
Professional Certificate in Social Media Research Methods (United Kingdom)

Sentiment Analysis and Opinion Mining

Affective Computing – related terms: Emotion detection, affective AI. A discipline that enables computers to recognize, interpret, and simulate human emotions. In sentiment analysis, affective computing techniques are used to infer emotional states from text, voice, or facial expressions. Example: An AI system classifies a tweet as expressing “joy” by detecting positive lexical cues and an exclamation mark. Practical applications include customer service chatbots that adapt tone based on user sentiment, and mental-health monitoring tools that flag depressive language. Challenges involve cultural variability in emotional expression, the subtlety of sarcasm, and the need for large, annotated datasets to train reliable models.

Aspect-Based Sentiment Analysis (ABSA) – related terms: Feature-level sentiment, fine-grained analysis. ABSA isolates sentiment toward specific attributes (aspects) of a product or service rather than providing an overall polarity. Example: In a restaurant review, “The pasta was excellent but the service was slow,” ABSA tags “pasta” with positive sentiment and “service” with negative sentiment. This granularity supports targeted improvements, such as menu refinement or staff training. Difficulties include correctly extracting aspect terms, handling implicit aspects (“The ambiance was cozy”), and dealing with overlapping aspects in complex sentences.

Bag-of-Words (BoW) – related terms: Vector space model, term frequency. BoW is a simple representation that records the frequency of each word in a document while disregarding grammar and word order. In sentiment analysis, BoW vectors feed classifiers such as Naïve Bayes or Support Vector Machines. Example: A review containing “great,” “fantastic,” and “poor” yields a vector where “great” and “fantastic” have high positive weights, while “poor” carries negative weight. BoW is easy to implement but suffers from sparsity, inability to capture context, and vulnerability to synonyms or polysemy.

Binary Classification – related terms: Two-class problem, positive/negative labeling. Binary classification assigns each text to one of two categories, typically “positive” or “negative” sentiment. Algorithms such as Logistic Regression, Decision Trees, and Neural Networks are commonly employed. Example: A model predicts that a product review is “negative” because it contains more negative adjectives than positive ones. While binary classification simplifies evaluation, it overlooks neutral or mixed sentiments, reduces nuance, and may misclassify ambiguous statements.

BERT (Bidirectional Encoder Representations from Transformers) – related terms: Transformer models, contextual embeddings. BERT is a deep-learning language model that captures bidirectional context, enabling superior performance on sentiment tasks. Fine-tuning BERT on a domain-specific dataset yields state-of-the-art accuracy. Example: A fine-tuned BERT model correctly interprets the sarcastic tweet “I just love waiting in line for hours,” recognizing negative sentiment despite the word “love.” Challenges include high computational cost, requirement for large annotated corpora, and the risk of overfitting on small datasets.

Bigram – related terms: N-gram, collocation. A bigram is a pair of consecutive words used to capture local context that single words miss. In sentiment analysis, bigrams help detect phrases like “not good” or “very happy.” Example: The bigram “not happy” receives a negative polarity score, overriding the positive polarity of “happy” alone. However, bigrams increase feature dimensionality and may still miss longer dependencies or idiomatic expressions.

Class Imbalance – related terms: Skewed distribution, minority class. Class imbalance occurs when one sentiment class (e.G., Positive) dominates the training data, causing models to bias toward the majority. Techniques such as oversampling, undersampling, or synthetic data generation (SMOTE) mitigate the issue. Example: A dataset with 90% positive reviews leads a classifier to achieve 90% accuracy by always predicting “positive,” yet it fails to detect negative feedback. Addressing imbalance improves recall for the minority class and yields more trustworthy sentiment monitoring.

Co-occurrence Matrix – related terms: Term-term association, word-embedding precursor. A co-occurrence matrix records how often pairs of words appear within a defined window, providing statistical information for constructing embeddings or sentiment lexicons. Example: The words “delicious” and “taste” co-occur frequently in positive restaurant reviews, strengthening their association in a sentiment model. Limitations include high memory usage for large vocabularies and inability to capture deeper syntactic relationships.

