International Research Journal of Multidisciplinary Sciences (Double - Blind Peer Review Journal)
VOL-2 ISSUE-3 March 2026 PP:37-44
ISSN:3107-930X(ONLINE)

A Review of Feature Selection for Automatic Answer Extraction from Online Web Forums

[1]Hemant Sharma,[2]Shyamol Banerjee

[1]M.Tech Student, Dept. of CSE, SRCEM, [2]Assistant Professor, Dept. of CSE, SRCEM

[1]hemantsharma8120@gmail.com, [2]shyamolgwl@gmail.com

***Abstract*-Online web forums have become vibrant spaces for exchanging knowledge, solving problems, and engaging in user-led discussions. However, their unstructured and noisy content creates considerable difficulties for automatically extracting useful answers. As the volume of user-generated data continues to expand, identifying responses that are relevant, accurate, and reliable is increasingly important for improving information retrieval systems.**

This review explores the significance of feature selection in Automatic Answer Extraction (AAE), emphasizing how the careful choice of meaningful linguistic, semantic, structural, and user-related features can greatly enhance system performance. Feature selection plays a key role in reducing dimensionality, eliminating noise, improving interpretability, and increasing computational efficiency, particularly in large datasets.

The study examines a wide range of features, including part-of-speech tags, n-grams, word embeddings, semantic similarity metrics, thread position, user credibility, and voting behavior, and discusses their impact in both traditional machine learning and deep learning frameworks. It also reviews advanced feature selection techniques such as filter, wrapper, and embedded methods, highlighting their relevance to answer extraction tasks.

Keywords- Feature Selection, Automatic Answer Extraction, Online Web Forums, Information Retrieval, Text Mining

I. INTRODUCTION

Community-driven question-answering platforms such as Stack Overflow, Quora, and Reddit have become prominent sources of shared knowledge, where users contribute questions and answers across diverse domains. These platforms collectively form a dynamic knowledge ecosystem. However, due to their unstructured and heterogeneous nature, extracting meaningful information automatically remains a challenging task.

Automatic Answer Extraction (AAE), a subfield of natural language processing and information retrieval, aims to identify relevant answers from such discussions. The success of AAE systems largely depends on the selection of appropriate features that effectively represent textual, semantic, and structural characteristics of the data (Jain et al.).

Feature selection plays a critical role by reducing redundant and irrelevant information, improving model accuracy, and enhancing computational efficiency. Poor feature selection may lead to overfitting, increased processing time, and reduced generalization capability. Various feature types—including lexical features (e.g., n-grams, POS tags), semantic embeddings (e.g., Word2Vec, BERT), structural attributes (e.g., answer position), and user-related signals (e.g., reputation, votes)—contribute to identifying high-quality answers (fig. 1).

However, incorporating all features without proper selection may introduce noise and redundancy. Therefore, techniques such as Information Gain, Chi-Square, Mutual Information, and Principal Component Analysis are widely used to select the most relevant features, thereby improving performance in answer extraction tasks (Gaikwad et al.).

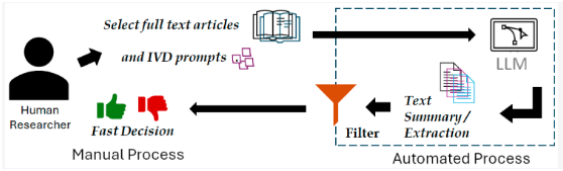

Fig. 1 Manual and Automated Processes for Text Extraction[3]

II. LITERATURE REVIEW

Recent studies have explored different aspects of feature selection and answer extraction in online forums. Zoratto et al. demonstrated that readability, contextual placement, and user interaction significantly influence best-answer prediction. Their findings suggest that combining linguistic and structural features improves classification accuracy.

Similarly, Hasnain et al. reviewed ensemble learning approaches for web service classification and highlighted their effectiveness in improving prediction accuracy and robustness.

Meyer and Elsweiler introduced a German dataset focused on health behavior change, emphasizing the importance of theory-driven feature selection in conversational systems.

Mubarak et al. applied machine learning techniques to predict student dropout in online learning environments, demonstrating how behavioral and temporal features can enhance predictive performance.

Nazah et al. proposed an unsupervised approach for analyzing Dark Web forums, achieving high accuracy in detecting patterns without relying on labeled data, thus highlighting scalability and adaptability challenges.

