



1 SEARCH TECHNOLOGIES

1.1 Search Engine approaches

Search engines exploit a range of different search techniques:

- The simplest search engines employ text string matching.
- Stemming is used to enable related words with similar root meanings (e.g. “run” and “runs”) to be found. Occasionally stemming leads to unwanted hits – for example, a search for (the metal) “lead” might return hits for (sales) “leads”.
- Statistical techniques rely on analysis of the terms in documents along with the frequency and contextual relationship of other terms whose occurrence is correlated with that of the terms. Searches return documents containing terms with a high correlation and similar contextual relationships to those in the search query. Statistical techniques can lead to erroneous correlations and hence inaccurate results; they also typically fail to recognise associations between synonymous terms that rarely occur in the same document.
- Synonyms can be used to find words that have similar meanings but dissimilar roots. A major benefit of synonyms is that they reduce the need for the searcher to guess the precise words used in the documents for which they are searching. However, synonyms can also lead to unwanted hits, especially when a term is expanded inappropriately – for example, “lawn” being substituted for “grass” when “inform” was the intended meaning.
- Semantic approaches provide the benefits of synonym-based techniques, but reduce unwanted hits by using semantic analysis to disambiguate words whose meaning depends on the (semantic) context in which they occur. Thus alternative interpretations of the word “lead” in a document would be ranked differently depending on other words occurring in the same document – for example, “pipe” or “sales” or “team”.
- Information consumers should not have to know or guess the exact words used by a producer. Statistical approaches are easily misled into inferring erroneous associations, and text matching with synonyms (but no semantic analysis) tends to make inappropriate substitutions. Semantic approaches, based on concepts rather than text strings, enable users quickly and easily to locate documents relevant to their needs.
- Fully semantic approaches handle features within documents (and search terms) not as words, but as “concepts”. This has the major advantage of enabling cross-language searching – searching in one language for documents authored in a different language. A search query for “llyn” entered in Welsh would find documents authored in English containing the word “lake”.

Cross-language searching should not be confused with *multi*-language searching offered by most search engines, including Autonomy. These latter approaches allow documents in multiple languages to be indexed, but searching is based on language-agnostic text-string matching. An Italian searching for “burro” (butter) could be returned hits for a Spanish document containing “burro” (donkey), but not for “butter”.

1.2 Additional User Features

- Fuzzy or phonetic matching is useful when the same word can be spelled in many ways – for example, as a result of transliteration from a foreign alphabet, or difficult or unusual spellings. Information consumers should not have to be concerned with the vagaries of the transliteration process, so there needs to be a means of matching alternative spellings.
- For information sources that maintain metadata associated with each item of (textual) content, it is highly desirable for the search engine to index the metadata. Some search engines combine metadata with free text when indexing, whilst others index metadata separately – in effect, maintaining a structured database. The latter approach enables users to express queries in terms of metadata values (or ranges thereof, where appropriate), or full text, or a combination of the two.
- Skilled users need to be able to construct complex queries from simpler query terms, using features such as wildcards, exact phrases, date ranges, boolean connectives (e.g. AND, OR, NOT, WITHIN, ADJ, BETWEEN), nesting of query clauses, and to have a range of options for the manner in which results are collated and presented to users.
- Many frequently occurring entities such as numbers, telephone numbers and dates, can have multiple representations. End users should not have to be concerned with which representation(s) occur in the documents being searched. The indexing and search components of the search engine should therefore, where practical, normalise the representation of such entities so that accurate matching can be effected independently of the representations, both in the user query and in the document indexes.
- Most searches will return multiple documents, which the user needs to be able to collate in various ways – for example, by date, by type, or by relevance. Ranking documents by relevance (a measure of how closely each document matches the user's search criteria) enables the user to focus on those documents most likely to be of interest.
- When a large document is returned from a search, the user needs to be able quickly to locate the section(s) of the document containing relevant material. A number of techniques can assist the user – for example, hit term highlighting, hyperlink embedding (enabling the user to jump quickly between occurrences of hit terms in a document) and summarisation.
- Facilities for manipulating the results returned by a search include: searching for “more like” a particular return, recurrent or refined searches, and dynamic classification.
- Dynamic classification enables users to analyse and classify results with purpose-defined classifications to create role- or task-oriented perspectives of the search results.
- Enterprise-scale search engines typically provide federated and distributed deployment options, enabling systems to be configured for :
 - Scalability and performance – for example, by distributing queries across multiple query servers, and partitioning indexes across multiple servers;
 - Resilience – for example, by providing automatic fail-over between multiple query servers.

2 CATEGORISATION TECHNOLOGY

2.1 Categorisation approaches

Categorisation is the process of analysing a document and assigning it to one or more of a set of categories, on the basis of its content and/or metadata. In effect, categorisation performs a search for the document against the criteria that define each category, and determines which of those categories result in the closest match with the document. This is significant, in that the search approach adopted by a product usually has a direct impact on the characteristics of its categorisation features.

