Evolutionary document classification for content-based recommender systems

John Pagonis
Pragmaticomm Limited,
November 3rd, 2010
Birkbeck College, University of London

Today's menu

- Recommender systems in brief
- Genetic algorithms in text classification
- Engene
- Basic Information retrieval used with GAs
- The truth about document classification with GAs
- The Engene classifier in content based recommendation

RecSys - what are they?

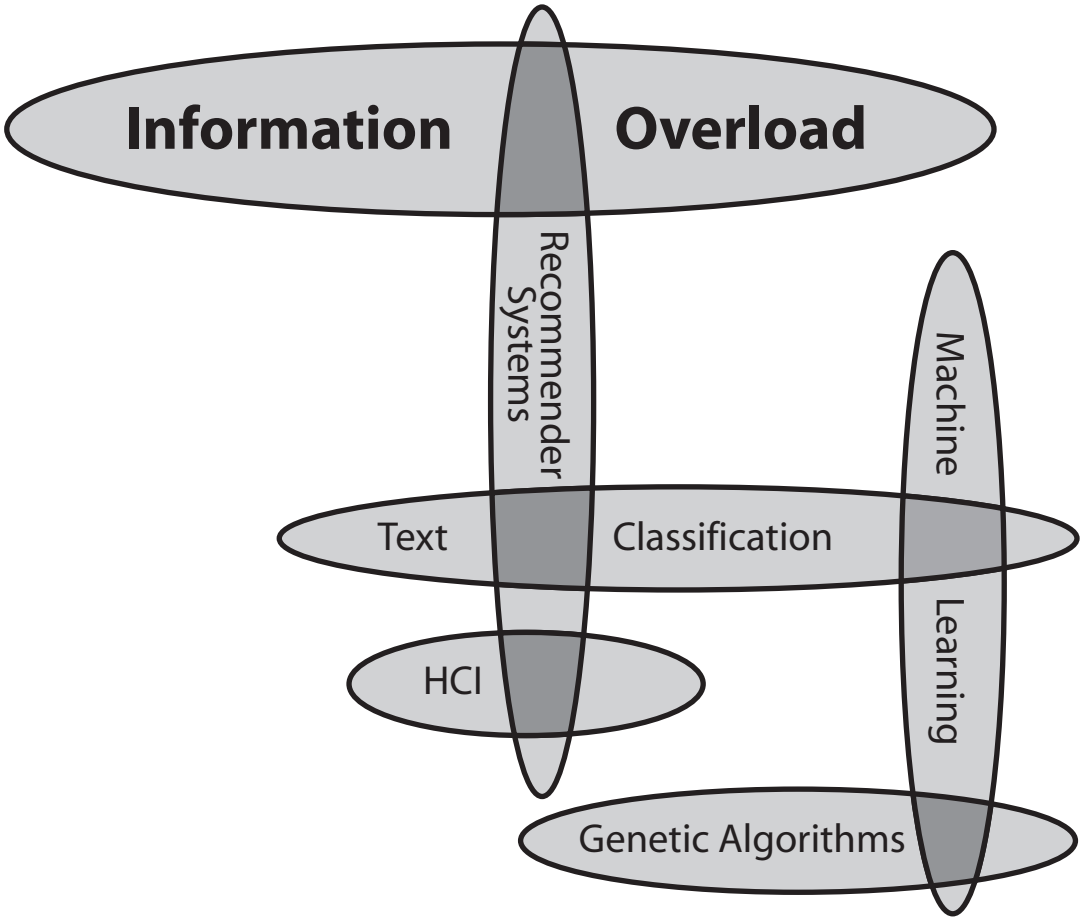
a.k.a Recommender Systems, Recommendation Systems, Recommenders

RecSys - almost everyone has a definition!

- a.k.a Recommender Systems, Recommendation Systems, Recommenders
- Recommenders normally support users in their information seeking, consumption and filtering activities.
- Typically, they are *information filtering tools* that given some prior knowledge of a user's preference and given a corpus of application specific information they attempt to make *personalised* recommendations. Such recommendations aim to *assist users in making a decision*, bring an item to their attention or convert information into knowledge. For that matter, they are also viewed as decision support tools
- Think of recommendation as classification!

About the GA document recommender domain

... it is a multi-discipline domain



RecSys dimension I - individual vs collaborative

Recommendations are computed using as context the profile of one or many actors. Recommenders can be individual or collaborative (or both).

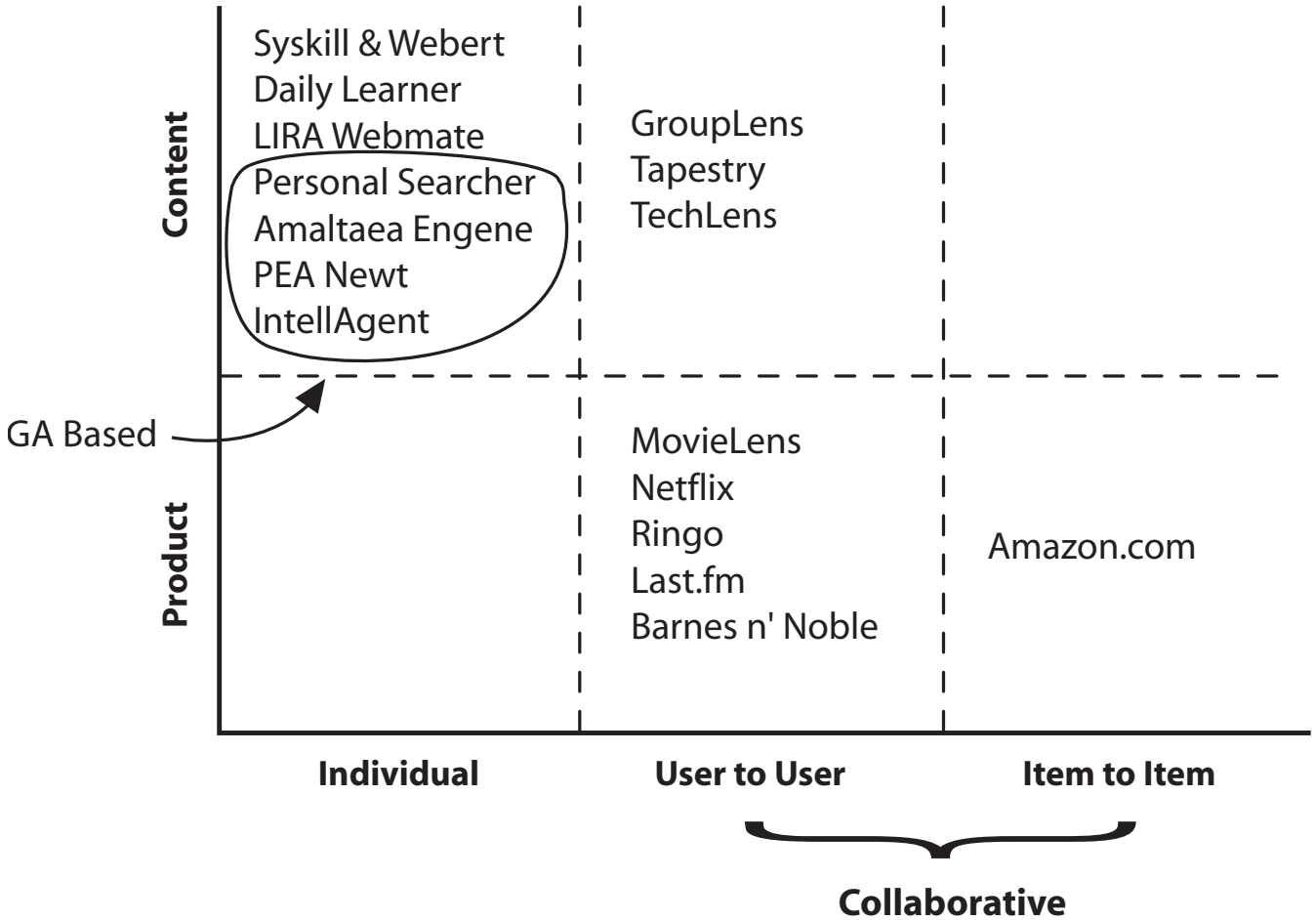
- Individual
  - … based on some prior knowledge of the user's profile
- Collaborative
  - … based on a user's profile similarity to those of other users
  - … based on an item's relationship to other items

