Grenze International Journal of Engineering and Technology, January Issue

How Does Your Prompt Get Responses: A Study of Distance Measuring Algorithms

Vishal Garg[1] and Dr. Ravinder Singh Madhan[2]
[1-2]IEC University, Baddi, Himachal Pradesh, India
Email:-vishalps2000@yahoo.com

***Abstract*—This paper presents a comprehensive analysis of various distance measuring algorithms used to generate responses for prompts in modern information retrieval systems. The study explores the transition from traditional keyword-based methods such as BM25 to advanced context-aware techniques including CrossEncoders and Dense Bi-Encoders. We examine multiple similarity and distance metrics including Euclidean distance, Manhattan distance, Jaccard similarity, Pearson correlation, dot product, Hamming distance, Minkowski distance, and KL divergence. Additionally, the paper investigates hybrid approaches that combine sparse and dense retrieval methods to achieve optimal performance. Through comparative analysis and practical implementations, we demonstrate how these algorithms work individually and in combination to improve document retrieval, question answering, and semantic search capabilities. Our findings highlight the importance of selecting appropriate distance metrics based on specific use cases, data characteristics, and desired outcomes in natural language processing applications.**

***Index Terms*— Distance Algorithms, Information Retrieval, Cosine Similarity, BM25, Cross-Encoders, Bi- Encoders, Semantic Search, Hybrid Retrieval.**

I. INTRODUCTION

In the era of artificial intelligence and natural language processing, understanding how prompts generate accurate and contextually relevant responses has become crucial. The mechanism behind prompt response generation relies heavily on distance measuring algorithms that evaluate similarity between query vectors and document embeddings. This paper investigates the fundamental question:
How does your prompt get responses?
The evolution of information retrieval has witnessed a significant transformation from simple keyword matching to sophisticated neural network-based approaches. Traditional methods like Boolean retrieval and TF-IDF focused primarily on exact word matching, which often failed to capture semantic meaning and contextual relationships. Modern approaches leverage vector embeddings and neural architectures to understand the deeper meaning behind queries and documents.

A. Research Objectives

- Provide comprehensive overview of distance measuring algorithms in prompt-response generation
- Analyze transition from keyword-based to context-aware techniques
- Evaluate hybrid approaches combining multiple retrieval methods
- Present practical implementations and comparative analysis

Grenze ID: 01.GIJET.12.1.540

II. Related Work

Early information retrieval focused on Boolean models and term frequency approaches. Key developments include:

- Vector Space Model by Salton and McGill
- TF-IDF by Sparck Jones for statistical document ranking
- BM25 by Robertson and Walker for probabilistic retrieval
- Word2Vec by Mikolov et al. enabling semantic understanding
- BERT by Devlin et al. revolutionizing natural language understanding
- Bi-Encoders and Cross-Encoders for neural retrieval
- Hybrid approaches combining sparse and dense methods

III. Distance Measuring Algorithms

A. Basic Distance MetricsEuclidean Distance

- Formula: $d(p,q) = \sqrt{(\Sigma(pi - qi)^2)}$
- Use: Numerical data, clustering (K-means), geometric distance interpretation

Manhattan Distance (L1)

- Formula: $d(p,q) = \Sigma|pi - qi|$
- Use: Grid structures, high-dimensional sparse data, reduced outlier influence

Minkowski Distance

- Formula: $d(p,q) = (\Sigma|pi - qi|^p)^(1/p)$
- Use: Flexible distance measurement (p=1 for Manhattan, p=2 for Euclidean)

Hamming Distance

- Formula: $d(A,B) = \Sigma(ai \neq bi)$
- Use: String/binary comparison, error detection, DNA sequences

B. Similarity Measures

Cosine Similarity

- Formula: $similarity = (A \cdot B) / (||A|| \times ||B||)$
- Use: Text embeddings, document similarity, direction-focused comparison

Dot Product

- Formula: $A \cdot B = \Sigma(ai \times bi)$
- Use: When magnitude and direction matter, neural networks, recommendations

Jaccard Similarity

- Formula: $J(A,B) = |A \cap B| / |A\ B|$
- Use: Set comparison, binary features, collaborative filtering

