Machine Learning and Statistical NLP Theory

Machine Learning for NLP—Theory and Background

- All tasks are classification (almost)
- Distributional semantics—less intelligence plus much more data equals a pretty good result!
- All data are points in hyperspace (almost)
- Algorithms and approaches
- Getting the best out of ML

Using Machine Learning in Bio-NLP

- We have just seen **chunk recognition**, which is the type of work we've been doing here with symptom identification
  - Facilitates statistical analysis for research and data visualisation, as well as other ML tasks by providing features
- We might also want to:
  - do **classification**, e.g. finding patients experiencing first episode psychosis or predicting suicide attempts from medical records
  - look for **relationships,** e.g. between illnesses and symptoms, or between drugs and adverse events
  - explore **unsupervised** data to find patient **clusters** and identify new syndromes or predictive patterns

Behind the Scenes in ML

- Much machine learning consists of trying to predict CLASS from ATTRIBUTES, and data needs to be representable as something like this:

ATTRIBUTES

CLASS

INSTANCES

Clouds?	Rain?	Temperature?	Go to park?
y	y	10	**n**
n	n	20	**y**
n	y	30	**n**

Chunk Recognition as Classification

- Batch Learning PR turns chunk recognition into a series of classification tasks

Capitalised?	Noun?	ANNIE says person?	Start of person?
y	y	y	**y**
y	y	n	**y**
n	n	n	**n**

Capitalised?	Noun?	ANNIE says person?	End of person?
y	y	y	**y**
y	y	n	**y**
n	n	n	**n**

Capitalised?	Noun?	ANNIE says location?	Start of location?
y	y	y	**y**
:	:	:	**:**

Chunk Recognition as Classification

- Having decided where persons (or symptoms) start and end, some simple logic pairs them up
- Alternative approaches
  - Separate NER stage finds the spans, then you can classify them
  - Rather than using beginnings and ends (or neither) as class, “BIO” is a popular approach (beginning, inside, outside)

Relationship Extraction as Classification

In a BBC interview, Tony Trotter of Analysts Inc said that Goldman Sachs front-man Tourre had his nose in the trough

Relationship Extraction as Classification

Person

In a BBC interview, Tony Trotter of Analysts Inc said that Goldman Sachs front-man Tourre had his nose in the trough

Person

Relationship Extraction as Classification

Org. Person Organization

In a BBC interview, Tony Trotter of Analysts Inc said that Goldman Sachs front-man Tourre had his nose in the trough

Organization Person

Relationship Extraction as Classification

Org. Person Organization

In a BBC interview, Tony Trotter of Analysts Inc said that Goldman Sachs front-man Tourre had his nose in the trough

Organization Person

Tony Trotter --- Analysts Inc

Relationship Extraction as Classification

Org. Person Organization

In a BBC interview, Tony Trotter of Analysts Inc said that Goldman Sachs front-man Tourre had his nose in the trough

Organization Person

Tony Trotter --- Analysts Inc

Tourre --- Goldman Sachs

Relationship Extraction as Classification

Org. Person Organization

In a BBC interview, Tony Trotter of Analysts Inc said that Goldman Sachs front-man Tourre had his nose in the trough

Organization Person

Tony Trotter --- BBC

Tony Trotter --- Analysts Inc

Tourre --- Goldman Sachs

Relationship Extraction as Classification

Org. Person Organization

In a BBC interview, Tony Trotter of Analysts Inc said that Goldman Sachs front-man Tourre had his nose in the trough

Organization Person

Tony Trotter --- BBC

Tony Trotter --- Analysts Inc

Tony Trotter --- Analysts Inc

Tourre --- Goldman Sachs

Relationship Extraction as Classification

Org. Person Organization

In a BBC interview, Tony Trotter of Analysts Inc said that Goldman Sachs front-man Tourre had his nose in the trough

