# A Holographic Associative Memory Recommender System

Matthew F. Rutledge-Taylor
Institute of Cognitive Science
Carleton University
Ottawa, Ontario, K1S 5B6
mrtaylo2@connect.carleton.ca

André Vellino
CISTI Research
National Research Council
Ottawa, Ontario K1A 0R6
andre.vellino@nrc.ca

Robert L. West
Institute of Cognitive Science
Carleton University
Ottawa, Ontario, K1S 5B6
robert_west@carleton.ca

## Abstract

*We describe a recommender system based on Dynamically Structured Holographic Memory (DSHM), a cognitive model of associative memory that uses holographic reduced representations as the basis for its encoding of object associations. We compare this recommender to a conventional user-based collaborative filtering algorithm on three datasets: MovieLens, and two bibliographic datasets such as those typically found in a digital library. Off-line experiments show that the holographic recommender is competitive in accuracy for predicting movie preferences and more accurate than collaborative filtering on very sparse data sets. However, DSHM requires significant amounts of computational resources which may may require a distributed implementation for it to be practical as a recommender for large data sets.*

## 1 Introduction

The purpose of a recommender system is to automate the task of providing recommendations for items (such as songs, books, movies or merchandise) that a user is likely to be interested in, given the collective preferences of a user community. Recommendation systems can be used not only to provide personalized web experiences on e-commerce web sites [12] but also to enhance the information retrieval experience in a digital library portal [6].

Most conventional recommender systems operate by clustering similar items according to some characteristic of the item (content-based recommendation) [11], by measuring the similarity among ratings that users have given to items (collaborative filtering – either memory-based or model-based) [3] or by combining the two in some manner (hybrid recommenders) [5].

This paper describes a somewhat different approach to recommendation, one that is based on a cognitive model of associative memory – Dynamically Structured Holographic Memory (DSHM). We were motivated by the intuition that applying a cognitive model of memory could enhance the effectiveness of an information management system such as a recommender by making it behave more like a human expert [8]. For example, often the best way to get a movie recommendation is to ask the video store clerk to recommend a movie based on what you have enjoyed viewing recently. Similarly, your best bet for finding relevant journal articles is to ask an expert in the field what to read next given a set of articles that you have found useful. Our objective was not so much to discover a recommendation technique that was more effective or efficient for typical recommender tasks in commercial applications as to verify the intuition that a cognitive model of memory was a viable alternative to purely statistical or probabilistic approaches.

The off-line experiments described in section 2 were intended only to characterise the accuracy of our DSHM recommender on both the MovieLens data – often used to benchmark CF recommenders – and on two very sparse bibliographic datasets taken from a digital library. The performance, serendipity, scalability or usability of such a system will be analysed in future work.

### 1.1 Collaborative Filtering

Recommender systems typically operate on three kinds of entities: users, items and the preference ratings that users have assigned to items. Given a set of ratings for certain items – whether they are obtained from users explicitly or implicitly from, for example, browsing patterns – a user-based collaborative filtering (CF) system will attempt to predict the rating of a previously unrated item for the active user based on how other (similar) users previously rated the same item.

Recommending journal articles in a digital library is more problematic than recommending other kinds of items because the usage data is sparse relative to the number of items in the collection [7]. One remedy for this problem is to use bibliographic citations as a proxy for user ratings

[16]. This was the technique we used for the experiments described in this paper.

In the experiments with CF discussed below, we used a user-based CF recommender that implements k-nearest neighbour and cosine correlation in the Taste framework (now part of the Apache-Lucene machine learning library Mahout [1]). Note that, in this instance, where user preferences are equivalent to article citations, a user-based approach is equivalent to an item-based one.

## 1.2 DSHM - Dynamically Structured Holographic Memory

DSHM is a cognitive model of human long-term memory [14, 15]. It is designed as a tool for understanding how the human mind organises knowledge, e.g., how it stores, confuses, forgets and accurately retieves information. Our implementation of DSHM is a self-contained Python program that does not depend on any other modeling software.

DSHM makes use of Holographic Reduced Representations (HRRs) to encode associations between concepts [13]. The system is composed of holographic items, (which we will refer to here as H-items, to avoid confusion with the term "item" as used to refer to rated entities, e.g., movies), which represent things to be remembered by the system. Each H-item consists, primarily, of two large vectors of floating point numbers, each with a Euclidean length of 1.0. The environmental vector is random and static (i.e., it does not change). It is used as the system's internal representation of the mental entity corresponding to the H-item. In contrast, the memory vector is dynamic (i.e., it changes over time). The memory vector of one H-item is used to store all the associations between that H-item and other H-items in the system. For the experiments described in this paper, the number of floating point numbers in each vector was set to 2048. The larger the number of elements, the better the memory capacity of the items. The trade-off is that larger vectors take greater amounts of computational resources to manipulate mathematically.