RecSys dimension II - product vs content

Recommendations may deal with products or content. The distinction lies with whether the object's dimensions (in the feature space) are used directly for the computation or whether some of its distinctive attributes (that describe the object) are used instead.

- Product based
  - … e-commerce related, books, DVDs, movies, goods in general
- Content based
  - … documents (text), music (sound),

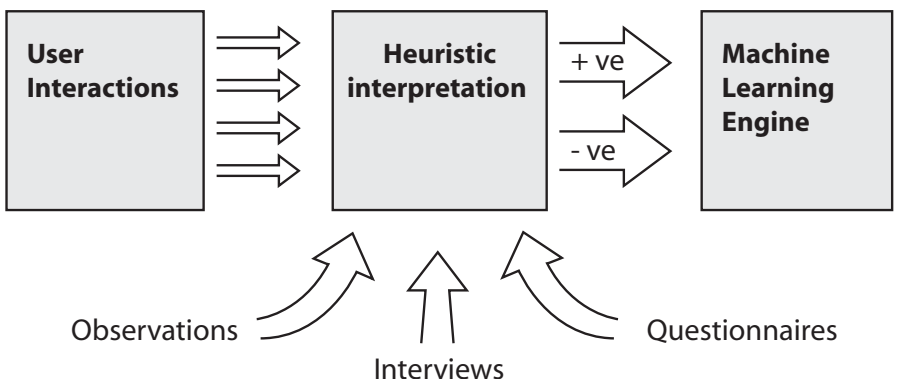
A RecSys perceptual map example



RecSys dimension III - user feedback

Recommender systems need user feedback in order to adapt their inference of a user's or item's profile. Such feedback is either implicit or explicit.

- Explicit
  - … the user gives direct feedback such as rating from a scale
- Implicit
  - … the system interprets signals from user interactions



RecSys and Machine Learning

Recommenders profile users and items.

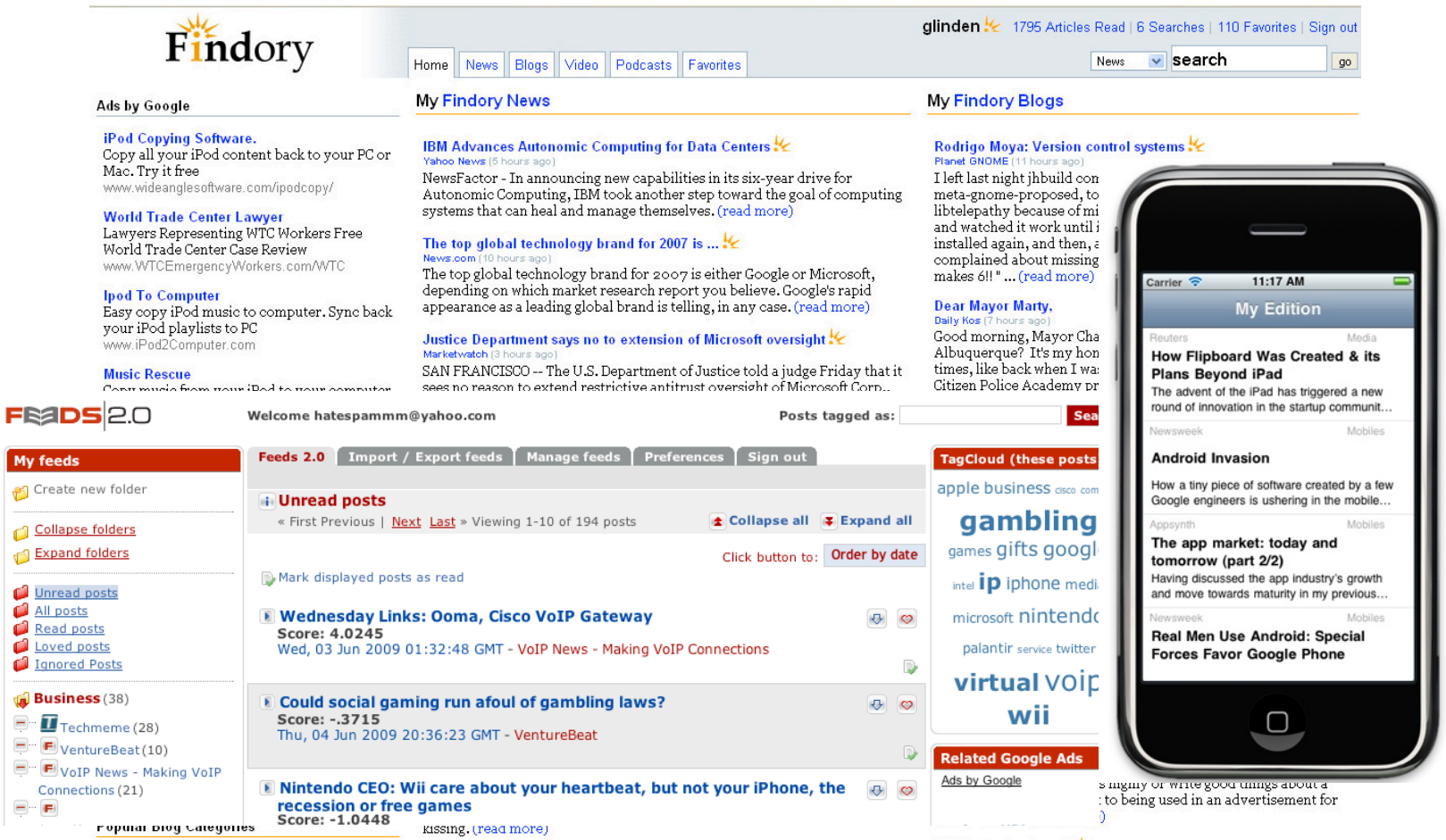
In most cases recommenders maintain their inference of a user's or item's profile by using machine learning techniques