Organization Person

Tony Trotter --- BBC

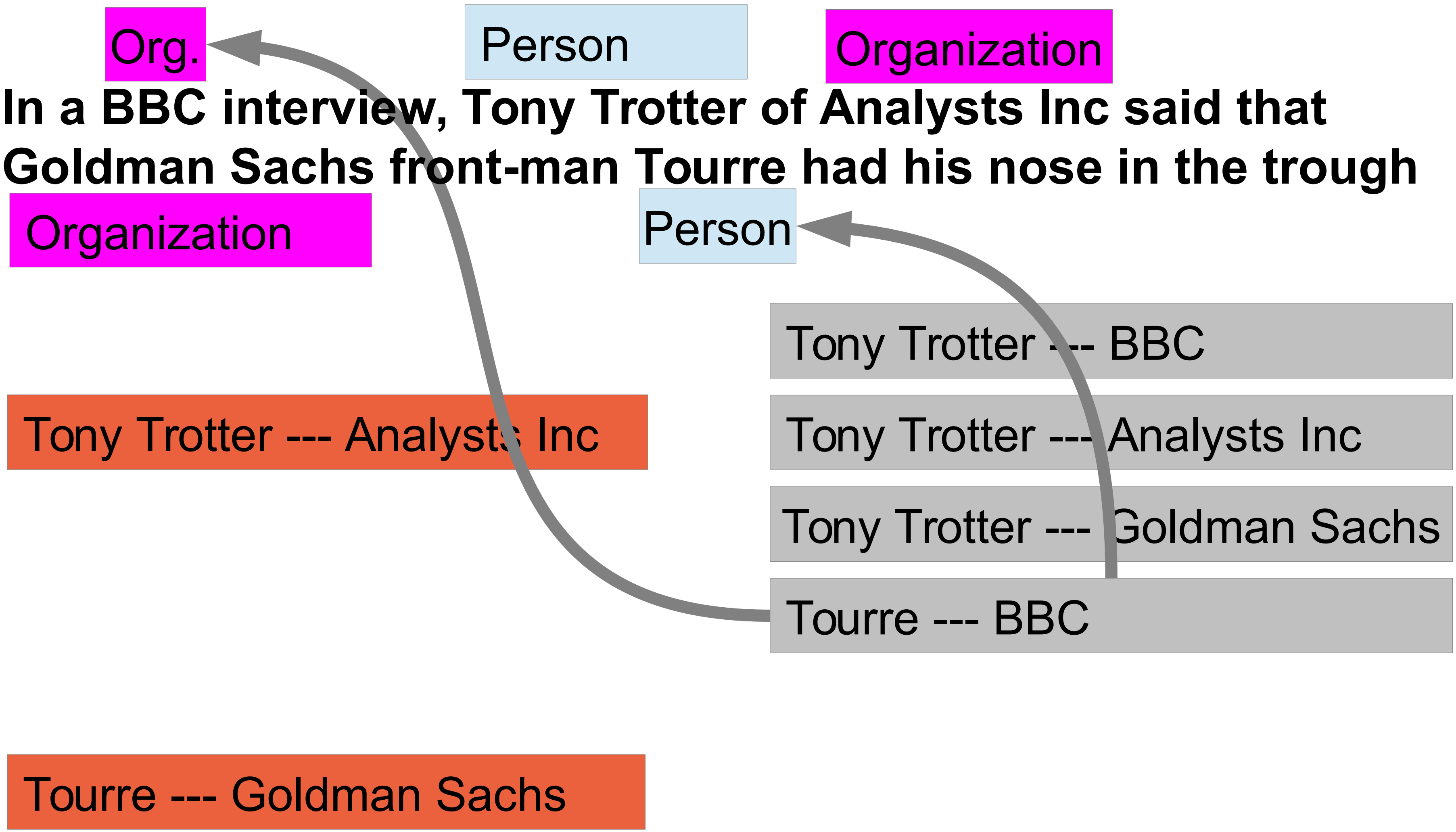
Tony Trotter --- Analysts Inc

Tony Trotter --- Goldman Sachs

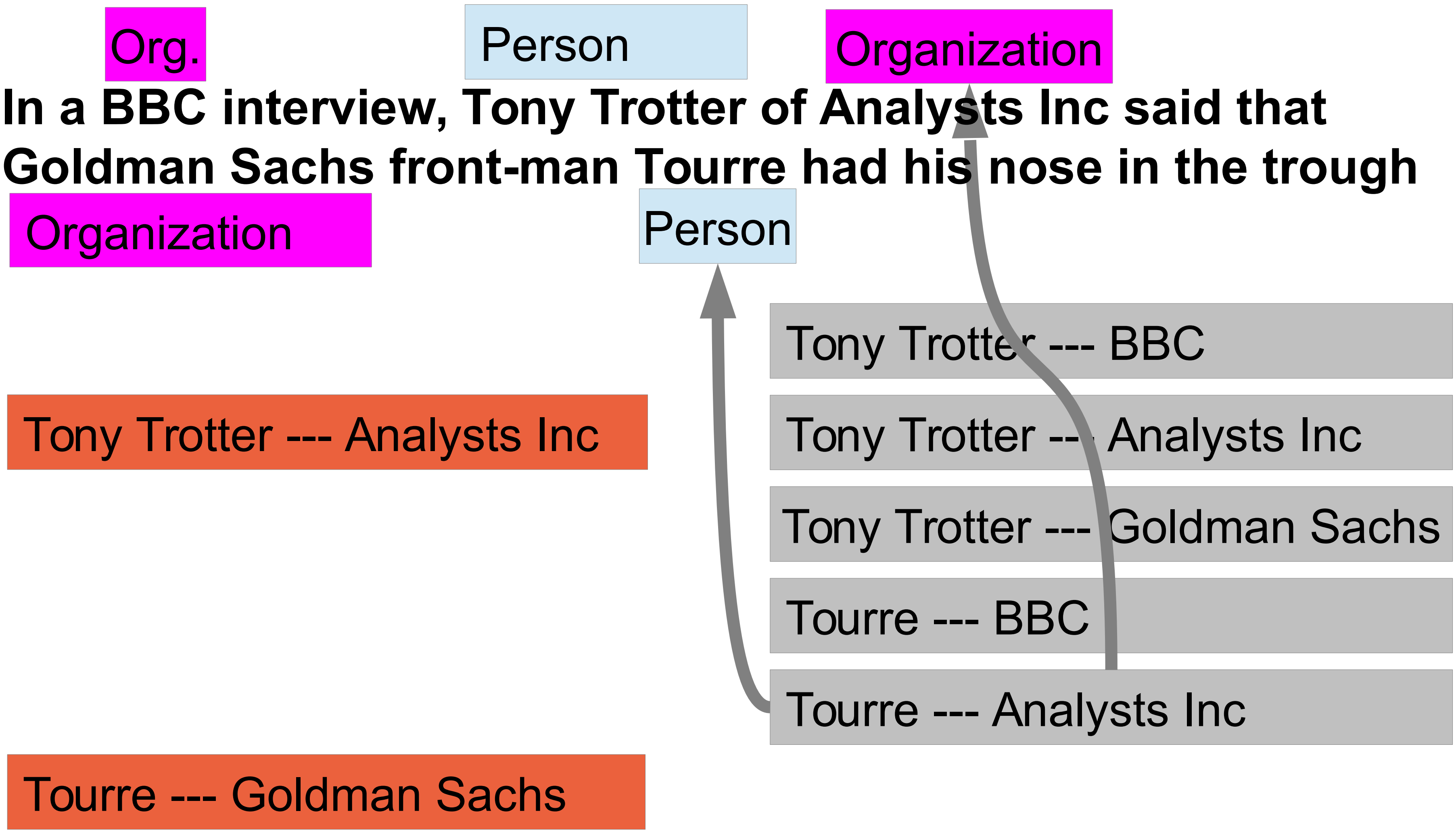
Tony Trotter --- Analysts Inc

Tourre --- Goldman Sachs

Relationship Extraction as Classification



Relationship Extraction as Classification



Relationship Extraction as Classification

Org. Person Organization

In a BBC interview, Tony Trotter of Analysts Inc said that Goldman Sachs front-man Tourre had his nose in the trough

Organization Person

Tony Trotter --- BBC

Tony Trotter --- Analysts Inc

Tony Trotter --- Goldman Sachs

Tourre --- BBC

Tourre --- Analysts Inc

Tourre --- Goldman Sachs

Tony Trotter --- Analysts Inc

Tourre --- Goldman Sachs

Relationship Extraction as Classification

Org. Person Organization

In a BBC interview, Tony Trotter of Analysts Inc said that Goldman Sachs front-man Tourre had his nose in the trough

Organization Person

Tony Trotter --- BBC

Tony Trotter --- Analysts Inc | Tony Trotter --- Analysts Inc

Tony Trotter --- Goldman Sachs

Tourre --- BBC

Tourre --- Analysts Inc

Tourre --- Goldman Sachs | Tourre --- Goldman Sachs

Relationship Extraction as Classification

Org. | Person | Organization