TABLE 1 LITERATURE SUMMARY

Author/ Year	Methodology	Findings	Research Gaps	Limitations
Sajid, Hasan & Ibrahim (2024) [13]	Learning-to-Rank in CQA; features from both question & answer; combines IR (TF, IDF, BM25) + BERT semantic similarity; tested on three standard datasets.	Achieved state-of-the-art NDCG@10 scores; inclusion of answer-side.	Exploration needed for feature selection methods (filter/wrapper/embedded) specifically in this context; generalization to low-resource or multilingual forums.	Datasets are standard CQA ones — domain bias; potential overfitting to dataset specifics
Pushpa Rani, Vidyullatha& Srinivas Rao (2024) [14]	Topic modeling with Dove Optimization, Gibbs sampling to select topic features for QA; web-based question datasets.	Optimized topic modeling helps in selecting relevant answers; topic-based features are strong indicators of relevance.	Need to compare with deep semantic embeddings & user-based features; little discussion of feature interpretability or user metrics.	Focused on web-based general QA; may not address forum thread structure, nor user reputation; optimization algorithm cost might be high.
Chau, 2020 [15]	Case study; qualitative; AISAS model; digital forums & interviews with users (Sociolla, Sociolla's forums, reviews, social media).	Marketing via forum allows audience to collect & share information; online discussions correspond well with AISAS model.	Lacks quantitative measurement of feature impacts; limited to one brand/product.	Single platform, single brand; qualitative findings may not generalize.
Zhang, 2020 [16]	Developed fact-checking dataset & neural model for answer veracity in product Q&A forums.	New dataset; their model outperforms baselines in veracity prediction.	Integration with features beyond text & evidence (e.g., user features, forum structure) remains under-explored.	Restricted to product Q&A; perhaps skewed data; generalization to other domains unclear.
Khan, 2020 [17].	Answer extraction as classification (relevant/partial/irrelevant); lexical & non-lexical features; LinearSVC; chi-square & univariate selection.	LinearSVC works well; feature selection (chi-square, univariate) helps reduce feature space and maintain performance.	Semantic/contextual embeddings not used; deeper analysis of noisy or unstructured data needed.	Only two datasets (Ubuntu, TripAdvisor); possible domain dependency; older feature types.

III. AUTOMATIC ANSWER EXTRACTION FROM ONLINE WEB FORUMS

Automatic Answer Extraction (AAE) from online web forums is a big step forward in the fields of information retrieval and natural language processing. Its goal is to automatically find and extract the most relevant answers from vast amounts of community-driven debates. Quora, Stack Overflow, Reddit, and Yahoo! Answers are just a few of the many online platforms where users can ask and answer questions. These sites have huge collections of user-generated information. These repositories are great sources of knowledge, but the fact that the content is informal and unstructured makes them hard to work with. Responses can include unrelated comments, extra information, slang, abbreviations, or phrases that aren't complete. Threads, on the other hand, sometimes have a mix of correct, partially correct, and wrong replies. Because of this, it can take a long time to find good replies by hand. This is why we need automated systems that can filter and extract excellent responses with little help from people[18].

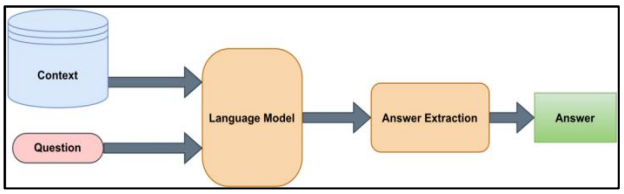

Fig. 2 Answer Extraction[19]

AAE works by figuring out how a question is related to the pool of possible answers in a thread. Early methods mostly used metrics of lexical similarity, such as keyword overlap, TF-IDF vectors, and cosine similarity. These methods are computationally efficient, but they don't work when synonyms or paraphrases are employed since they don't capture semantic equivalence. To address these deficiencies, research has transitioned to semantic and contextual methodologies, employing distributed word representations such as Word2Vec and GloVe, alongside contextual embeddings from transformer-based models like BERT and RoBERTa, to convey meaning that transcends superficial text. These models allow for better semantic alignment between questions and replies, which makes extraction better[20].

In addition to language and semantic clues, the structure of forum conversations is very important to AAE. You can tell if an answer is good by looking at things like where it is in the thread, how long it is, whether it has hyperlinks, formatting indicators (like bullet points or code samples), and even when it was posted (early or late). User-based elements like reputation scores, a history of contributions, and endorsements through likes, upvotes, or "accepted answer" markings all give a layer of social validation that can help answer extraction algorithms. By combining these different traits, systems can better tell the difference between useful replies and noise[20].