(hence such work is oftentimes referred to as personalisation or user profiling).

Machine learning is more important of course for individual recommenders than it is for collaborative recommenders because the latter can always refer to the opinion of the majority in order to make suggestions, if necessary.

Content-based Recommender Systems

Perhaps the simplest manifestation of a content-based recommender is the 'personal digest' (aka personal newspaper, Daily Me, etc).



Evolutionary content-based RecSys

We created a personal content recommendation system.

It uses a novel genetic algorithm classifier as the primary machine learning technique. We call this Engene :-)

It filters and recommends news and articles of pertinence to its user.

The system was created from scratch and has a Web as well as a mobile interface.

Fact!

- Genetic Algorithms in the past decade have been under-utilised in the fields of recommender systems and text categorisation.

Why?

Problem with using GAs in text classification

- Typically, GA document classification measures performance by asking the user to give feedback after each or few generations.

Problem with using GAs in text classification

- Typically, GA document classification measures performance by asking the user to give feedback after each or few generations.
- So the user becomes part of the fitness function
- Therefore the supervision loop is opened
  - => This is not practical for “outside the lab” deployment

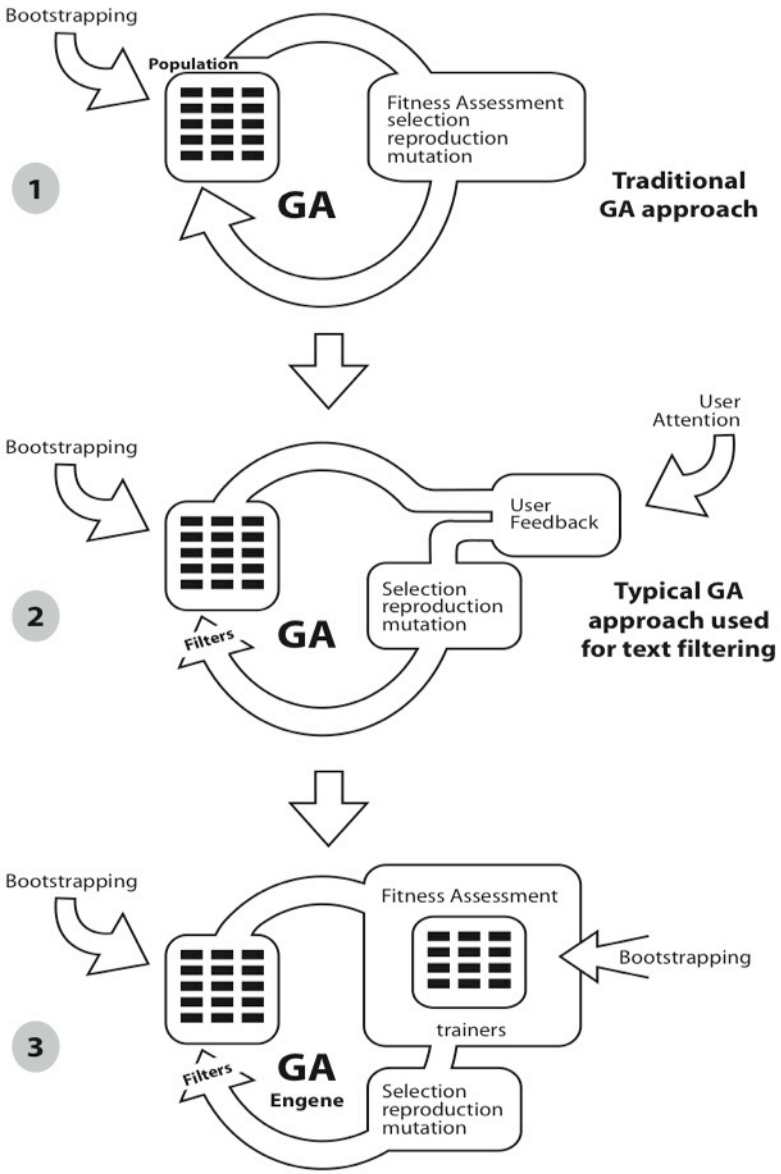
Closing the supervision loop...

What if
instead of the user
we could use
a proxy to the user?

Engene: Closing the evolution feedback loop

- **Engene** creates a proxy to the user (and therefore closes the supervision loop) by using a collection of documents that represent the user's feedback.
- In this arrangement 2 multi-dimensional vector sets are used to represent each user interest (class).
- This ensemble includes :
  - a document collection that never evolves (in the EC sense), referred to as the trainer set
  - a population of information filters (the trainees) which is evolved under the direction of the trainer set. These filters are genetic algorithm individuals.

GAs in text classification open the feedback loop and Engene closes it



18

User is removed from the fitness function

But with removing the user from the fitness function and the need for constant attention, proper bootstrapping becomes vital and thus demands much more work (if not automated)

- ... though bootstrapping is a GA-based text classifier's dirty secret anyway...