Categories can, in principle, be used to classify almost any aspect of a document. However, the real challenge for categorisation, and that for which it is most commonly used, is to determine the subject or topic of a document, based on its content and, optionally, its metadata. Although it is possible for categories to be entirely unrelated, typically they are organised into a hierarchic taxonomy, where the sub-categories of a category represent a more specific subject than the (parent) category.

The structure of a corporate taxonomy needs to be carefully architected:

- a) from the perspective of making it intuitive for users - e.g. for navigation,
- b) because it colours the way in which organisation thinks about itself;
- c) it has to be technically feasible.

Two related aspects of any categorisation technology need to be considered:

- Setting up and maintaining a taxonomy and the associated the category definitions;
- Analysis of a document and assigning it to one or more taxonomy categories.

How these are accomplished depends on the approach to categorisation taken by the technology. There are three principal, very different approaches:

- Automatic generation of a taxonomy from a sample document set.

This is the approach adopted by Autonomy and many statistical-based search engines (e.g. Verity "Concept Extraction"). Essentially, the sample document set is statistically analysed by a technique known as cluster analysis that groups documents with similar content into "clusters". The taxonomy is generated by creating a category wherever a distinct cluster can be clearly identified.

It is an approach in which technical feasibility and ease of implementation are the prime considerations. Autonomy, for example, claims 100% accurate categorisation with this approach, although it should be recognised that the process outlined above for identifying categories is designed to ensure that the categories are clearly distinguishable – whether or not those distinctions have any relevance at all to business needs.

This approach works best where the material to be categorised comprises documents of similar structure, style and language – for example, news articles or directories (e.g. Jane's Fighting x), and where there is no business requirement for a corporately defined taxonomy.

It must also be noted that over time, the generated taxonomy will change, as the nature and the clustering of the concepts within the document corpus change. This can be a good thing in that the taxonomy itself is self-maintaining. However it is also a challenge in two particular respects: control over the nature of the taxonomy is further diminished (this could particularly be significant in considering the make-up of a user interface); and the mapping between the generated taxonomy and any other (e.g. retrieval) taxonomy could potentially remain in permanent flux. This is particularly relevant where the taxonomy underpins a subscription service, since the basis of those subscriptions is continually changing.

- Automatic categorisation to a pre-defined taxonomy, training the categorisation system with sample documents for each category;

This is the approach adopted by, for example, Verity (“Automatic Classification”), InXight, MoHoMine and GammaSite. Its main drawback is the need to identify a representative set of training documents (typically 30-50 per ‘concept’ or category, however this is also dependent on the nature of the content to be categorised). For small taxonomies, this is not too onerous. However, with larger taxonomies, not only are there more categories, but also a larger number of training documents is required for each category, in order to maintain the precision with which the categorisation system discriminates between categories with finer distinguishing characteristics. Thus the size of the training set increases more than linearly in the number of categories. The task of identifying a representative training set has been a major bottleneck in some past projects, requiring extensive effort by domain experts to identify and manually categorise documents in the training set.

It is possible to introduce workflow to enable the user community to help train the auto-categoriser (for example flag ‘typical’ documents for a category, enabling their subsequent inclusion in the training set). However, this only works well up to a point. For best performance, skilled taxonomists are needed to understand how best to optimise categorisation (for example, multiple passes against multiple different training sets, intelligently designed, are far more accurate than one single training set against the whole taxonomy). Currently, this is still very much a black art. Given the variable nature of content being categorised, and differences between target taxonomies, it is not in general possible to ‘buy’ accurate training sets.

This approach works best where the material to be categorised comprises documents of similar structure, style and language – for example, news articles or directories (e.g. Jane’s Fighting x), and where there is a business requirement for a corporately defined taxonomy, provided it has no more than a few hundred categories.

- Rule-based (deterministic) - manual specification of category definitions;

This is the approach adopted by Convera RetrievalWare and Verity “Business Rules” (formerly known as “Verity Topics”). The success of the approach depends on providing domain experts with tools that assist them in the manual specification of features that characterise each category.

The approach of specifying rules that define each category may, at first sight, seem far more demanding of domain expertise and taxonomy specialists than that of using a set of sample documents to train a categorisation system. However, the required expertise is similar to that needed to identify and categorise documents for a training set; and, as noted above, the number of documents required in a training set may be very large.

A major consideration, when evaluating these technologies, is the terms in which category definitions are specified. At the lowest level, category definitions are specified in terms of particular words or phrases; at the other extreme, they are specified in terms of “concepts” that are independent of the words or phrases in which they are expressed.

The latter approach, used by RetrievalWare, uses an automatic, lower level categorisation step, in which the document is analysed to identify *domain-independent concepts*. This analysis is simpler and more deterministic than the higher level categorisation of the subject of a whole document in terms of *domain-specific categories*. It can be carried out in a manner that is independent of the higher level taxonomy, since the analysis is primarily dependent on the language (which may include specialist terms for a particular knowledge domain). Categorisation at the higher level can be performed very efficiently, purely in terms of the concepts identified by the lower level categorisation. Indeed, with this approach, categories can be defined and populated dynamically, enabling users to perform dynamic, taxonomy-based analyses of search results.