Associations between H-items are formed when a set of H-items is given to the system as input. From a cognitive perspective this can be interpreted as the H-items co-occurring in a thought, a verbal utterance or a perception. The system distinguishes between sets for which the order of the elements is essential to the content of the set as a whole (e.g., the words in a sentence) from those that are not (e.g., the things scattered about my work desk). If the set of H-items is unordered, every H-item is associated with every subset of the other H-items in the set, up to a predefined maximum number of elements. In the experiments presented in this paper, this maximum was set to the lowest permissible value (namely one) ; i.e., each element of a set is only associated with every other element of the set, but not with any combinations of pairs and other n-tuples of elements. The effect of increasing this maximum is to improve the context sensitivity of the system at the cost of additional computational resources. A typical DSHM model of memory[15], applied to smaller data-sets, would use a value of two or three.

These associations between elements are recorded by binding the environmental vectors of the H-items in each subset together, and adding the resulting vector to the memory vectors of the other H-items in the set. A binding is formed by recursively computing the circular convolution of the environmental vector of an H-item from the given subset and the binding of the remainder of the subset. The circular convolution of vectors is commutative, and thus the order in which the H-items of an unordered set are bound together does not affect the resulting aggregate binding. However, if the set is an ordered list, the neighbours of every H-item up to a system-defined maximum distance are associated with the given H-item in a manner that preserves the order of the H-items [14]. The result of these methods of associating H-items together is that the memory vector of each H-item encodes information about all of the other H-items with which it has co-occurred.

### 1.2.1 The structure of DSHM

A collection of H-items in a DSHM system defines a multi-dimensional state-space where each environmental and memory vector is a point on the surface of a hypersphere with radius 1.0. This state-space implements a complex semantic network occupied by the items represented in the system. One property of this organisation of H-items is that given an incomplete pattern, the provided, known, H-items from the pattern can be used to predict the most likely candidates for completing the pattern. This is done first by generating a set of "probe" vectors based on the memory vectors of the known H-items in the pattern. Each probe is computed by reversing the binding process described above and predicts an environmental vector that approximates the vector of the item that best completes the pattern [14]. We refer to this method of prediction as "decoding".

Another interesting property of the organisation of H-items in the system is the relationships between H-items' memory vectors. When a large set of patterns has been entered into the DSHM system, and the associations between H-items have been computed, the memory vectors of the H-items in the system will cluster. These clusters are usually open to a meaningful interpretation relating to the content represented by the H-items in the sets of patterns originally presented to the system. For example, as is the case with BEAGLE – a system upon which DSHM is based, and which is equivalent to a special case of DSHM where the patterns are sentences and the items are words

– the H-items will cluster according to semantic similarity, or synonymy [10]. The reason for this is that items, which are in some way equivalent and can be interchanged for one another, will tend to have the same sets of neighbours, and will therefore develop similar memory vectors. Thus, given an H-item, its memory vector can be used as a probe to be compared to other H-items' memory vectors. The matches found will be the H-items that are similar to the one providing the probe. We refer to this method of prediction as "clustering". It is important to emphasize that the similarity between two H-items discussed here is not based on any content about the items provided to the DSHM system. Rather, the system is inducing the similarity of H-items based only on the patterns in which the items occur. This ability is, of course, part of what make DSHM an interesting cognitive model of memory.

#### 1.2.2 Recommendation in DSHM

Given a set of items (e.g., books, movies, or journal articles), and a set of users who are defined by what subset of the items they have rated, the purpose of a CF system is to accurately predict what rating a user is likely to assign an item that he or she has not yet rated. Given a test item and a test user, CF assigns to the item a value based on two factors. The first (in user-based CF) is the similarity of the test user to a neighbourhood of other users who have also rated the item and the second is the ratings assigned to the item by the other users. There are various functions that can be used to compute this value. What such a function provides is a metric for how consistent the test item is with the items rated by the test user.