- For each subject of interest the user has to supply 20 to 30 documents that represent that interest

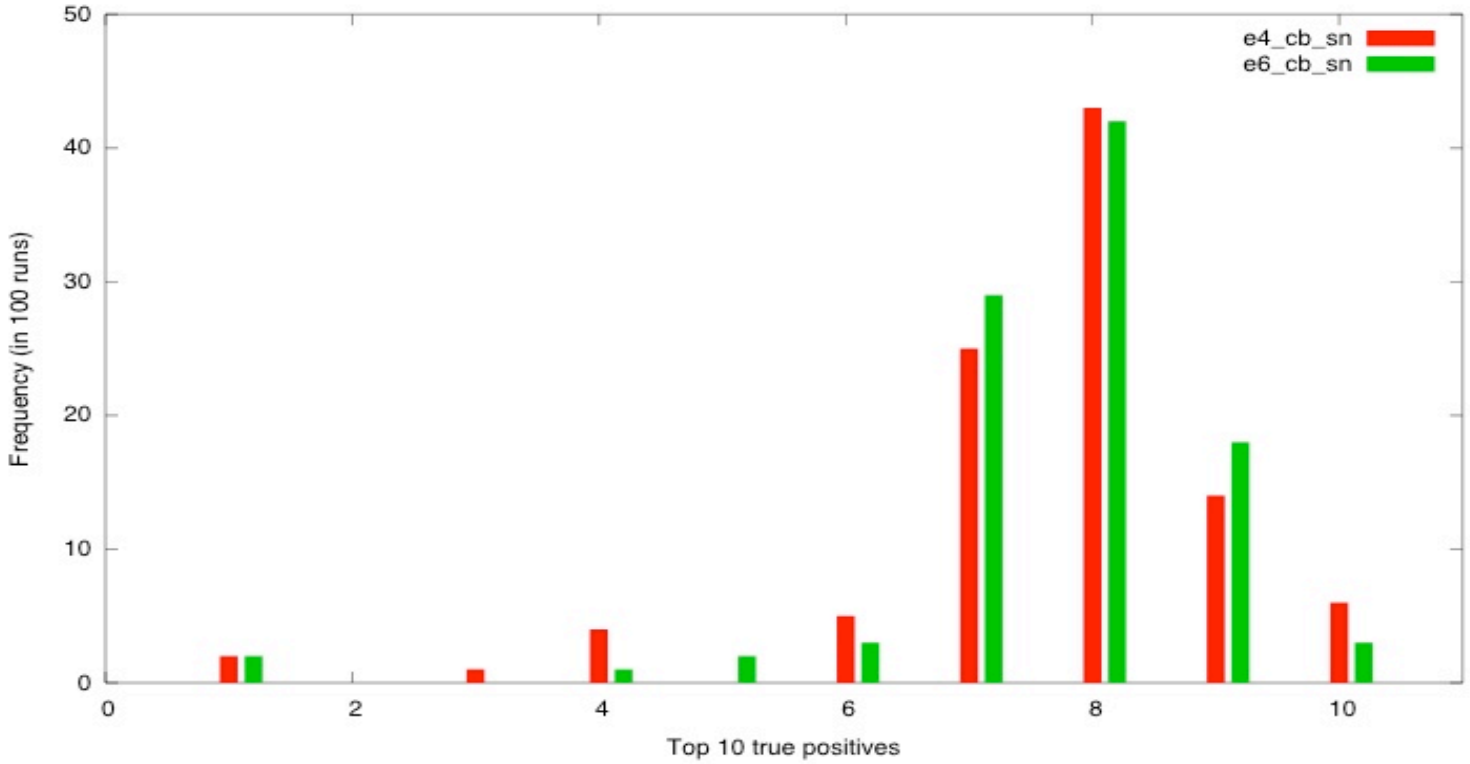
- When used in batch mode, this is all that is needed to build the classifier. After 100 generations a small elite of the evolved filters is called to classify the test documents.

Engene

- Is a GA-based one-class soft textual document classifier that may be used either incrementally or in batch mode.
- Is primarily used incrementally as part of a personal content-based recommender system.
- Operates unattended without needing constant user feedback.
- Engene is applied as a filter to incoming documents and ranks them according to their pertinence to a given category.
- Ranking is achieved using the vector space model and cosine similarity function which are popular in the field of information retrieval.

Engene as a one-class batch text classifier

- Achieves a mean of 7.65 with std.dev. 1.39 in top-10 ranking tests, giving a precision of 76.5% with 10.62% recall.



- There is a 92% chance that Engene will produce between 7 and 10 true positives in the top-10 list.

Engene's performance as a batch classifier compared...

- This performance level is similar to what Naive Bayes classifiers usually claim. Not as impressive as other techniques such as SVMs though.
- In fact, for the same corpus used to test Engene, a variant of the One-class k-NN classifier performs better, with 90% precision at 12.5% recall, albeit being 5+ times slower.
- So, there is still room for improvement...
- **... but**

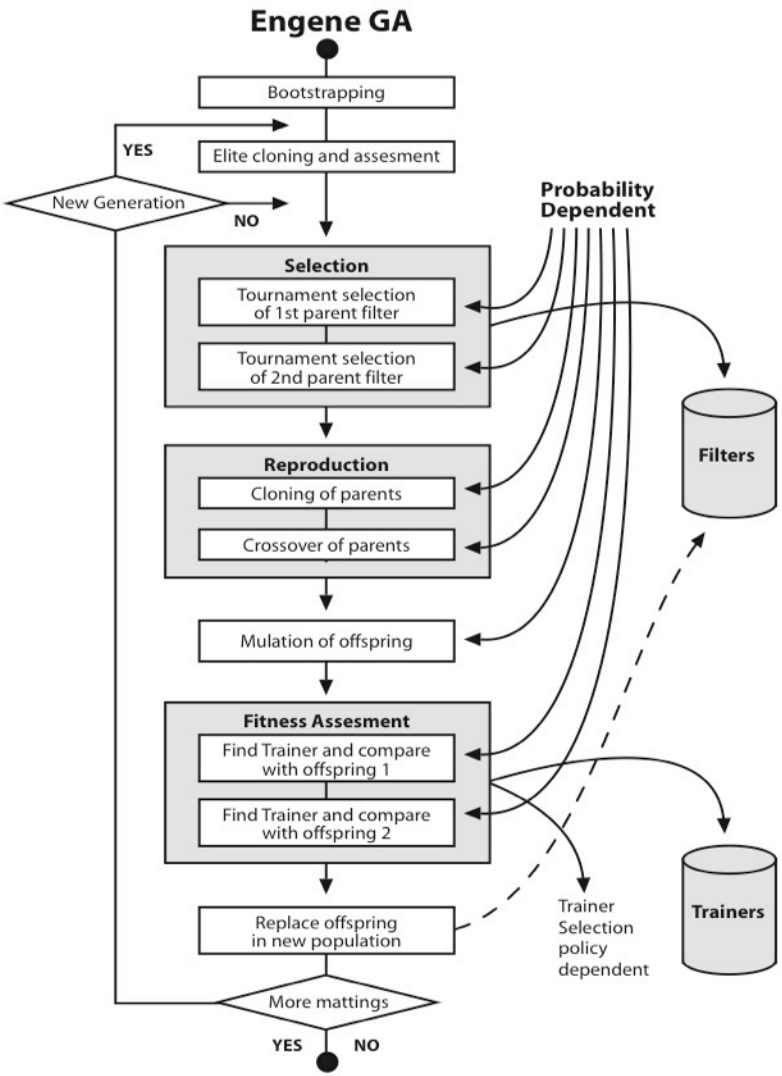
Engene's performance as a batch classifier compared...

