents. It views the documents and queries as a set of words rather than as a sequence of words.

18. We can include the title in the document vector by treating it as a part of the document. Our set of terms (equal to the dimensionality of our vector space) would also (mostly) increase.
    Consider the following example for a better understanding.
    S_1 → {{AB},{CDEF}}
    S_2 → {{C},{AFG}}
    In both documents, the title and the body are enclosed within the first and the second curly brackets, respectively.
    Their vector representations will be as follows:

    $$\vec{S}_1 = 1 * log_2(2/2)\vec{A} + 1 * log_2(2/1)\vec{B} + 1 * log_2(2/2)\vec{C} + 1 * log_2(2/1)\vec{D} + 1 * log_2(2/1)\vec{E} + 1 * log_2(2/2)\vec{F} + 0 * log_2(2/1)\vec{G} = log_2(2)\vec{B} + log_2(2)\vec{D} + log_2(2)\vec{E}$$

    $$\vec{S}_2 = 1 * log_2(2/2)\vec{A} + 0 * log_2(2/1)\vec{B} + 1 * log_2(2/2)\vec{C} + 0 * log_2(2/1)\vec{D} + 0 * log_2(2/1)\vec{E} + 1 * log_2(2/2)\vec{F} + 1 * log_2(2/1)\vec{G} = log_2(2)\vec{A} + log_2(2)\vec{C} + log_2(2)\vec{F} + log_2(2)\vec{G}$$

    For weighing the contribution of the title three times that of the document, we can make the term frequency of the words present in the title thrice.
    E.g., the above documents will be converted to their below-mentioned equivalents.
    S_1 → {{ABABAB},{CDEF}}
    S_2 → {{CCC},{AFG}}

19. Advantage - Bigrams are better at modelling the sequences as it considers two words at a time, giving a better insight into the order of words.

Disadvantage - The vector space formed by bigrams is high dimensional as the number of bigrams formed is way more than the unigrams. This leads to a sparse vector representation
and require more computational time.

20. The methods to obtain relevance feedback from the user implicitly are:
    a. Noting which documents users do and do not select for viewing
    b. the duration of time spent viewing a document
    c. page browsing or scrolling actions

References

- Class slides
- https://towardsdatascience.com/mean-average-precision-at-k-map-k-clearly-explained-538d8e032d2
- https://en.wikipedia.org/wiki/Relevance_feedback#:~:text=Relevance feedback is a feature,to perform a new query.