In contrast with CF methods which use algorithmic statistics to generate recommendations, the most natural way to recommend items in DSHM, in a manner equivalent to user-based CF, is to treat a user as simply a set of unordered ratings. The ratings are considered to be unordered because we assume that the value given to an item is highly, although not completely, independent of any considerations that may impose an order on the ratings, such as the dates the ratings were provided. We define a rating as the combination of an item and a preference value. When imported into the DSHM system, the ratings are converted into H-items, which we will refer to as H-ratings, and are associated with one another by the binding process described above. Thus, for each H-rating, information about other H-ratings with which it has co-occurred is stored in the memory vector of the H-rating. Once all of the users' ratings have been imported into the system, the state-space defined by the H-ratings' memory vectors will cluster as described above.

It should be noted that, strictly speaking, in DSHM, only the decoding method corresponds to traditional user-based

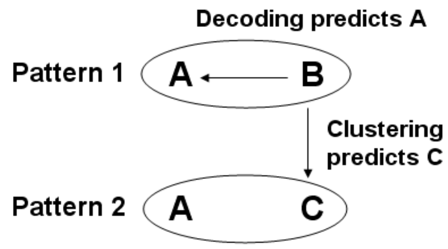


**Figure 1. Decoding vs. Clustering in DSHM**

CF. A simple example will illustrate the difference between the decoding and clustering methods. If an arbitrary object A co-occurs with B in one instance, and with C in another instance, the decoding method, when applied to B, will predict A. This is because only A co-occurs with B. The clustering method, when applied to B will predict C because only C has the same neighbours (i.e., A) as B. We have included reference to the clustering method in this paper because it makes use of the exact same data as does user-based CF and produces noteworthy results.

## 2 Experiments

Which methodology to choose for evaluating a recommender's performance depends a great deal on the kind of user task for which the recommender is applied, the datasets being used to perform the evaluation as well as what characteristics (e.g., accuracy, usefulness, serendipity) of the recommender are being evaluated [9]. For this study, our objective was to compare DSHM with a conventional CF recommender to understand both whether DSHM is an applicable technique and how it compares in accuracy with CF on sparse bibliographic datasets. We first establish a baseline for DSHM on the MovieLens dataset and then extend the comparison between these two approaches to two different datasets obtained from a repository of biomedical journal articles.

### 2.1 MovieLens

Our first experiment was to establish a baseline of predictive accuracy comparisons between DSHM and CF on the MovieLens dataset. This test was done using the usual 10%-90% cross-validation methodology. Previous studies of CF on this data show a Mean Absolute Error (MAE) of approximately 0.73, depending on parameters for the neighbourhood size [4].

The greatest challenge to predicting ratings using DSHM, is the fact that the current implementation has no innate ability to represent magnitude. Given that the goal is to predict a numerical rating, some means of accommodating magnitudes needed to be incorporated into the representation of ratings in the system. We decided that for every movie in the MovieLens dataset, five distinct H-items would be created, one for each possible rating, e.g., "Toy Story (1995)" with a rating value of 4 would be a single atomic entity in the system, as would "Toy Story (1995)" with a rating value of 5. The cognitive interpretation of this is that the mental representations that correspond to liking a given movie very much, and liking it only somewhat, differ in many dimensions, and not just on a single numerical scale. Hence, representing each rating for each movie as entirely different H-items presumes nothing about how the ratings for the same movie ought to be related to one another. Thus, prior to the learning phase, no associations between any H-ratings exist in the DSHM recommender. Given this method of representing ratings, a rating prediction then becomes the task of determining which of the five H-ratings for a given movie is most highly associated with the H-ratings corresponding to a test user's other ratings.

Of the 6040 MovieLens users, approximately 10% (614) were randomly removed from the sample and used as test users. The DSHM recommender was trained on the ratings provided by the remaining 5426 users. For each of the test users, the DSHM recommender made predictions for ten of the test user's ratings. These predictions were done one at a time, and used all of the user's other ratings as sources from which to base the predictions.

#### 2.1.1 Results

As mentioned above, there are two distinct, but related, ways in which DSHM can predict a rating. In the case of the decoding method, DSHM is being asked to find ratings that are likely to have co-occurred with the known ratings. Here, DSHM produced a MAE of 1.23. This poor performance was due to our choice of how to represent ratings. The drawback of not presuming any relationship between the pair of ratings corresponding to a given movie with values of 4 and 5, which CF can leverage, is obvious. By creating five H-ratings for each movie, there are too many items relative to the number of users for DSHM to discover reliable co-occurrence patterns of preferences for the test users. However, via the clustering method, the DSHM recommender produced a competitive MAE of 0.71. In this latter case, DSHM is being asked to find ratings that are similar to the known ratings. This task is more resilient to noise because the value of an H-rating's memory vector is the accumulated influence of many associations, which, on average, represents a reliable location in the state-space occupied by relevantly similar ratings. Note that this method of measuring H-rating similarity would allow for clusters to represent movie preference profiles. For example, nothing would prohibit a romance movie rated 1 from clustering with an action movie rated 5. Additionally, given the high-dimentionality of the vectors representing these items, there is the potential for very nuanced associations between items to develop in the system.