Technologies that adopt a rule-based approach typically provide a range of pre-defined taxonomies, with associated category rule definitions, for specific application areas. Furthermore, many taxonomies, such as geography, are sufficiently generic to be widely applicable. These can often be used as a starting point for defining the category definitions for related business-specific taxonomies, or can be used as supplied. Substantial savings can be achieved by using pre-generated, professionally maintained rule-sets, rather than starting from scratch. Similarly, RetrievalWare provides pre-defined vocabularies, and language and taxonomy “cartridges” that support the lower level categorisation process.

The latter two approaches support pre-defined taxonomies. In both cases, careful consideration should be given to how the taxonomy evolves. On the one hand, the topics that the taxonomy needs to cover, and their relative priorities (and hence, usually, the depth of the taxonomy under each category), will inevitably change over time. On the other hand, where a taxonomy is used to underpin a publish-and-subscribe service, existing subscriptions can be adversely affected when the structure of the taxonomy, or the definitions of categories within it, change. Extension of the taxonomy by adding new sub-categories will have minimal impact on subscriptions that specify a category *and all its sub-categories*. At the other extreme, replacement of several categories by a new set of categories is likely to have a major impact on subscriptions.

3 INFORMATION EXTRACTION TECHNOLOGY

Information Extraction attempts to bridge the disparity between structured (e.g. metadata) and unstructured (e.g. free text) data, by creating metadata from information found within the free text of a document. The metadata thus created can be indexed, or used as a basis for categorisation, or for the automatic generation of navigation structures or dynamic link insertion.

3.1 Named Entity Recognition

A corpus of documents could be analysed to identify telephone numbers, postcodes, names of people or place names, or terms from a specialist vocabulary (e.g. specific chemicals, military components and hardware etc.) contained in the text body of each document. The items to be identified are termed “entities”.

Logically, the metadata model is extended to include (repeated) attributes for each type of entity being identified. For example, for each document there might be a metadata attribute containing a list of the place names discovered within the document.

The metadata created by entity extraction can be used for several purposes:

- A category can be associated with each value of an entity type – for example, for each place name.
- If the categories associated with entity values can be organised into a structure – e.g. a hierarchy of place names, by locality and size – a hierarchic navigation structure can be created automatically.
- When a document that has been analysed in this way is presented to a user, occurrences of recognised entities can be automatically hyperlinked to enable rapid retrieval of other documents containing the same entity value.

Recognition of entities based on controlled vocabularies is relatively straightforward, although it may be necessary to allow for alternative spellings, etc.. In other cases, recognition may be based on syntactic or lexical form – e.g. dates or telephone numbers. In both of these examples, there are alternative representation conventions and potential ambiguities that can only be resolved with the aid of contextual data (e.g. is 02/03/2005 a US or UK date?).

Identifying names of e.g. people can be challenging, depending on the source and quality of the document. Different occurrences of names of people (especially with misspellings, diminutives, nick-names, etc.) cannot, in general, be reliably matched. Specialist tools (e.g. from SSA) can be used to assess the *likelihood* that two names match, but this uncertainty (and the fact that many people share the same name) entails very different handling of the results.

Note that these uncertainties and potential ambiguities exist, to a degree, in almost every information source – for example, same place names in different countries, or incomplete names (e.g. first name only) – but contextual information can often be used to reduce the resulting errors to an acceptable level.

3.2 Entity Relationship Recognition

The approach commonly used for entity relationship recognition is to parse each sentence in the document, identifying named entities and semantic constructs. The semantic analysis is the most challenging, and as with other natural language processing techniques, is most effective in limited domains, and with documents sharing a common style.

Pragmatically, one of the most effective techniques for identifying semantic constructs is to infer pattern-matching rules from a manually annotated training set. The rules are applied statistically (i.e. pattern-matching yields a *probability* that a particular construct has been recognised) and the final step is to determine a compatible set of matches that have highest joint probability.

The rule set can be refined and extended with some (limited) manual intervention. Examples of new constructs can be annotated to provide further training. Some systems also allow “negative examples” to be included in the rule inference process, to eliminate or reduce false matches. It should be observed that such rules can identify not only instances of previously encountered relationship types, but also new types of relationship.

Increasingly, ontologies are being used to formalise the types of entities and the types of relationships that may exist between them. These can be used not only to drive (or constrain) the analysis process but also to maintain what is, in effect, an evolving schema of entity types and relationships. Thus, as with categorisation, there is a variety of approaches that range from pre-defined ontologies to the system inferring the ontology for itself, on the basis of minimal training.

Current technology for recognising relationships between entities in free text is relatively immature. Typical measures for precision and recall in a non-specialised corpus are around 60-70%.

Companies with strong credentials in Information Extraction include Semagix, InXight, and CognIT.