TABLE 2. APPROACHES AND FEATURES IN AUTOMATIC ANSWER EXTRACTION

Approach	Description	Common Features Used	Strengths	Limitations
Rule-based Methods	Handcrafted rules or heuristics for answer identification	Keywords, patterns, structural cues (length, position)	Simple, interpretable, domain-specific	Poor scalability, domain dependence
Lexical Matching	Surface-level similarity between Q–A pairs	TF-IDF, cosine similarity, n-grams	Computationally efficient	Cannot capture semantic equivalence
Semantic Approaches	Embedding-based similarity for deeper semantic understanding	Word2Vec, GloVe, BERT embeddings	Captures meaning beyond keywords	High computational cost, domain adaptation
Machine Learning	Supervised models classify/rank answers as relevant or not	Linguistic (POS, n-grams), structural (length, position), user-based (reputation, votes)	Flexible, improved accuracy with feature sets	Requires labeled data, sensitive to feature noise
Deep Learning	Neural networks learn contextual representations automatically	Contextual embeddings, hybrid features	Strong generalization, reduced manual effort	Data-hungry, interpretability challenges
Hybrid Approaches	Combines statistical, semantic, structural, and user-based signals	Mixture of handcrafted + learned features	Best of both worlds, robust performance	Complex design, higher computational demands

The table shows the main ways that people use to automatically extract answers from online forums, along with their pros and cons. Rule-based and lexical approaches depend on patterns made by hand or surface similarities. They are simple but not very scalable. Semantic methods use embeddings to better capture meaning, but they are expensive to run. Machine learning models use a variety of linguistic, structural, and user-based criteria to sort or rank answers. Deep learning approaches, on the other hand, develop contextual representations on their own but need a lot of data. Hybrid methods use different sorts of features to get strong and reliable results. Overall, feature selection is still very important for all methods to find the right balance between efficiency and efficacy.

There are a number of methodological paradigms that support AAE research. Rule-based systems depend on manually created language or structural patterns. They can be understood, but they can't be scaled up. Machine learning techniques view AAE as a supervised classification or ranking issue, utilizing methods like SVM, RF and Logistic Regression on labeled datasets. These models depend a lot on well-designed features, therefore choosing the right ones is an important step in improving performance. Deep learning architectures like CNNs, RNNs, and transformers have been more popular in recent years because they can automatically learn hierarchical and contextual representations. Hybrid pipelines that mix handcrafted features with deep embeddings have worked better in community question answering. This shows how explicit feature selection and implicit representation learning may work together. Even while AAE has made a lot of progress, it is still not an easy process. The fact that forums are made up of people from different backgrounds makes it harder to extract information since they use different languages, jargon, and data that is combined with code. Also, the subjective character of "best answers" changes depending on the situation. For example, what is regarded most relevant in a technical forum may be very different from what is deemed most relevant in a health or legal forum. As forums grow at an alarming rate, scalability and computing efficiency are becoming more and more important. At the same time, fairness and interpretability are also important to make sure that systems don't put too much value on popularity or unintentionally leave out individuals who aren't as well-known. To deal with these problems, we need strong feature selection algorithms, scalable structures, and models that find the right balance between semantic depth and transparency[21].

IV. FEATURE SELECTION IN ANSWER EXTRACTION

Feature selection is a key part of automated answer extraction (AAE) from online forums since it affects how well computers can filter out noise, find meaningful signals, and give correct results. Responses in online forums can be very different in terms of length, syntax, meaning, and user purpose. This makes them very unstructured places. If you don't choose the right features, your models could be flooded with unnecessary or duplicate variables, which would make them less accurate, overfit them, and cost a lot of money to run. Well-chosen characteristics, on the other hand, lower dimensionality, make things more efficient, and make them easier to understand. This lets researchers figure out which attributes have the biggest effect on finding the

right answers. Feature selection is important since it helps both classic machine learning methods and deep learning pipelines by making them easier to understand and more scalable when dealing with a lot of data[22]–[24].

There are four main types of characteristics that AAE uses: linguistic, semantic, structural, and user-based. Linguistic characteristics look at the grammatical and textual parts of replies, such as part-of-speech (POS) tags, n-grams, syntactic patterns, and stylistic markers like punctuation or capitalization. These assist tell the difference between noise and content that sounds like an answer. For example, answers are more likely to have declarative sentences or technical phrases, while comments that aren't relevant commonly employ informal or conversational tones. Semantic characteristics go beyond the surface level to find meaning. They use distributed word representations (Word2Vec, GloVe) and contextual embeddings (BERT, RoBERTa) to find how comparable the meanings of questions and replies are. These properties are very important for dealing with synonyms, paraphrase, and language that is specialized to a certain field. Structural elements focus on how forum threads are put together, like where an answer is, how long it is, if it has links, how it is formatted (for example, lists or code snippets), or even time indicators like early vs. late responses. These kinds of traits frequently go hand in hand with better solutions, especially in technical societies. Finally, user-based features incorporate contributor metadata including reputation, competence, past contributions, and community endorsements (likes, upvotes, and approved answers). These signals show social trust and knowledge, which makes them a great addition to text-based features[25]–[29]. To help you better grasp what these types of features do, the table below lists their main traits, contributions, and limitations in AAE research:

TABLE 3. TYPES OF FEATURES IN AUTOMATIC ANSWER EXTRACTION

Feature Type	Examples	Role in AAE	Advantages	Limitations
Linguistic	POS tagging, n-grams, syntactic patterns, punctuation, capitalization	Identifies grammatical and stylistic cues of answer-like responses	Simple, interpretable, useful in noisy contexts	Limited semantic depth
Semantic	Word2Vec, GloVe, BERT embeddings, semantic similarity, topic modeling	Captures meaning and context beyond keyword matching	Handles synonyms/paraphrases, domain adaptability	Computationally expensive, domain adaptation
Structural	Answer position, length, hyperlinks, code snippets, formatting, timestamps	Reflects organizational and presentation aspects of answers	Enhances quality detection, captures thread dynamics	May vary across domains and forum designs
User-based	Reputation, expertise, history of contributions, upvotes, likes, acceptance	Incorporates social validation and contributor credibility	Strong reliability signals, complements textual features	Biased toward popular users, ignores new experts

This taxonomy shows how several aspects work together to make answer extraction systems more reliable. For example, a brief but meaningful answer from a user with a good reputation may be better than a long but irrelevant one. This shows how linguistic, semantic, structural, and user-based clues often work together. But adding all conceivable features without thinking is not helpful. There may be too many attributes that are the same (for example, lexical overlap and semantic embeddings both showing how similar two pieces of text are) or not useful (for example, answers that are too brief and don't give much information). This is when feature selection methods come in handy. They assist you get rid of variables that aren't helpful and focus on the ones that are[30]–[32].

There is a big difference between feature selection and feature engineering. Feature engineering is the process of making new features or changing raw data into useful representations. For example, you might make n-grams, develop syntactic rules, or make embeddings. Feature selection, on the other hand, is the process of cutting down this larger number of features to just those that improve predicted accuracy the most. Engineering and selection work together to make a better pipeline. Engineering improves how data is represented, while selection makes sure that the process is quick and focused. The significance of feature selection fluctuates according on the modeling paradigm. In traditional machine learning models like Logistic Regression, SVMs, or Random Forests, the features you choose are very important because these algorithms depend a lot on the quality of their inputs. Extra or noisy features can make performance worse, make things more complicated, and make it harder to understand. Deep learning models, especially transformers, are made to automatically learn rich contextual features from raw text. However, feature selection is still important in neural settings for three reasons: (1) handcrafted features like user reputation or structural cues add non-textual information to embeddings; (2) selected features cut down on the amount of computation needed for large-scale training; and (3) selection makes it easier to understand which non-linguistic signals (like upvotes or position) help find the best answers. Hybrid models that mix neural embeddings with carefully chosen handcrafted features have consistently done better than end-to-end deep models when it comes to answering questions in a community[33]–[37].

V. FEATURE SELECTION TECHNIQUES

Feature selection techniques are methods used to pick a subset of relevant attributes (features) from the full set engineered for automatic answer extraction. These techniques help improve learning by reducing overfitting, lowering computational cost, enhancing generalization, and providing interpretability. Several classes of techniques are commonly used: filter methods, wrapper methods, embedded methods, and ensemble / hybrid methods. Below are descriptions of each, along with examples (including in QA / answer-extraction tasks) and when they are most appropriate.

1. Filter Methods

Filter approaches choose features based on their statistical qualities, not on a specific predictive model. They use measures like Information Gain (IG), Chi-Square test, Fisher score, Pearson correlation, or ANOVA to figure out how relevant characteristics are on their own. For instance, Information Gain and Chi-Square have been commonly employed to rank features (words, n-grams) in text categorization tasks prior to their input into classifiers. These techniques are quick and perform well with large amounts of data (like a lot of candidate text features), but they don't take into account how features interact with each other or how they operate together in the learning procedure. Information Gain: This tells you how much a feature helps to make the answer label less unclear. The Chi-Square test checks to see if the occurrences of a feature are not related to the class label. Features that are strongly linked to the label are kept. A lot of research on generic text categorization shows that these metrics work well. For example, "Advancements in feature selection and extraction methods for text mining: a review" talks about filter, wrapper, and embedded methods and lists IG, Chi-Square, and other metrics[38].

2. Wrapper Methods

Wrapper methods use a predictive model as a "wrapper" to test how well different groups of characteristics work together. They look through different combinations of features using methods like forward selection, backward elimination, or recursive feature elimination (RFE). Wrappers generally yield superior performance compared to pure filter approaches, as they consider feature interaction and the impact of feature combinations on model correctness. Their main problem is that they cost a lot of computing power, especially when the feature space is big. For example, recursive feature removal is a common method. A model (such a Support Vector Machine or logistic regression) is trained, features are rated, the least important ones are deleted, and the process is repeated until the desired amount of features is left. The article "Ensemble Filter-Wrapper Text Feature Selection Methods for Text Classification" examines wrapper, filter, and embedding methods. RFE is one of the most effective wrapper-based methods [39].

