



Spam Email Detection — Core Criteria

In **machine learning**, detecting spam emails involves identifying **patterns** that are statistically or semantically **indicative of spam**, based on training data.

Here's a **deep breakdown** of the **key criteria used** (both manually and ML-based):

1. Keyword-Based Indicators (Lexical Features)

Words and phrases **commonly used in spam**:

 Spammy Keywords	 Examples
Free gifts or prizes	"Win a free iPhone", "Claim your reward now"
Urgency	"Act Now", "Limited Time Offer", "Hurry Up"
Financial promises	"Make \$1000 a day", "Get rich quick"
Scams/Phishing attempts	"Update your bank details", "Verify your account"
Suspicious links	URLs with bit.ly, random domains, etc.
Sexual/Pharmaceutical content	"Buy Viagra", "Increase performance", etc.

Used with **TF-IDF** or **CountVectorizer** to extract features.

2. Statistical/Structural Features

These are used heavily in **Naive Bayes**, **Logistic Regression**, and even **XGBoost** models.

Feature	Description
Word frequency	Higher frequency of certain spam-related words
Capital letters ratio	Spam often uses all caps like "GET FREE MONEY"
Special character ratio	Use of !, \$, %, etc. unusually often
Length of email	Spam tends to be very short or very long
Number of links	Excessive links or suspicious shortened URLs
HTML tags	Spam often contains raw HTML or styled links
Email header anomalies	Forged From: field, mismatched Reply-To:

3. Content-Based Features

Includes **semantic and structural NLP-based analysis**.

Feature	Technique/Use
Bag-of-Words or TF-IDF	Captures keyword patterns
Word embeddings (Word2Vec, BERT)	Captures context of language usage (advanced NLP models)
Spammy phrases detection	Phrase match via n-grams or pattern matching
Misspellings and obfuscation	Spammers often use tricks like "Fr33 M0ney"

4. Sender/Source-Based Features

Used by traditional spam filters (e.g., Gmail, Outlook), not always in ML datasets:

Feature	Description
Sender domain reputation	Known spam domains or blacklisted IPs
SPF/DKIM/DMARC validation	Missing these can indicate forged email origins
Number of recipients	Spam often sent to mass mailing lists
"From" mismatch	"From" != sender domain (spoofing)

5. Behavioral & Contextual Features (Advanced)

Feature	Example
Response baiting	Asking user to click, reply, or enter credentials
Previously marked spam	Learning from previous user flags
Email interaction history	If sender is in your contact list or not

How ML Models Learn from These:

In your spam detector using **Naive Bayes + TF-IDF**, the criteria is learned from:

```
X = tfidf.fit_transform(df["text"])
```

```
y = df["label"]
```

```
model = MultinomialNB()
```

```
model.fit(X, y)
```

So your model detects spam by learning:

- Which **terms** are more likely to appear in **spam** vs **ham**
- The **weight** of those terms (TF-IDF gives importance)
- Statistical **likelihood** using Bayes theorem

Summary: Spam Detection Criteria

Category	Examples
Lexical	"free", "win", "prize", "urgent"
Statistical	Length, punctuation, capital letters
Structural	Links, attachments, HTML formatting
Sender Info	Domain, headers, SPF/DKIM failures
Semantic NLP	Context-based patterns (in modern models like BERT)