# EinCite: The Key to Document Similarity

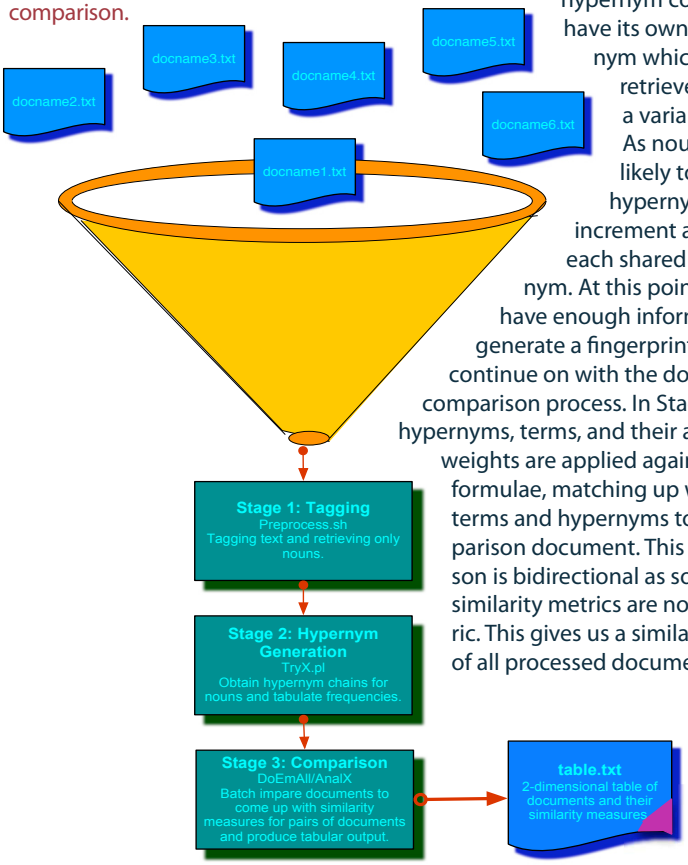
## Document Fingerprinting & Document Comparison

What is a document fingerprint? How are they built? What is a hypernym? How do we compare documents?

Hypernyms are the key to building a document fingerprint, which is a collection of graphs depicting the number and strength of relationships between nouns in the document. A hypernym of a noun is simply a more general, abstracted idea of the noun, generally denoted as being the "kind of" relationship. Apple is a kind of edible fruit, so edible fruit is a hypernym of apple. Most search engines are strongly keyword-based, but what if a document doesn't mention fruit at all, but you still want to look for articles about fruit or aren't interested in whether a specific word occurs in a document? The theory is that the use of hypernyms, which abstract the concepts in a document, can be used to build a more general understanding of a document. Will each "fingerprint" be unique as human fingerprints are? Probably not, but that is an open question.

The overall process of generating a document fingerprint is partially depicted in Figure 1 outlining how to compare documents. A document is first digested by a set of natural language parsing tools to tag and retrieve only the nouns by context as hypernym relations are not available for other parts of speech. A hypernym is retrieved, if available, from the WordNet lexicographic database for each noun in the list. Each hypernym could also have its own hypernym which is also retrieved back to a variable depth. As nouns are likely to share hypernyms, we increment a count for each shared hypernym. At this point we have enough information to generate a fingerprint, but we continue on with the document comparison process. In Stage 3, the hypernyms, terms, and their associated weights are applied against some formulae, matching up weighted terms and hypernyms to a comparison document. This comparison is bidirectional as some of the similarity metrics are not symmetric. This gives us a similarity matrix of all processed documents.

**Figure 1:** Overall comparison process
Parsing, hypernym generation, and algorithm application for comparison.




## EinCite (Phase 1): The 30-Second Summary

What if a document had a fingerprint? What if you could tell how similar two fingerprints were? What if you could group fingerprints by their similarities? Welcome to EinCite.

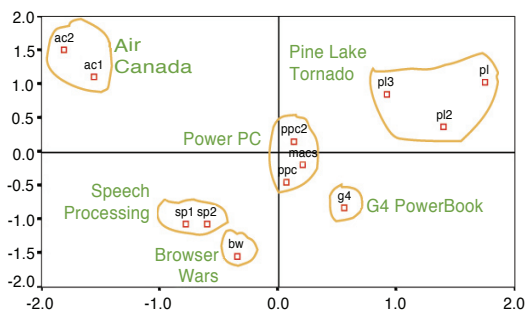
EinCite is a multi-phase document classification, information retrieval, and intelligent agent system for retrieving interesting hypertext documents on the basis of a fingerprint calculated by using semantic relations from WordNet to determine a document's subject. In the document classification phase (Phase 1), each concrete noun in a document can be said to belong to a more general class of nouns. These more general classes are aggregated and provide meta-information that can be used to build the document fingerprint and to classify the document.

## The Experiments & Results So Far

Prototypes and people. Clustering is grouping similar documents together well, at least with some algorithms.

A set of two experiments have been run using sets of twelve documents, eight people and two automated metrics. The people and the prototypes follow the same sequence of steps: read through each document in pairs and decide how similar each pair of documents is on a scale. The result is a two-dimensional similarity matrix which can then be analysed using multidimensional scaling (MDS) (Borg, 1997) to produce a clustering of how similar the documents are (See Figure 2). The primary prototype algorithm encapsulates the notion that frequently occurring hypernyms are more representative of the overall document subject and should be more important. The second algorithm uses a simple pairwise comparison with cosine normalization, common in information retrieval tasks. MDS allows us to aggregate the rankings by people to produce an overall clustering, but it does not allow us to easily compare two sets of rankings to see how close they are because placement in the vector space is dependent on the data used to make the space. Adjusting the prototype algorithms is therefore difficult as it is not possible to quantify the level of improvement. Nevertheless, some initial output from the prototypes (e.g., Figure 2) looks very promising.

**Figure 2:** Sample Similarity Clustering
The clusters are circled, with nearest green text indicating document subject. This was using one of the in-house similarity metrics.




IDEAs Lab, Informatics, University of Sussex
See full text at http://einiverse.eingang.org/publications/2004hct-eincite.pdf
**Michelle A. Hoyle (eingang@sussex.ac.uk)**