Pearson Correlation

- Formula: $r = \Sigma((xi - \bar{x})(yi - \bar{y})) / \sqrt{(\Sigma(xi - \bar{x})^2 \times \Sigma(yi - \bar{y})^2)}$
- Use: Statistical analysis, feature correlation, recommendation systems

C. Advanced Retrieval Methods

BM25 (Best Matching 25)

Probabilistic retrieval model with term frequency saturation and document length normalization

- Formula: $score(D,Q) = \Sigma\ IDF(qi) \times (f(qi,D) \times (k1+1)) / (f(qi,D) + k1 \times (1-b+b \times |D|/avgdl))$
- Advantages: Effective keyword retrieval, computationally efficient
- Limitations: Cannot capture semantic relationships, requires exact term matching

Dense Bi-Encoders

Separate encoders for queries and documents creating dense vectors

- Process: Pre-compute document embeddings → encode query → similarity search → return top-k
- Advantages: Scalable to millions of documents, captures semantic similarity, fast retrieval
- Limitations: Lower accuracy than cross-encoders, fixed representations

Cross-Encoders

Jointly process query-document pairs for fine-grained interaction

- Process: Concatenate [CLS] query [SEP] document [SEP] → transformer → relevance score Advantages: Highest accuracy, captures nuanced relationships
- Limitations: Computationally expensive, cannot pre-compute embeddings

D. Hybrid Approaches

Combines sparse (BM25) with dense neural retrieval:
Architecture:

- Stage 1 - Sparse Retrieval: BM25 retrieves initial candidates (top 1000)
- Stage 2 - Dense Re-ranking: Bi-Encoder/Cross-Encoder re-ranks candidates
- Score Fusion: Weighted combination of scores Fusion Strategies:
- Linear combination: final_score = α × sparse_score + (1-α) × dense_score
- Reciprocal Rank Fusion
- Learning-to-Rank

KL Divergence

Formula: $D_{KL}(P||Q) = \Sigma P(i) \times \log(P(i)/Q(i))$
Use: Topic modeling, probability distribution comparison

E. Proposed Multi-Stage Pipeline

- Candidate Generation (BM25): Retrieve top 1000-5000 candidates
- Semantic Re-ranking (Bi-Encoder): Reduce to top 100-500 documents
- Fine-grained Scoring (Cross-Encoder): Final relevance scores
- Score Fusion: Combine scores with learned weights

IV. Test Results

A. Experimental Setup

Datasets: MS MARCO (8.8M passages), Natural Questions, Custom corpus (100K docs)
Metrics: MRR@10, NDCG@10, Recall@100, Query latency

TABLE I: DISTANCE METRICS PERFORMANCE

Metric	Euclidean	Manhattan	Cosine	Dot Product	Jaccard
MRR@10	0.312	0.298	**0.385**	0.371	0.245
NDCG@10	0.428	0.415	**0.492**	0.478	0.356
Recall@100	0.682	0.671	**0.748**	0.735	0.598
Latency (ms)	2.3	2.1	2.5	**1.8**	3.2

Key Findings:

Cosine similarity achieved highest accuracy
Dot product best balance of speed and accuracy
Jaccard performed poorly for continuous embeddings

Key Findings:

- Cross-Encoder highest accuracy but significant latency
- Bi-Encoder good balance for large-scale retrieval
- BM25 competitive baseline with fastest retrieval
- Multi-stage pipeline achieved near cross-encoder accuracy with 8.5× speedup

B. Query Type Analysis

- Keyword Queries: BM25 superior (MRR@10: 0.425)
- Semantic Queries: Cross-Encoder best (MRR@10: 0.456)

Hybrid Queries: Hybrid systems optimal (MRR@10: 0.421)

TABLE II: RETRIEVAL METHODS PERFORMANCE

Method	MRR@10	NDCG@10	Recall@100	Latency (ms)
BM25	0.312	0.401	0.712	8.5
Bi-Encoder	0.368	0.465	0.758	15.2
Cross-Encoder	**0.421**	**0.538**	0.742	1250.0
Hybrid (BM25+Bi)	0.392	0.493	0.791	23.7
Multi-stage	0.418	0.531	**0.798**	145.8

V. DISCUSSION

A. Distance Metrics Analysis

Cosine similarity most effective for text embeddings due to normalization properties focusing on directional similarity. Dot product competitive when magnitude information contributes to relevance. Euclidean and Manhattan distances acceptable but inferior for high-dimensional text embeddings. Jaccard similarity not optimal for continuous embeddings.