Confusion Matrix – related terms: Performance metrics, true/false positives. A confusion matrix tabulates correct and incorrect predictions across classes, enabling calculation of precision, recall, and F1-score. In sentiment analysis, it reveals whether a model confuses “neutral” with “positive,” for instance. Example: A model that misclassifies 30% of neutral tweets as positive will show high recall for positive but low precision. Understanding these trade-offs guides model refinement and threshold adjustment.

Contextual Embedding – related terms: Dynamic word vectors, transformer embeddings. Contextual embeddings generate word representations that change depending on surrounding text, unlike static embeddings (e.G., Word2Vec). They capture polysemy, making sentiment analysis more accurate for ambiguous terms. Example: The word “cold” receives a different vector in “cold drink” (neutral) versus “cold attitude” (negative). Implementing contextual embeddings improves nuance detection but demands substantial computational resources and careful fine-tuning.

Cross-Domain Sentiment Transfer – related terms: Domain adaptation, transfer learning. This approach transfers a sentiment model trained on one domain (e.G., Movie reviews) to another (e.G., Hotel reviews) where labeled data are scarce. Methods include feature alignment, adversarial training, and fine-tuning on limited target data. Example: A model trained on Amazon product reviews is adapted to analyze airline feedback, preserving general sentiment cues while learning domain-specific terminology. Challenges involve vocabulary shift, differing aspect importance, and maintaining performance without overfitting to the source domain.

Data Pre-processing – related terms: Tokenisation, stop-word removal, stemming. Pre-processing prepares raw social-media text for analysis by cleaning noise (URLs, emojis), normalising case, and segmenting into tokens. Example: Removing “http://” links from a tweet prevents the model from misclassifying URL tokens as sentiment-bearing words. Over-aggressive cleaning can discard useful signals (e.G., Emojis that convey

sentiment), so a balanced pipeline is essential.

Deep Learning – related terms: Neural networks, representation learning. Deep learning models automatically learn hierarchical features from raw text, often outperforming traditional machine-learning methods in sentiment tasks. Architectures include Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and Transformers. Example: A CNN captures local n-gram patterns, while an RNN models sequential dependencies, both contributing to accurate sentiment detection. Drawbacks include opacity (harder to interpret), data hunger, and susceptibility to bias present in training corpora.

Dependency Parsing – related terms: Syntactic analysis, grammatical relations. Dependency parsing identifies the grammatical structure of a sentence, linking words to their heads. Sentiment analysis leverages this to locate negation scopes, intensifiers, and aspect-sentiment pairs. Example: In “The film was not only boring but also predictable,” parsing reveals that “not only” intensifies the negative sentiment attached to “boring” and “predictable.” Parsing errors, especially on informal social-media text, can degrade sentiment accuracy.

Document-Level Sentiment – related terms: Overall polarity, holistic analysis. Document-level sentiment assigns a single polarity to an entire piece of text, such as a blog post or long review. Techniques often aggregate sentence scores or use hierarchical models that first encode sentences then combine them. Example: A news article praising a policy’s benefits receives a positive document-level label, despite occasional neutral factual statements. Limitations include loss of aspect-specific insights and difficulty handling mixed-sentiment documents.

Ensemble Methods – related terms: Model stacking, bagging, boosting. Ensembles combine predictions from multiple classifiers to improve robustness and accuracy. In sentiment analysis, an ensemble might merge a Logistic Regression model, a CNN, and a BERT-based classifier. Example: Voting across models reduces variance, yielding higher F1-score on a test set of tweets. However, ensembles increase computational cost and can be harder to interpret.

Feature Engineering – related terms: Handcrafted features, lexical cues. Feature engineering creates informative variables from raw text, such as sentiment lexicon scores, part-of-speech counts, or punctuation frequency. Example: Counting exclamation marks improves detection of heightened emotion in social-media posts. While powerful for traditional models, extensive manual feature design is less critical for deep-learning approaches that learn representations automatically.