3. Embedded Methods

Embedded approaches choose features while training the model. The model itself either ignores or gives little weight to features that aren't important by using penalties or internal measurements of feature relevance. Some common embedding techniques are LASSO (L1 regularization), Decision Trees / Random Forests (feature importance through splits), and regularized logistic regression. These strategies find a middle ground between being effective and being efficient. For example, tree-based models are employed in various machine-learning pipelines for QA or community question answering (CQA) to not only classify and rank things, but also to show which attributes (structural, user, textual) are important. Embedded approaches are very useful when there are a lot of possible features, yet some of them are redundant, correlated, or noisy.

4. Hybrid / Ensemble Techniques

Many studies employ hybrid or ensemble feature selection to get the best of both worlds. For example, they might initially use a filter to get rid of features that aren't plainly useful, and then use wrapper or embedding methods to make the feature set even better. Ensemble feature selection combines the findings of several feature selection methods to make them more reliable and less biased. Meta-heuristic optimization methods, like genetic algorithms and PSO, can also be used as wrappers or ensembling strategies to choose features when there are limits. A recent study titled "Feature Importance Feedback with Deep Q Process in Ensemble-Based Metaheuristic Feature Selection Algorithms" combines metaheuristic techniques and embedded feedback loops to determine feature subsets[40].

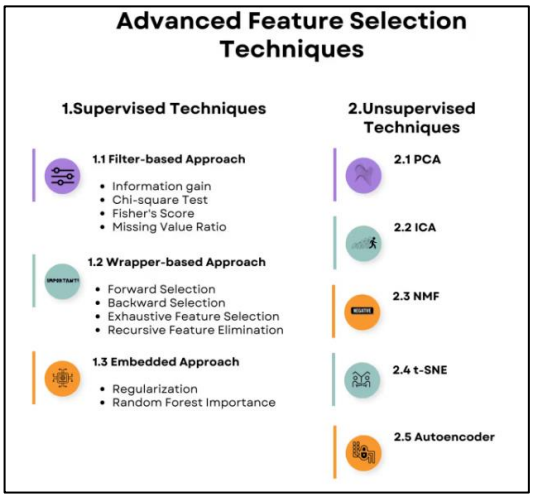

Fig. 3 Advanced Feature Selection techniques[13]

5. Case Studies & Examples

Sharma and Mittal (2018) in their study “Answer Extraction in Question Answering using Structure Features and Dependency Principles” (arXiv) explored the integration of lexical, syntactic, semantic, and structural features for answer extraction. Their results demonstrated that structural features derived from dependency parsing significantly improved answer identification by capturing relationships beyond surface-level similarity. This highlights the importance of syntactic and dependency cues in filtering correct answers from noisy forum content [41]. Similarly,

"A Study on Influential Features for Predicting Best Answers in Community Question-Answering Forums" (MDPI) analyzed both textual and non-textual indicators, such as readability, user interactions, and presence of code examples, in StackOverflow discussions. Findings revealed that non-textual factors, especially community signals like upvotes and contributor expertise, strongly correlated with best-answer selection. Together, these case studies emphasize the complementary role of linguistic, structural, and user-based features in effective answer extraction and underline the need for robust feature selection strategies.

VI. CHALLENGES AND RESEARCH GAPS

One big problem with choosing features for automatic answer extraction is that it doesn't work well with enormous amounts of forum data. Every day, forums create a lot of content, and response extraction systems have to go through millions of threads, which often have long answers, images/code snippets, and multiple conversations inside them. Traditional feature extraction and selection techniques, particularly wrapper methods or exhaustive embedding techniques, are computationally expensive and may not scale to large data quantities without distributed or approximate methods. Also, because forum content changes all the time (new threads, new subjects), systems need to change their feature sets over time. This is also a problem when it comes to handling data that is in more than one language or has code mixed in. Many forums have posts that are written in more than one language, postings that are written in more than one language, transliterations, or spelling that isn't standard. Transformer-based models offer some support for multilingualism (e.g. multilingual BERT, XLM-R) but still often perform poorly in code-mixed or low-resource language settings. For example, studies on complex named entity recognition in multilingual/code-mixed text (MultiCoNER) have shown deficiencies in entity detection when morphological and script variability increases, especially under limited training data [42].