- ... but we also found that Engene's False Positives are in most cases Near True Positives!
- This is a desired property for recommender systems!
- The whole point is to filter AND explore the search space, because predictable recommendations have little value!

Engene's GA configuration

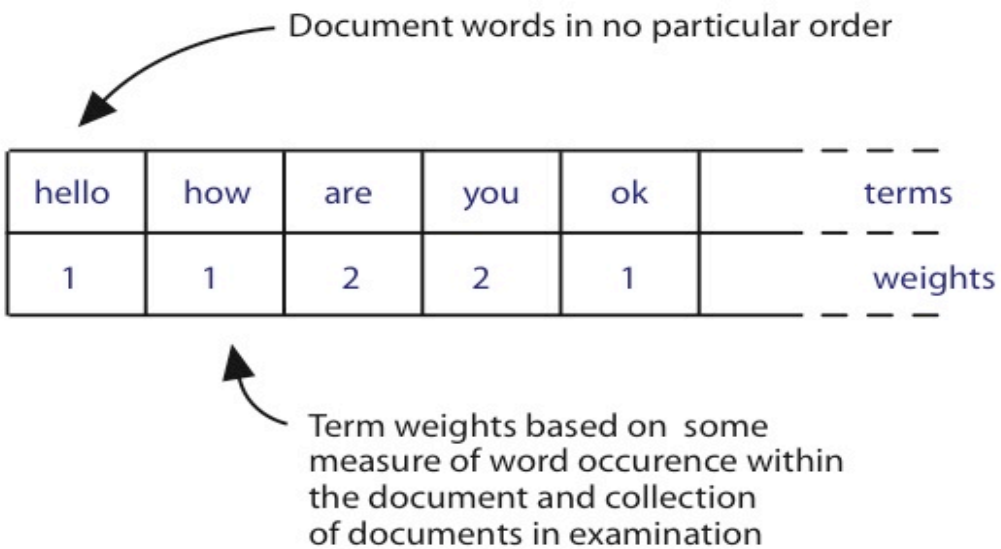
- Dual population of vectors (per profile) that encode documents using the vector space model (i.e it uses multi-dimensional weighted term vectors)
- Fitness function is the cosine similarity measure that assesses every filter's fitness by its similarity to a trainer
- Generational by design
- Variable length chromosomes
- (out of 4) tournament selection
- A random two-point crossover (at 40%)
- Elitism with re-assessment
- Mutation operator randomly changes a gene's allele by 20% (at 3%)
- Simple cloning (at 60%)

Engene GA


Engene GA
Bootstrapping
Elite cloning and assesment
YES
New Generation
NO
Probability Dependent
Selection
Tournament selection of 1st parent filter
Tournament selection of 2nd parent filter
Filters
Reproduction
Cloning of parents
Crossover of parents
Mulation of offspring
Fitness Assesment
Find Trainer and compare with offspring 1
Find Trainer and compare with offspring 2
Trainers
Trainer Selection policy dependent
Replace offspring in new population
More mattings
YES
NO

Encoding documents and individuals

- A typical way of representing documents in the vector space model is that of encoding them in weighted term vectors. In which case every document term represents a dimension of the vector while its weight denotes its scalar size.



- This fits naturally to the GA chromosome metaphor.

Term weighting

- The most popular weighting scheme for a term is that of using its Term Frequency within a document multiplied by the Inverse Document Frequency of the term in the corpus of all documents examined.
- So that:

$$W_{term} = TF*IDF$$

TF*IDF

TF denotes how popular the word is in this document

IDF denotes how unique (therefore discriminating) is a word in a collection of documents

most basic form of TF*IDF is:

$$W_{term} = TF * \log_{10} (N/DF)$$

where:

- TF is the occurrence count of the term in the document

- N is the number of documents examined
  DF is the number of times that the term occurs in the document collection
  ... so TF*IDF is perfect for Information Retrieval

TF*IDF

TF denotes how popular the word is in this document

IDF denotes how unique (therefore discriminating) is a word in a collection of documents

most basic form of TF*IDF is:

$$W_{term} = TF * \log_{10} (N/DF)$$

where:

- TF is the occurrence count of the term in the document

- N is the number of documents examined
  DF is the number of times that the term occurs in the document collection

... so TF*IDF is perfect for Information Retrieval

But...

TF*IDF vs Engene

For a small number of documents that are supplied by a user in order to define an interest (class), frequently the important terms are repeated in all documents.

thus $DF \approx N$ hence $\log_{10}(N/DF)$ --> zero

TF*IDF vs Engene

For a small number of documents that are supplied by a user in order to define an interest (class), frequently the important terms are repeated in all documents.

thus $DF \tilde{} N$ hence $\log_{10}(N/DF)$ --> zero

Hence we discovered that using no IDF or $\log_{10}(N^2/TF)$ instead yields excellent results. We also normalise the TF by the term length of the vector to cater for variable length individuals.