B. Retrieval Method Effectiveness

Traditional Methods
- BM25 demonstrates remarkable resilience with consistent performance on keyword queries
- Computational efficiency makes it indispensable for first-stage candidate generation
- Limited in capturing semantic relationships

Neural Retrieval
- Bi-Encoders capture contextual meaning with scalable pre-computation
- Cross-Encoders achieve superior accuracy through joint modeling but at computational cost
- Attention mechanisms capture subtle relevance signals

Hybrid Systems
- Most balanced performance profile across query types
- Multi-stage pipeline achieves 99% of cross-encoder accuracy with 8.5× speedup
- Modular design enables component optimization

C. Practical Recommendations

Large-Scale Search (>10M documents)
- Use Bi-Encoder with approximate nearest neighbor search
- Consider hybrid approach for improved accuracy

High-Precision Applications
- Deploy multi-stage pipeline with cross-encoder re-ranking
- Use BM25 for efficient candidate generation

Resource-Constrained Environments:
- BM25 provides excellent baseline with minimal requirements
- Consider distilled neural models

Domain-Specific Applications:
- Fine-tune neural models on domain data
- Combine with domain-specific keyword matching

Limitations
- Computational resources required for neural methods
- Substantial training data needed for dense retrieval
- Domain adaptation challenges
- Limited interpretability of neural methods
- Ongoing maintenance overhead

VI. CONCLUSION

This study presented comprehensive analysis of distance measuring algorithms in modern prompt response systems, from traditional keyword-based to sophisticated neural architectures.

Key Contributions:

- Comprehensive taxonomy of distance metrics and retrieval methods
- Empirical comparison demonstrating strengths and limitations
- Multi-stage pipeline combining sparse and dense methods effectively
- Practical guidelines for method selection based on requirements Key Findings:
  - Cosine similarity most effective for text embeddings
  - Hybrid sparse-dense approaches offer best balance of accuracy and efficiency
  - Multi-stage pipeline achieves excellent results suitable for production
  - Traditional methods like BM25 remain crucial for efficient candidate generation
  - The future of information retrieval lies in intelligently combining traditional methods with neural approaches. Understanding the entire retrieval pipeline enables practitioners to build effective systems delivering accurate, relevant results while meeting practical constraints.

REFERENCES

[1] G. Salton and M. J. McGill, "Introduction to Modern Information Retrieval," McGraw-Hill, 1983.

[2] K. Sparck Jones, "A statistical interpretation of term specificity and its application in retrieval,"Journal of Documentation, vol. 28, no. 1, pp. 11-21, 1972.

[3] S. Robertson and S. Walker, "Some simple effective approximations to the 2-Poisson model forprobabilistic weighted retrieval," in Proc. SIGIR '94, pp. 232-241, 1994.

[4] T. Mikolov et al., "Efficient estimation of word representations in vector space," arXiv preprintarXiv:1301.3781, 2013.

[5] J. Devlin et al., "BERT: Pre-training of deep bidirectional transformers for language understanding,"in Proc. NAACL-HLT, pp. 4171-4186, 2019.

[6] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence embeddings using Siamese BERTnetworks," in Proc. EMNLP-IJCNLP, pp. 3982-3992, 2019.

[7] V. Karpukhin et al., "Dense passage retrieval for open-domain question answering," in Proc. EMNLP,pp. 6769-6781, 2020.

[8] L. Xiong et al., "Approximate nearest neighbor negative contrastive learning for dense text retrieval,"in Proc. ICLR, 2021.

[9] H. Zamani et al., "Learning to rank in the age of muppets: Effectiveness-efficiency tradeoffs in multistage ranking," in Proc. ICTIR, pp. 71-78, 2020.

[10] J. Lin et al., "Pretrained transformers for text ranking: BERT and beyond," Synthesis Lectures onHuman Language Technologies, vol. 14, no. 4, pp. 1-325, 2021.

[11] P. Bajaj et al., "MS MARCO: A human generated machine reading comprehension dataset," arXivpreprint arXiv:1611.09268, 2016.

[12] O. Khattab and M. Zaharia, "ColBERT: Efficient and effective passage search via contextualized lateinteraction over BERT," in Proc. SIGIR, pp. 39-48, 2020.