Fine-Tuning – related terms: Transfer learning, domain adaptation. Fine-tuning adapts a pre-trained language model (e.g., BERT) to a specific sentiment dataset by continuing training on task-specific examples. Example: A BERT model pre-trained on Wikipedia is fine-tuned on a corpus of customer reviews, resulting in higher accuracy than training from scratch. Risks include catastrophic forgetting of general language knowledge and over-fitting on small sentiment corpora.

F1-Score – related terms: Harmonic mean, precision-recall balance. The F1-Score combines precision and recall into a single metric, useful for imbalanced sentiment classes. Example: A classifier with precision 0.80 and recall 0.70 yields an F1-Score of 0.74, indicating balanced performance. Relying solely on accuracy can

mask poor detection of minority sentiments; F1 provides a more nuanced assessment.

GloVe (Global Vectors for Word Representation) – related terms: Static embedding, word-level vectors. GloVe generates word embeddings by factorising a global word-co-occurrence matrix, capturing semantic similarity. In sentiment analysis, GloVe vectors serve as input features for classifiers. Example: The words “happy” and “joyful” obtain similar vectors, aiding the model in generalising sentiment across synonyms. Static embeddings cannot differentiate word sense in different contexts, limiting performance on ambiguous phrases.

Hashtag Sentiment – related terms: Hashtag mining, topic sentiment. Hashtags often encapsulate sentiment or stance (e.G., #Lovelt, #Fail). Mining sentiment from hashtags complements text-based analysis, especially when the tweet body is short. Example: A tweet containing “#GreatService” reinforces a positive label even if the accompanying text is neutral. Challenges include sarcasm in hashtag use, evolving hashtag meanings, and multilingual variations.

Hybrid Approach – related terms: Rule-based + machine learning, combined methods. Hybrid models integrate lexical rule-based systems with statistical classifiers to leverage strengths of both. Example: A rule-based lexicon flags obvious sentiment words, while a machine-learning component resolves ambiguous cases and learns from data. This can improve coverage and interpretability but requires careful coordination to avoid conflicting predictions.

Intent Detection – related terms: Purpose classification, user goal inference. While not strictly sentiment, intent detection helps contextualise opinion mining by distinguishing complaints, praise, or queries. Example: A customer tweet “Why is my bill so high?” Is classified as a “complaint” intent, prompting a negative sentiment label and a routing rule for support. Integrating intent improves response prioritisation but adds another classification layer that must be trained accurately.

Lexicon-Based Methods – related terms: Sentiment dictionaries, rule-based analysis. Lexicon approaches assign polarity scores to words using pre-compiled lists such as SentiWordNet or VADER. Sentiment of a text is computed by aggregating these scores, often adjusted for negation and intensifiers. Example: The sentence “The movie was not terrible” receives a slightly positive score because the negation flips the negative term “terrible.” Lexicons are fast and interpretable but may lack coverage of slang, domain-specific jargon, and evolving language.

Long-Short Term Memory (LSTM) – related terms: Recurrent neural network, gated architecture. LSTM networks address the vanishing gradient problem in RNNs, enabling learning of long-range dependencies in text. They are widely used for sentiment classification of sequences, capturing context over many tokens. Example: An LSTM correctly links the sentiment cue “although” to the clause that follows, recognizing a mixed sentiment statement. Training LSTMs requires substantial data, and they can be slower than newer transformer models.

Machine Translation for Sentiment Transfer – related terms: Cross-lingual sentiment, multilingual models. Translating sentiment-laden text from one language to another before analysis can leverage high-resource language models. Example: A Spanish tweet is translated to English, then processed by an English-trained

BERT sentiment classifier. Translation errors may distort sentiment cues, especially idioms, so quality control is essential.