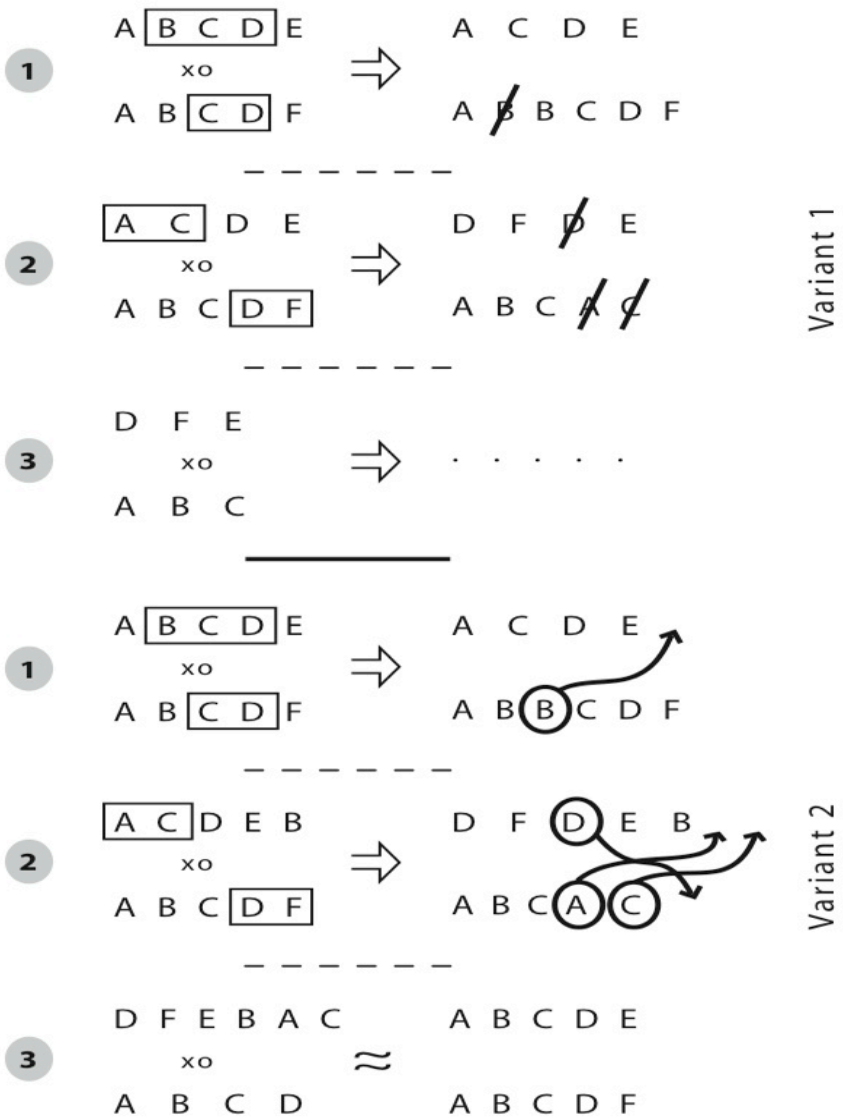
Therefore we concluded that the effects of the classic IDF are damaging during evolution, though useful during final ranking of incoming documents.

Engene's random 2-point crossover

- we tested 2 variants
- The first prunes common terms after recombination by throwing away common terms
- The second moves redundant terms back to the individual where they came from

In all cases the former performed better

Engene's random 2-point crossover



Engene's random 2-point crossover

Q: So what magic does this simple Engene GA do to create better filters?

Engene's random 2-point crossover

Q: So what magic does this simple Engene GA do to create better filters?

A: Feature Selection + Dimensionality Reduction

...but without losing any of the gene pool!

Bottom line

Engene (like some other GA based classifiers) produces shorter more focused filters, thus doesn't suffer from over-fitting.

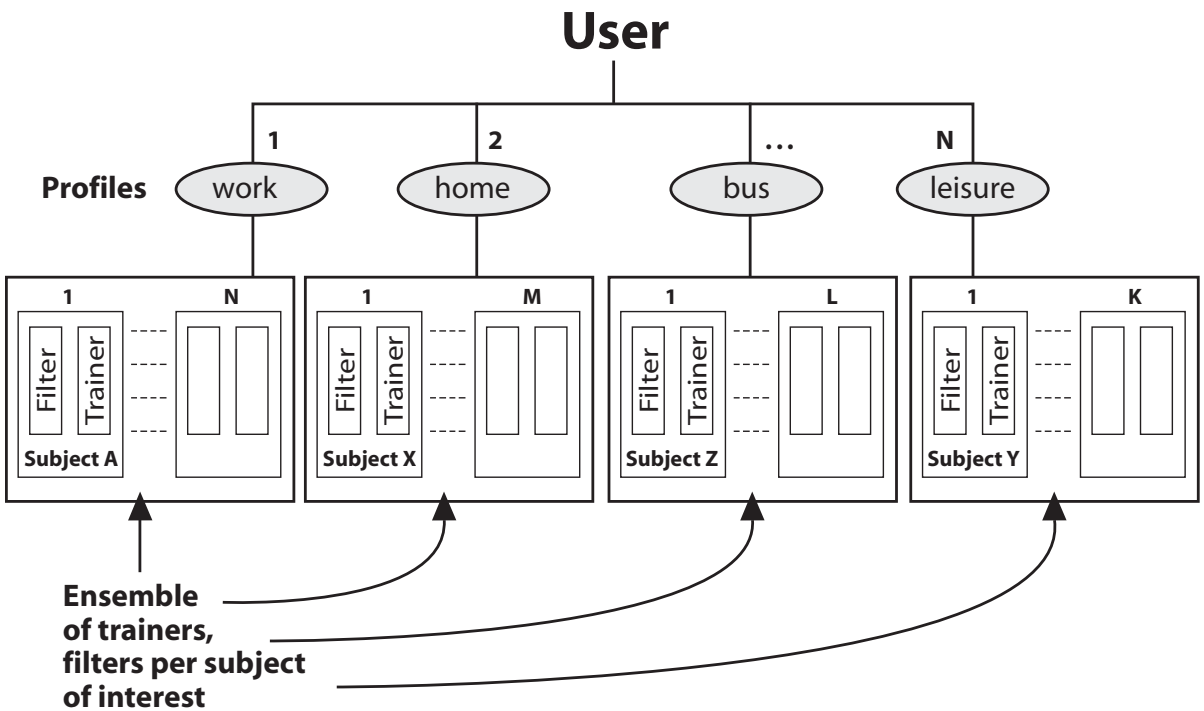
Using the classic TF*IDF weighting harms this process during evolution.

These filters (vectors) tend to include the most important terms and therefore can be used with IR techniques and in combination with other classifiers to retrieve filtered sets of documents.

At the same time when feature selection is performed, information is not thrown away because it remains in the gene pool. This is desired when used in incremental mode and fore recovering from concept drifts.

Engene doesn’t demand constant attention from the user.

Using Engene in a content-based RecSys



Each user may have many user profiles.

Each profile is made of many subjects of interest.

A subject of interest is represented by an ensemble of vectors (denoting a class).

To prepare a digest for a given profile all profile classes are used to rank inbound content.