Another significant gap lies in bias, fairness, and interpretability of the selected features. Feature selection strategies may favor attributes associated with socially or structurally privileged users (e.g., individuals with high reputation or numerous upvotes), so exacerbating popularity bias and marginalizing content from new or less active contributors. Also, it is sometimes hard to grasp the selected feature sets in deep learning or embedded methods: while transformers learn representations, it is hard to connect those back to features that people can understand. This is very important in fields where trust is very important, like medical or legal forums. Lastly, there's the issue of how to mix explicit feature selection (linguistic, structural, user-based) with the representations that transformers learn. This is especially important for transformer-based models like BERT and RoBERTa. Transformers can learn a lot of patterns without being told to, but explicit feature selection can give them extra signals (like thread position or user reputation. However, it is still an open question how to efficiently add these signals to end-to-end neural pipelines and make sure they improve performance without causing overfitting or redundancy.

VII. CONCLUSION

This review highlights the critical role of feature selection in enhancing Automatic Answer Extraction systems. While deep learning models have improved representation learning, explicit feature selection remains essential for handling noise, improving efficiency, and incorporating non-textual information.

Hybrid approaches combining handcrafted features with neural models show superior performance, particularly in complex and noisy environments. However, challenges such as scalability, bias, multilingual adaptability, and interpretability persist.

Future research should focus on developing adaptive, fair, and interpretable feature selection techniques that integrate seamlessly with modern deep learning frameworks.

REFERENCES

[1] A. K. Jain, S. R. Sahoo, and J. Kaubiyal, "Online social networks security and privacy: comprehensive review and analysis," *Complex Intell. Syst.*, vol. 7, no. 5, pp. 2157–2177, 2021, doi: 10.1007/s40747-021-00409-7.

[2] M. Gaikwad, S. Ahirrao, S. Phansalkar, and K. Kotecha, "Online Extremism Detection: A Systematic Literature Review with Emphasis on Datasets, Classification Techniques, Validation Methods, and Tools," *IEEE Access*, vol. 9, no. C, pp. 48364–48404, 2021, doi: 10.1109/ACCESS.2021.3068313.

[3] A. Ye, A. Maiti, M. Schmidt, and S. J. Pedersen, "A Hybrid Semi-Automated Workflow for Systematic and Literature Review Processes with Large Language Model Analysis," *Futur. Internet 2024, Vol. 16, Page 167*, vol. 16, no. 5, p. 167, May 2024, doi: 10.3390/FI16050167.

[4] E. Tseng *et al.*, "The tools and tactics used in intimate partner surveillance: An analysis of online infidelity forums," *Proc. 29th USENIX Secur. Symp.*, pp. 1893–1909, 2020.

[5] M. M. Tadesse, H. Lin, B. Xu, and L. Yang, "Detection of suicide ideation in social media forums using deep learning," *Algorithms*, vol. 13, no. 1, pp. 1–19, 2020, doi: 10.3390/a13010007.

[6] A. Shrestha, E. Serra, and F. Spezzano, "Multi-modal social and psycho-linguistic embedding via recurrent neural networks to identify depressed users in online forums," *Netw. Model. Anal. Heal. Informatics Bioinforma.*, vol. 9, no. 1, 2020, doi: 10.1007/s13721-020-0226-0.

[7] A. S. Lan, J. C. Spencer, Z. Chen, C. G. Brinton, and M. Chiang, "Personalized thread recommendation for mooc discussion forums," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 11052 LNAI, pp. 725–740, 2019, doi: 10.1007/978-3-030-10928-8_43.

[8] V. Zoratto, D. Godoy, and G. N. Aranda, "A Study on Influential Features for Predicting Best Answers in Community Question-Answering Forums," *Inf. 2023, Vol. 14, Page 496*, vol. 14, no. 9, p. 496, Sep. 2024, doi: 10.3390/INFO14090496.

[9] M. Hasnain, I. Ghani, S. R. Jeong, and A. Ali, "Ensemble learning models for classification and selection of web services: A review," *Comput. Syst. Sci. Eng.*, vol. 40, no. 1, pp. 327–339, 2024, doi: 10.32604/CSSE.2022.018300.

[10] S. Meyer and D. Elsweiler, “GLoHBCD: A Naturalistic German Dataset for Language of Health Behaviour Change on Online Support Forums,” *2022 Lang. Resour. Eval. Conf. Lr. 2022*, no. June, pp. 2226–2235, 2022.

[11] A. A. Mubarak, H. Cao, and W. Zhang, “Prediction of students’ early dropout based on their interaction logs in online learning environment,” *Interact. Learn. Environ.*, vol. 30, no. 8, pp. 1414–1433, 2022, doi: 10.1080/10494820.2020.1727529.

[12] S. Nazah, S. Huda, J. H. Abawajy, and M. M. Hassan, “An Unsupervised Model for Identifying and Characterizing Dark Web Forums,” *IEEE Access*, vol. 9, pp. 112871–112892, 2021, doi: 10.1109/ACCESS.2021.3103319.