Negation Handling – related terms: Scope detection, polarity reversal. Negation words (e.G., “Not,” “never”) invert the polarity of nearby sentiment terms. Proper handling requires identifying the scope of negation. Example: In “The food is not good,” the model flips the positive polarity of “good” to negative. Simple heuristics (e.G., Flipping the next adjective) often fail on complex sentences, necessitating syntactic parsing or learned patterns.

Named Entity Recognition (NER) – related terms: Entity extraction, information tagging. NER identifies proper nouns (people, brands, locations) within text, enabling targeted sentiment analysis toward specific entities. Example: A tweet mentioning “Apple” and “Samsung” allows the system to assign separate sentiment scores to each brand. Errors in NER, especially with ambiguous or misspelled names common on social media, can misattribute sentiment.

Neutral Sentiment – related terms: Objective stance, non-opinionated. Neutral sentiment denotes a lack of clear positive or negative affect. Distinguishing neutral from weakly positive/negative is crucial for accurate monitoring. Example: “The device arrived on Monday” is labeled neutral because it conveys factual information. Models often over-classify neutral statements as positive or negative, reducing precision; calibrating thresholds and using fine-grained labels can mitigate this.

Opinion Mining – related terms: Sentiment analysis, stance extraction. Opinion mining encompasses the broader process of detecting, extracting, and summarising subjective information from text, including sentiment, attitudes, and arguments. Example: Analyzing a set of product reviews to produce a summary that highlights common praises (“battery life”) and complaints (“screen glare”). The field faces challenges such as sarcasm, domain drift, and the need for explainable outputs.

Part-of-Speech Tagging (POS) – related terms: Grammatical labeling, syntactic features. POS tagging assigns word categories (noun, verb, adjective) that help isolate sentiment-bearing adjectives and adverbs. Example: Recognising “great” as an adjective allows the model to apply its polarity score, while ignoring “great” used as a noun in “the great of the story.” Social-media text often contains non-standard grammar, making POS tagging less reliable without domain-adapted taggers.

Polarity Shift – related terms: Sentiment reversal, contrastive conjunctions. Polarity shift occurs when conjunctions like “but,” “however,” or “although” introduce a contrast that changes the overall sentiment direction. Example: “The camera is good, but the battery is terrible” results in a final negative sentiment because the contrastive clause dominates. Detecting shifts requires understanding discourse structure, which simple bag-of-words models miss.

Pre-trained Language Model – related terms: Unsupervised pretraining, transfer learning. Models such as BERT, RoBERTa, and GPT are trained on massive corpora to capture linguistic knowledge before being fine-tuned for sentiment tasks. Example: A pre-trained RoBERTa model quickly adapts to a new brand’s tweet stream, achieving high accuracy with minimal labelled data. Limitations include model size, inference latency, and potential bias inherited from the pretraining data.

Precision – related terms: Positive predictive value, false positive rate. Precision measures the proportion of correctly predicted positive instances among all predicted positives. Example: If a classifier labels 100 tweets as positive and 80 are truly positive, precision is 0.80. High precision indicates few false alarms, which is valuable when downstream actions (e.G., Alerting a crisis team) are costly. However, focusing solely on precision can reduce recall, missing genuine sentiment signals.

Qualitative Sentiment Coding – related terms: Manual annotation, thematic analysis. Human coders read text and assign sentiment labels, often creating a gold standard for training or evaluation. Example: Researchers manually label a sample of Facebook comments to capture nuanced sarcasm that automated systems miss. While labor-intensive, qualitative coding provides high-quality data, essential for domains with specialized jargon or cultural references.

Recall – related terms: Sensitivity, true positive rate. Recall quantifies the proportion of actual positive instances that the model correctly identifies. Example: A system that detects 70 out of 100 negative reviews has a recall of 0.70. High recall ensures most problematic sentiment is captured, crucial for risk monitoring. Balancing recall with precision is a common optimisation challenge.

