1. \*\*TF-IDF (Term Frequency-Inverse Document Frequency):\*\* TF-IDF is a classic text analysis technique that evaluates the importance of words within a document. You can apply TF-IDF to both the main post and comments separately. Then, you can calculate the cosine similarity between the TF-IDF vectors of the main post and each comment. Higher cosine similarity indicates greater relevance.

2. \*\*Word Embeddings (e.g., Word2Vec, GloVe, FastText):\*\* Word embeddings capture semantic relationships between words. You can represent the main post and comments using word embeddings and then compute a similarity score between the main post and each comment. Similarity measures like cosine similarity or Euclidean distance can be used.

3. \*\*LSTM (Long Short-Term Memory) or Transformer Models:\*\* Deep learning models like LSTMs or Transformers can capture the context and semantics of text. You can create embeddings for the main post and comments using these models and then use a similarity measure like cosine similarity or attention mechanisms to assess relevancy.

4. \*\*Topic Modeling (e.g., Latent Dirichlet Allocation - LDA):\*\* Topic modeling can help identify the main themes or topics in the main post and comments. By comparing the topics extracted from the main post to those in the comments, you can determine the relevance of each comment based on the shared topics.

5. \*\*Sentiment Analysis:\*\* Sentiment analysis models can evaluate the sentiment (positive, negative, neutral) of both the main post and comments. Comments with sentiment that aligns with the sentiment of the main post may be considered more relevant.

6. \*\*BERT (Bidirectional Encoder Representations from Transformers):\*\* BERT is a pre-trained transformer-based model that excels in capturing context and semantics in text. Fine-tuning BERT on your main post and comments data can help determine the relevancy of comments. You can use techniques like cosine similarity or pooling strategies (e.g., max-pooling) to compute similarity scores.

7. \*\*Text Classification Models:\*\* You can train text classification models to classify comments as relevant or irrelevant to the main post. Common algorithms for this task include Naive Bayes, Support Vector Machines, or even neural networks like CNNs (Convolutional Neural Networks) or RNNs (Recurrent Neural Networks).

8. \*\*Graph-Based Models:\*\* Represent the relationship between the main post and comments as a graph, where nodes are posts/comments and edges indicate relationships. Graph algorithms such as PageRank or centrality measures can help identify relevant comments based on their connections to the main post.

9. \*\*User Engagement Metrics:\*\* Assess relevancy based on user engagement metrics, such as the number of upvotes, likes, or replies a comment receives. Comments with higher engagement metrics are often considered more relevant to the main post.

10. \*\*Hybrid Models:\*\* Combine multiple relevancy models to provide a more comprehensive assessment. For example, you can use a combination of TF-IDF, sentiment analysis, and a neural network-based model to calculate a weighted average or ensemble score for each comment's relevancy.