### 2.2 Journal Articles

The experiments we performed are modeled after off-line experiments undertaken on TechLens+, which measured recommender effectiveness by computing Top-N recommendations [16]. We selected set of test articles whose references are known and, for each article in that set – the active article – we systematically removed one reference at a time and tested whether the recommender predicts the removed reference. If the removed reference ranks highest in the list of recommendations, it belongs to the Top-1 recommendations, if it ranks in the first five recommendations it belongs to the Top-5, etc. As with TechLens+, we treated articles as "users" and articles' lists of references as lists of boolean "ratings" for other articles (although we note that while bibliographic references in an article are an indicator of relevance they are not necessarily an indication of *favourable* relevance in the mind of the author).

Our experiment compares DSHM and CF on Top-N results on two bibliographic datasets: one was extracted from a collection of 31,000 articles from 39 Medicine journals and the other from a collection of 114,000 articles from 107 Biology journals. The Medicine collection was reduced to 7495 articles by eliminating articles for which references were unavailable. In addition, the references we used to populate the preferences matrix were only the references that were made to articles in that collection. Overall, the total number of references in the collection was over 273,000, but only 4100 of them were references to articles in the collection, for an average of only 0.55 references per article. In other words, the collection was both very sparse and very loosely connected. The Biology collection was reduced to 38,667 and also had a small average number of references (1.15 per article) to articles in the collection. The connectivity of the article collections — measured as the number of references in an article plus the number of articles that cite it — was slightly above 1 for the Medicine collection and slightly above 2.3 for the Biology collection, as compared to 14 in the CiteSeer collection used in Techlens+ [16].

Our experimental method also differs from the TechLens+ study in some respects. One is that the cross-validation was not 10-fold and not random. Instead, we chose to perform leave-one-out evaluations exhaustively on a sample of the articles biased towards those with the most references to items in the bibliographic collections. One reason for using this strategy rather than the random selection strategy was that the likelihood of picking a random article with only one or fewer references to articles in the collection was quite high. Leaving one reference out for each of the articles with the most references seemed more likely to produce a recommendation that was correct. Thus we chose a subset of 95 test articles in each collection which had between 17 and 5 references per article in the Medicine collection, for a total of 570 prediction attempts and between 32 and 13 references in the Biology collection, for a total of 1491 prediction attempts.

For this experiment, the DSHM implementation was essentially the same as for the MovieLens experiment, except that there needed to be only one H-item representation for each unique article. For each of the non-test articles, the H-items representing the article's references were associated together. To make a prediction, the memory vectors of the H-items corresponding to the remaining references were used to generate probes used to rank all of the other articles in the collection according to how highly they were associated with the probes. Again, the two distinct methods of generating these probes, as described above, were used. The DSHM recommender was asked either to recommend articles that were most likely to have co-occurred with the provided references, or to find references that were similar to the provided references.

### 2.2.1 Results

The results of these experiments are summarized in Figure 2.2.1. In the case of the journal article recommendations, the DSHM recommender produced very good results via the decode method, presented here. Unlike for the MovieLens data, the clustering method did not produce better results for journal article recommendation. In the case of the Medicine collection, DSHM correctly predicted the Top-1 references almost 3 times as often as CF (76 versus 28), and converged to a 36% improvement over CF for the Top-30. For the Biology collection the accuracy of DSHM is a little less dramatic but exhibits the same trends; e.g., 127 Top-1 predictions for DSHM versus 51 for CF, and a 30% improvement for Top-30.

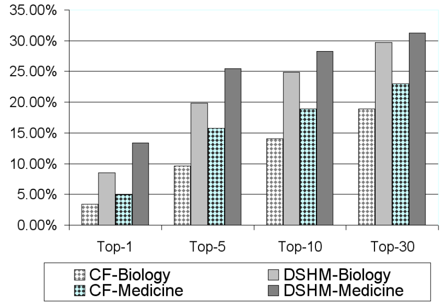


**Figure 2. Top-N results for both CF and DSHM on two article collections.**

## 3 Conclusion

Predicting bibliographic citations from a holographically reduced representation of bibliographic information shows that recommending Top-N items offers better accuracy than CF on very sparse datasets. We interpret this superiority of DSHM as resulting from its ability to self-organise based on the information extracted from the data.