Sentiment Lexicon – related terms: Opinion dictionary, polarity list. A sentiment lexicon is a curated collection of words with associated sentiment scores (e.G., -1 To +1). Popular lexicons include AFINN, NRC, and VADER. Example: The word “awesome” receives a +0.9 Score, contributing positively when aggregated across a tweet. Lexicons need regular updating to incorporate emerging slang, emojis, and domain-specific terminology.

Sentiment Shift Detection – related terms: Temporal analysis, trend monitoring. This process identifies changes in sentiment over time, useful for tracking brand reputation or public reaction to events. Example: A sudden spike in negative sentiment after a product recall triggers a rapid response from the PR team. Detecting genuine shifts versus noise requires smoothing techniques and anomaly detection algorithms.

Support Vector Machine (SVM) – related terms: Margin maximisation, kernel trick. SVMs are supervised classifiers that separate data points with the widest possible margin, often used for sentiment classification with high-dimensional BoW features. Example: An SVM trained on movie reviews distinguishes positive from negative reviews by finding a hyperplane that best separates the two classes. SVMs perform well on smaller datasets but can be sensitive to feature scaling and kernel choice.

Term Frequency-Inverse Document Frequency (TF-IDF) – related terms: Weighting scheme, feature scaling. TF-IDF reflects how important a word is to a document relative to the corpus, down-weighting common words and up-weighting rare, informative ones. Example: The term “refund” appears frequently in complaint tweets, receiving a high TF-IDF weight that influences the classifier toward negative sentiment. TF-IDF does not account for word order or semantics, limiting its effectiveness on nuanced sentiment.

Topic Modeling – related terms: LDA, latent topics. Topic modeling uncovers hidden thematic structures in a corpus, which can be combined with sentiment analysis to produce aspect-level insights. Example: An LDA model identifies topics “delivery” and “quality” within reviews; sentiment scores are then computed for each topic to reveal that delivery is praised while quality receives criticism. Challenges include selecting the

appropriate number of topics and ensuring topics align with meaningful aspects.

Transfer Learning – related terms: Domain adaptation, fine-tuning. Transfer learning reuses knowledge from a source task (e.g., Language modelling) to improve performance on a target sentiment task. Example: A model pre-trained on general web text is transferred to analyse political tweets, achieving better results than training from scratch. Potential pitfalls include negative transfer when source and target domains differ substantially.

Unsupervised Sentiment Analysis – related terms: Clustering, lexicon-free methods. Unsupervised approaches infer sentiment without labeled data, often by exploiting seed words or distributional semantics. Example: Seed words “good” and “bad” initialise clusters that grow to include synonyms, enabling polarity assignment to new terms. Accuracy is generally lower than supervised methods, and evaluation is more difficult without ground truth.

VADER (Valence Aware Dictionary and sEntiment Reasoner) – related terms: Rule-based lexicon, social-media sentiment. VADER is a sentiment analysis tool specifically tuned for micro-blogging platforms, handling emojis, slang, and punctuation. Example: The tweet “I love this! 😍” Receives a high positive score due to the exclamation mark and heart-eyes emoji. VADER is fast and interpretable but may struggle with complex sarcasm or domain-specific jargon.

Word Embedding – related terms: Vector representation, dense features. Word embeddings map words into continuous vector spaces where semantic similarity corresponds to geometric closeness. Techniques include Word2Vec, GloVe, and FastText. Example: The vectors for “happy” and “joyful” lie near each other, enabling the model to generalise sentiment across synonyms. Static embeddings cannot resolve polysemy; contextual embeddings address this limitation at higher computational cost.

Zero-Shot Sentiment Classification – related terms: Few-shot learning, prompt engineering. Zero-shot methods predict sentiment for a target domain without any labelled examples, often by framing the task as a textual entailment problem. Example: A large language model receives the prompt “Is the following review positive?” And directly outputs a label for a brand it has never seen. While promising for rapid deployment, performance may be inconsistent, especially on niche domains or nuanced language.