[13] N. Sajid, M. R. Hasan, and M. Ibrahim, “Feature engineering in learning-to-rank for community question answering task,” *Int. J. Comput. Appl.*, vol. 46, no. 8, pp. 555–566, Aug. 2024, doi: 10.1080/1206212X.2024.2380647.

[14] K. Pushpa Rani, P. Vidyullatha, and K. Srinivas Rao, “An optimized topic modeling question answering system for web-based questions,” *Multimed. Tools Appl.*, vol. 83, no. 27, pp. 69581–69599, Aug. 2024, doi: 10.1007/S11042-024-18166-3/METRICS.

[15] M. Chau, T. M. H. Li, P. W. C. Wong, J. J. Xu, P. S. F. Yip, and H. Chen, “Finding people with emotional distress in online social media: A design combining machine learning and rule-BASED classification,” *MIS Q. Manag. Inf. Syst.*, vol. 44, no. 2, pp. 933–956, 2020, doi: 10.25300/MISQ/2020/14110.

[16] W. Zhang, Y. Deng, J. Ma, and W. Lam, “AnswerFact: Fact checking in product question answering,” *EMNLP 2020 - 2020 Conf. Empir. Methods Nat. Lang. Process. Proc. Conf.*, pp. 2407–2417, 2020, doi: 10.18653/v1/2020.emnlp-main.188.

[17] A. Khan *et al.*, “Machine Learning Approach for Answer Detection in Discussion Forums: An Application of Big Data Analytics,” *Sci. Program.*, vol. 2020, 2020, doi: 10.1155/2020/4621196.

[18] F. Martin, C. Wang, and A. Sadaf, “Facilitation matters: Instructor perception of helpfulness of facilitation strategies in online courses,” *Online Learn. J.*, vol. 24, no. 1, pp. 28–49, 2020, doi: 10.24059/olj.v24i1.1980.

[19] P. Nakov, L. Màrquez, W. Magdy, A. Moschitti, J. Glass, and B. Randeree, “SemEval-2015 Task 3: Answer Selection in Community Question Answering,” *SemEval 2015 - 9th Int. Work. Semant. Eval. co-located with 2015 Conf. North Am. Chapter Assoc. Comput. Linguist. Hum. Lang. Technol. NAACL-HLT 2015 - Proc.*, pp. 269–281, 2015, doi: 10.18653/v1/s15-2047.

[20] A. Melo and H. Paulheim, “Local and global feature selection for multilabel classification with binary relevance: An empirical comparison on flat and hierarchical problems,” *Artif. Intell. Rev.*, vol. 51, no. 1, pp. 33–60, 2019, doi: 10.1007/s10462-017-9556-4.

[21] W. T. Yue, Q. H. Wang, and K. L. Hui, “See no evil, hear no evil? Dissecting the impact of online hacker forums,” *MIS Q. Manag. Inf. Syst.*, vol. 43, no. 1, pp. 73–95, 2019, doi: 10.25300/MISQ/2019/13042.

[22] E. Papegnies, V. Labatut, R. Dufour, and G. Linarès, “Conversational Networks for Automatic Online Moderation,” *IEEE Trans. Comput. Soc. Syst.*, vol. 6, no. 1, pp. 38–55, 2019, doi: 10.1109/TCSS.2018.2887240.

[23] P. Atanasova *et al.*, “Automatic fact-checking using context and discourse information,” *J. Data Inf. Qual.*, vol. 11, no. 3, 2019, doi: 10.1145/3297722.

[24] T. Chakrabarty, C. Hidey, S. Muresan, K. McKeown, and A. Hwang, “AmperSand: Argument mining for persuasive online discussions,” *EMNLP-IJCNLP 2019 - 2019 Conf. Empir. Methods Nat. Lang. Process. 9th Int. Jt. Conf. Nat. Lang. Process. Proc. Conf.*, pp. 2933–2943, 2019, doi: 10.18653/v1/d19-1291.

[25] T. Zhang, D. Yang, C. Lopes, and M. Kim, “Analyzing and Supporting Adaptation of Online Code Examples,” *Proc. - Int. Conf. Softw. Eng.*, vol. 2019-May, pp. 316–327, 2019, doi: 10.1109/ICSE.2019.00046.

[26] J. Wang, R. Wen, C. Wu, Y. Huang, and J. Xiong, “FDGars: Fraudster detection via graph convolutional networks in online app review system,” *Web Conf. 2019 - Companion World Wide Web Conf. WWW 2019*, pp. 310–316, 2019, doi: 10.1145/3308560.3316586.

[27] H. Phan, H. A. Nguyen, N. M. Tran, L. H. Truong, A. T. Nguyen, and T. N. Nguyen, “Statistical learning of API fully qualified names in code snippets of online forums,” *Proc. - Int. Conf. Softw. Eng.*, pp. 632–642, 2018, doi: 10.1145/3180155.3180230.