In addition to being more accurate for sparse datasets, the flexibility of holographic recommenders offers promising possibilities for recommending items that have ratings in multiple dimensions as well as item correlations that are content-based. DSHM provides a unified mechanism for implementing what would otherwise be considered a "hybrid" recommender.

The benefits of using DSHM as a recommender for large datasets are outweighed by the considerable computational cost of producing them. This is due to the cost in space and time of performing thousands of matrix computations to produce each recommendation. Hence, variants on collaborative filtering techniques are still more practical for digital library recommender systems under most foreseeable circumstances. Nevertheless, in cases where significant pre-compiling of information is feasible, e.g., for relatively static or very small datasets, or when few unique queries are made to the system, DSHM may be useful for maximally digesting the available data in advance.

## 4 Future Work

Our future research on DSHM as a recommender system will focus on examining exactly how information is exploited differently in DSHM compared to CF systems. This will include cluster analysis of the vector state-space of DHSM recommenders, as well as a detailed examination of how learning in DSHM differs from model building CF. We would also like to compare the accuracy of a DSHM system to which content information (e.g., movie genres or article abstracts) has been added, against the accuracy of typical hybrids of CF and content-based filtering. In addition, we intent to compare the serendipity characteristics of DSHM recommendations. We believe DSHM may distinguish itself my mimicking the serendipity of recommendations that human experts provide and differ significantly from the serendipity of collaborative filtering. Finally we intend to build an open-source, Java implementation of DSHM and evaluate it's performance on a Hadoop platform [2].

## References


[1] http://lucene.apache.org/mahout.

[2] http://hadoop.apache.org/.

[3] G. Adomavicius and A. Tuzhilin. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *Knowledge and Data Engineering, IEEE Transactions on*, 17(6):734–749, 2005.

[4] B. Sarwar, G. Karypis, J. Konstan and J. Reidl. Item-Based Collaborative Filtering Recommendation Algorithms. In *WWW '01: Proceedings of the 10th International Conference on World Wide Web*, pages 285–295, New York, NY, USA, 2001. ACM Press.

[5] R. Burke. Hybrid Recommender Systems: Survey and Experiments. *User Modeling and User-Adapted Interaction*, 12(4):331–370, 2002.

[6] E. Frias-Martinez, G. Magoulas, S. Chen, and R. Macredie. Automated User Modeling for Personalized Digital Libraries. *International Journal of Information Management*, 26(3):234–248, June 2006.

[7] Z. Huang, H. Chen, and D. Zeng. Applying Associative Retrieval Techniques to Alleviate the Sparsity Problem in Collaborative Filtering. *ACM Trans. Inf. Syst.*, 22(1):116–142, January 2004.

[8] M. Huggett, H. Hoos, and R. Rensink. Cognitive principles for information management: The principles of mnemonic associative knowledge (p-mak). *Minds Mach.*, 17(4):445–485, 2007.

[9] J.L. Herlocker, J.A. Konstan, L.G. Terveen and J.T. Riedl. Evaluating Collaborative Filtering Recommender Systems. *ACM Trans. Inf. Syst.*, 22(1):5–53, 2004.

[10] M. Jones and D. Mewhort. Representing word meaning and order information in a composite holographic lexicon. *Psychological Review*, 114(1):1–37, 2007.

[11] M. Pazzani and D. Billsus. Content-based recommendation systems. pages 325–341. 2007.

[12] D. Pierrakos, G. Paliouras, C. Papatheodorou, and C. D. Spyropoulos. Web Usage Mining as a Tool for Personalization: A Survey. *User Modeling and User-Adapted Interaction*, 13(4):311–372, November 2003.

[13] T. Plate. Holographic reduced representations. *IEEE Transactions on Neural Networks*, 6(3):623–641, May 1995.

[14] M. Rutledge-Taylor and R. West. MALTA: Enhancing ACT-R with a holographic persistent knowledge store. *Proceedings of the XXIX Annual Conference of the Cognitive Science Society, Nashville, TN.*, pages 1433–1438, 2007.

[15] M. Rutledge-Taylor and R. West. Modeling the fan effect using dynamically structured holographic memory. *Proceedings of the XXX Annual Conference of the Cognitive Science Society, Washington, DC.*, 2008.

[16] R. Torres, S. McNee, M. Abel, J. Konstan, and J. Riedl. Enhancing Digital Libraries with TechLens+. *Proceedings of the 2004 Joint ACM/IEEE Conference on Digital Libraries*, pages 228–236, 2004.