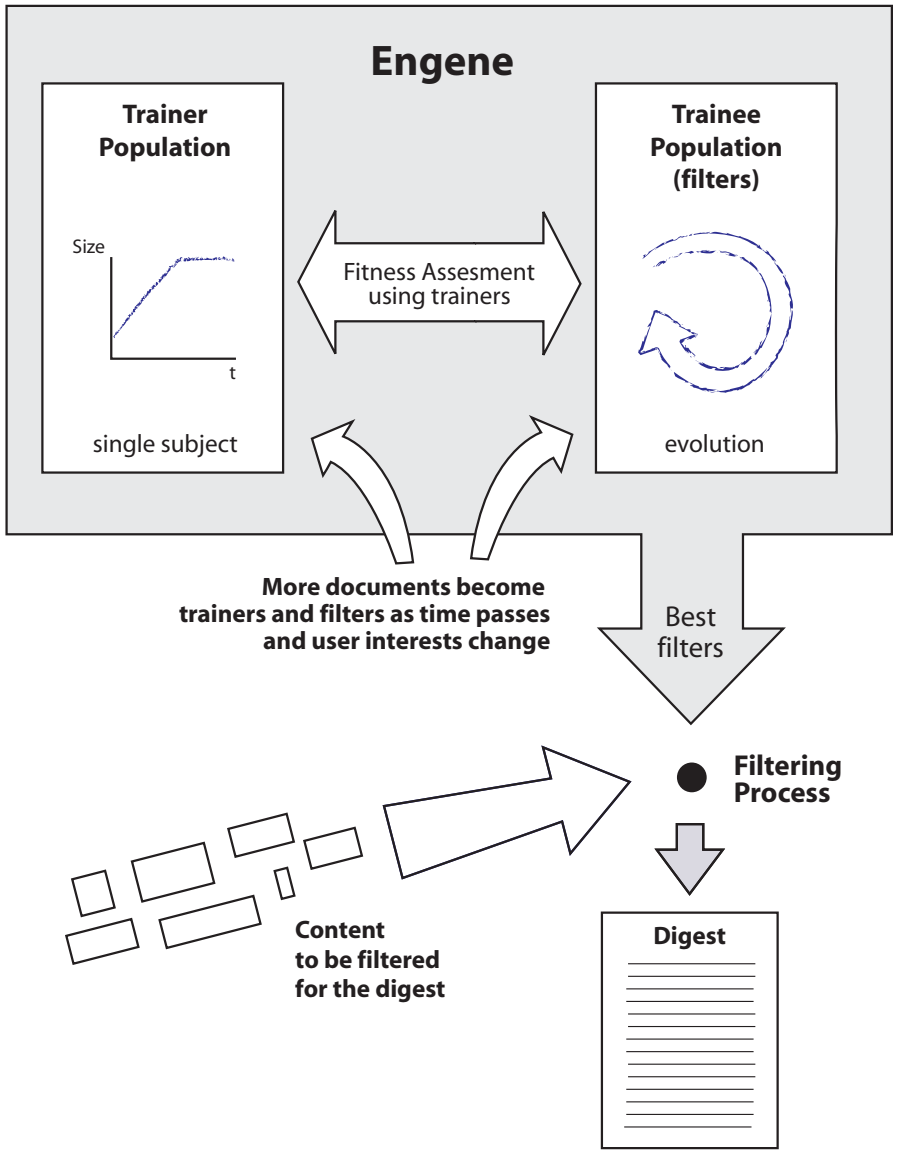
Using Engene in a content-based RecSys

The RecSys is presented with documents that need to be ranked.

For a given profile a digest is built. The system uses Engene to rank each document in the digest.

This ranking is achieved using the elite of the filter population of each category (subject of interest) that the user is interested in.

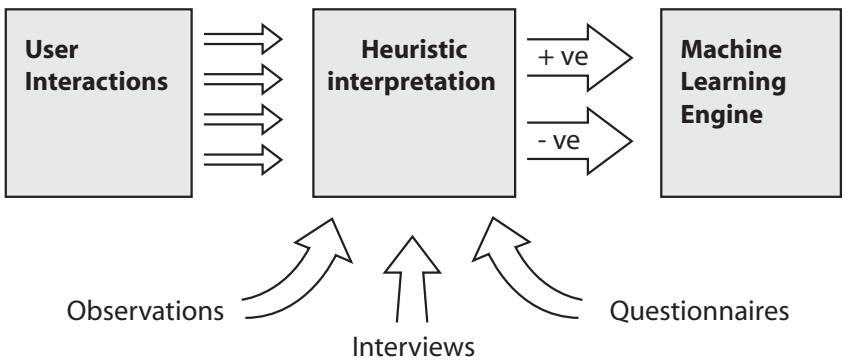
The filter population is evolved by using the trainer population vectors as input to the fitness function.



Use of implicit feedback

Every time the user interacts with the digest, feedback is implicitly interpreted into seven discrete signals.

- positive
- strong positive
- very strong positive
- neutral
- negative
- strong negative
- very strong negative



These signals drive the machine learning mechanism

Adaptation to interests and concept drifts

As opposed to techniques that only reinforce the weights of terms in vectors as part of positive feedback, with Engene...

☑ Positive signals lead to the addition of document vectors in the corresponding ensemble of a subject of interest.

☑ This updates the genetic material of the filters.

☑ It makes up for mediocre bootstrapping.

☑ The profile of the user is updated with new vocabulary.

☑ Trainer vectors represent ‘memories’ whereas filters represent the most important amalgamation of such memories.

Engene is continuously refreshed

Engene never stops evolving populations of filters.

As new vectors enter an ensemble, evolution continues.

The steps are:

* retrieve content
* build a digest
* present content
* receive feedback signals
* process feedback signals
* adjust populations
* evolve filters
* filter

rinse and repeat...

(sounds a lot like an event loop doesn't it?:-)

An example: processing of a positive signal

Following the selection of a digest entry from the user the associated document is added into the corresponding subject of interest ensemble.

1. The digest entry document is processed into a multi-dimensional vector

2. Then it is made into an individual which is placed in the population of filters.

3. The same vector is also added into the population of trainers.

3. During addition to a population the new one replaces one of the weakest ones if the maximum number of vectors has been reached. The weakness of filters is indicated in filter populations by their fitness due to the last evolution, whereas for the trainer population by their age in the system.

... and then evolution continues

An example: negative signals

Very strong negative:

This signal leads to the deletion from the system of the vectors responsible for its reception.

When a user selects to delete an entry from the digest, this signal is received and the filter(s) which recommended it are removed together with the trainers most similar to them.

Negative:

When received, the filter(s) that produced the recommendation for the associated digest entry are penalised in the subsequent evolution.

Before selection of individuals for reproduction is completed the fitness of the penalised filter(s) is reduced (by a factor which currently is set to 20%).

Presentation is key in feedback interpretation!

Facebook becoming mobile ad platform

One of the most explosive successes over the past year has been Apple's iPhone. Now, with tens of thousands of developers building applications to run on the iPhone, these developers are looking for the next step: a nice way to make money from all the users they're getting through the iPhone.

on VentureBeat [remove] [read]

Nintendo patent dispute sorted

A federal court has dismissed a lawsuit brought against Nintendo over patents related to the Wii and GameCube game controllers.