[28] B. Desmet and V. Hoste, “Online suicide prevention through optimised text classification,” *Inf. Sci. (Ny).*, vol. 439–440, pp. 61–78, 2018, doi: 10.1016/j.ins.2018.02.014.

[29] L. Battle, P. Duan, Z. Miranda, D. Mukusheva, R. Chang, and M. Stonebraker, “Beagle: Automated extraction and interpretation of visualizations from the web,” *Conf. Hum. Factors Comput. Syst. - Proc.*, vol. 2018-April, pp. 1–8, 2018, doi: 10.1145/3173574.3174168.

[30] S. Ji, C. P. Yu, S. F. Fung, S. Pan, and G. Long, “Supervised learning for suicidal ideation detection in online user content,” *Complexity*, vol. 2018, 2018, doi: 10.1155/2018/6157249.

[31] H. C. Shing, S. Nair, A. Zirikly, M. Friedenberg, H. Daumé, and P. Resnik, “Expert, crowdsourced, and machine assessment of suicide risk via online postings,” *Proc. 5th Work. Comput. Linguist. Clin. Psychol. From Keyboard to Clin. CLPsych 2018 2018 Conf. North Am. Chapter Assoc. Comput. Linguist. Hum. Lang. Technol.*, pp. 25–36, 2018, doi: 10.18653/v1/w18-0603.

[32] S. Pastrana, A. Hutchings, A. Caines, and P. Buttery, “Characterizing eve: Analysing cybercrime actors in a large underground forum,” *Lect. Notes Comput. Sci.*, vol. 11050 LNCS, pp. 207–227, 2018, doi: 10.1007/978-3-030-00470-5_10.

[33] T. K. F. Chiu and T. K. F. Hew, “Factors influencing peer learning and performance in MOOC asynchronous online discussion forum,” *Australas. J. Educ. Technol.*, vol. 34, no. 4, pp. 16–28, 2018, doi: 10.14742/ajet.3240.

[34] T. Zhang, G. Upadhyaya, A. Reinhardt, H. Rajan, and M. Kim, “Are code examples on an online Q&A forum reliable?: A study of API misuse on stack overflow,” *Proc. - Int. Conf. Softw. Eng.*, pp. 886–896, 2018, doi: 10.1145/3180155.3180260.

[35] D. Hazarika, S. Poria, S. Gorantla, E. Cambria, R. Zimmermann, and R. Mihalcea, “CASCADE: Contextual sarcasm detection in online discussion forums,” *COLING 2018 - 27th Int. Conf. Comput. Linguist. Proc.*, pp. 1837–1848, 2018.

[36] S. Zhang, X. Zhang, H. Wang, L. Guo, and S. Liu, “Multi-scale attentive interaction networks for Chinese medical question answer selection,” *IEEE Access*, vol. 6, pp. 74061–74071, 2018, doi: 10.1109/ACCESS.2018.2883637.

[37] T. Mihaylova *et al.*, “Fact checking in community forums,” *32nd AAAI Conf. Artif. Intell. AAAI 2018*, pp. 5309–5316, 2018, doi: 10.1609/aaai.v32i1.11983.

[38] L. Ayash, A. Algarni, and O. Alqahtani, “Advancements in feature selection and extraction methods for text mining: a review,” *Discov. Appl. Sci.*, vol. 7, no. 8, pp. 1–43, Aug. 2025, doi: 10.1007/S42452-025-07587-W/TABLES/8.

[39] O. P. Ige and K. H. Gan, “Ensemble Filter-Wrapper Text Feature

Selection Methods for Text Classification," *Comput. Model. Eng. Sci.*, vol. 141, no. 2, pp. 1847–1865, Sep. 2024, doi: 10.32604/CMES.2024.053373.

[40] J. L. Potharlanka and M. Nirupama Bhat, "Feature importance feedback with Deep Q process in ensemble-based metaheuristic feature selection algorithms," *Sci. Rep.*, vol. 14, no. 1, p. 2923, Dec. 2024, doi: 10.1038/S41598-024-53141-W.

[41] L. K. Sharma and N. Mittal, "Answer Extraction in Question Answering using Structure Features and Dependency Principles," Oct. 2018, Accessed: Sep. 15, 2025. [Online]. Available: https://arxiv.org/pdf/1810.03918

[42] A. El Mekki, A. El Mahdaouy, M. Akallouch, I. Berrada, and A. Khoumsi, "UM6P-CS at SemEval-2022 Task 11: Enhancing Multilingual and Code-Mixed Complex Named Entity Recognition via Pseudo Labels using Multilingual Transformer," *SemEval 2022 - 16th Int. Work. Semant. Eval. Proc. Work.*, pp. 1511–1517, 2022, doi: 10.18653/V1/2022.SEMEVAL-1.207.