Nintendo said it is pleased with the decision. But the company has been the regular target of patent holders who are trying to collect royalties.

on VentureBeat [remove] [read]

AIG Posts List of Beneficiaries of Government Largesse Counterparties

As foretold, Goldman tops the list. From the Financial Times (hat tip reader Dwight):
AIG paid out $22.4bn of collateral related to credit default swaps, $27.1bn to help cancel swaps and another $43.7bn to satisfy the obligations of its securities lending operation. The payments were made between September 16 and the end of last year.

On Naked capitalism [remove] [read]

Where does twitter time come from?

That sucking sound you hear coming from PCs, Laptops and phones every where is the drain of hours upon hour of people twittering day and night. Twitter is capturing the imagination of millions lately. From Ashton Kutcher to Shaq to yours truly @mcuban. Businesses are tweeting too.

Twittering is undeniably of interest for any number of reasons, but the question I have, is where does Twitter Time come from ? Keeping an eye on the stream of tweets that can come your way, or watching tweets to keep up with what people are discussing or just curiousity and the voyeurism that is part of tweeting can suck hours in the day. So what did we used to do that we have replaced with twittering ?

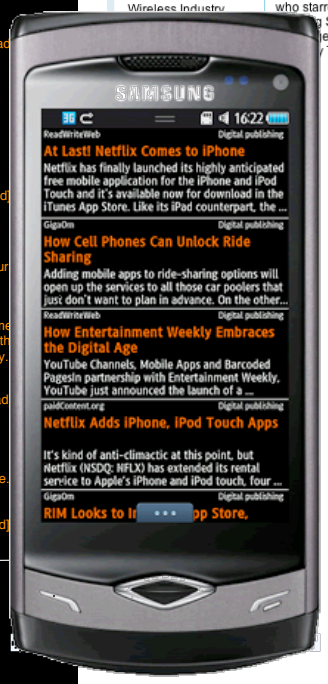
on Blog Maverick [remove] [read]

Tomtom vs Microsoft

The whole FAT licensing saga between Microsoft and TomTom just got a whole lot more complicated. Microsoft sued TomTom because the satnav maker had not licensed FAT from Microsoft, even though several others have. This left TomTom in a difficult position: not license it, and face legal penalties - license it, and violate the GPL.

on OSNews [remove] [read]

+ Intel: AMD Has broken 2001 cross licencing agreement
+ Nokia fires 1700
+ Pragmaticomm release Ruby 1.9.1p0 for Symbian OS
+ Twine could soon surpass Delicious, prepares ontology authoring tool
+ Toyotas residual values seen falling more than competitors



Important 'details' :

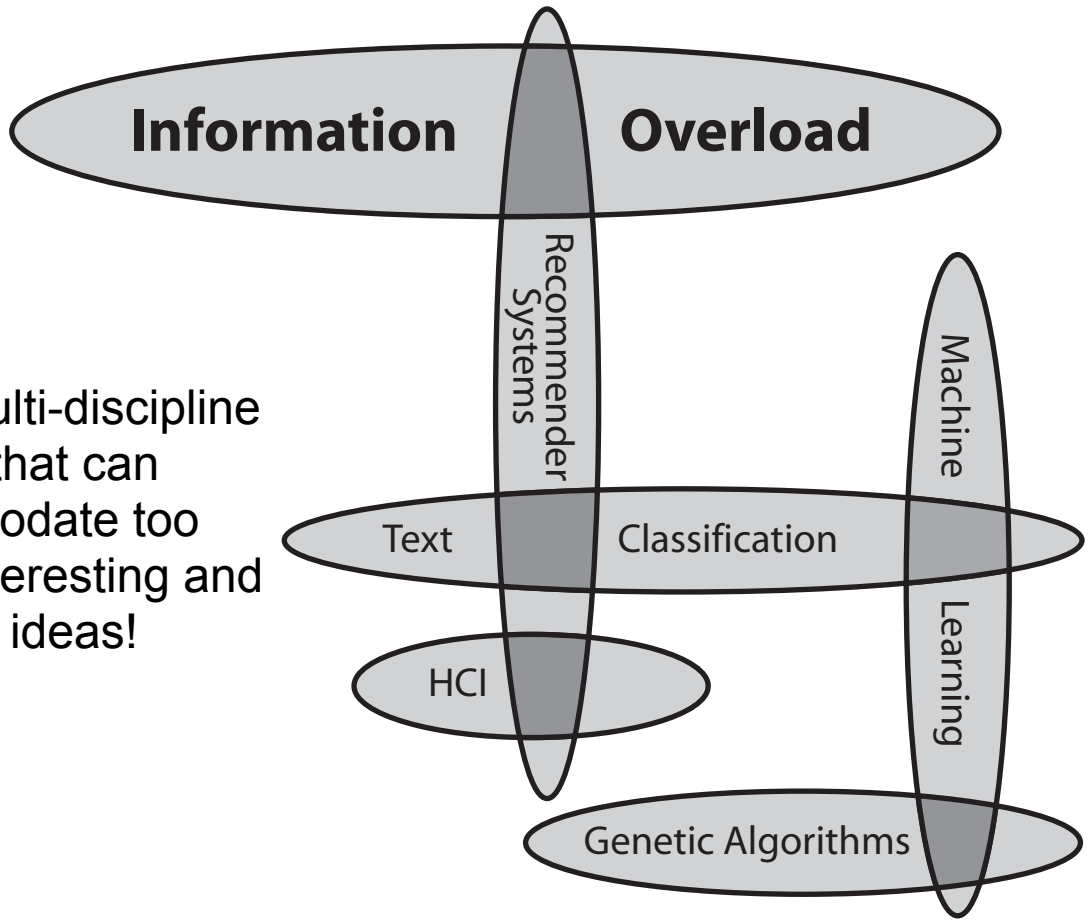
* The presentation layout.
* The locus of a selection.
* The order of selections.
* Whether there is a summary or not.
* The number of selected entries.
* The medium used (mobile, Web, audio)
* ...

So why use GAs in recommenders after all?

- No over-fitting
- Serendipity
- Excellent exploration as well as good filtering
- Excellent adaptation to concept drifts when docs are added to the gene pool
- No premature convergence towards an average profile
- Gene pool (terms) diversity

Engene is a soft one-class genetic algorithm classifier that can operate unattended in the core of a content recommender system. Such recommender may use only implicit feedback.

About the document recommendation domain



This is just a high-level over-simplified view!

It is a multi-discipline domain that can accommodate too many interesting and practical ideas!

It is vital to deal with information overload

There is immense practical, academic and commercial value in dealing with RecSys!

Thank you :-)

www.pagonis.org/Publications.html

www.pragmaticomm.com