In a BBC interview, Tony Trotter of Analysts Inc said that Goldman Sachs front-man Tourre had his nose in the trough

Organization | Person

	−	Tony Trotter --- BBC
Tony Trotter --- Analysts Inc		Tony Trotter --- Analysts Inc
	−	Tony Trotter --- Goldman Sachs
	−	Tourre --- BBC
	−	Tourre --- Analysts Inc
Tourre --- Goldman Sachs		Tourre --- Goldman Sachs

Relationship Extraction as Classification

Org. | Person | Organization

In a BBC interview, Tony Trotter of Analysts Inc said that Goldman Sachs front-man Tourre had his nose in the trough

Organization | Person

	−	Tony Trotter --- BBC
Tony Trotter --- Analysts Inc	+	Tony Trotter --- Analysts Inc
	−	Tony Trotter --- Goldman Sachs
	−	Tourre --- BBC
	−	Tourre --- Analysts Inc
Tourre --- Goldman Sachs	+	Tourre --- Goldman Sachs

Relationship Extraction as Classification

- How does that look as instances?
- What about features?
  - As well as features of the person and organization mentions, we can also use features such as distance between them

Person	Organization	Distance	CLASS
Trotter	Goldman Sachs	27	**n**
Trotter	Analysts Inc	4	**y**
Tourre	Goldman Sachs	11	**y**
:	:	:	**:**

Unsupervised Approaches

- Everything we looked at so far requires annotated data and forces the classes we decide are important
- Unsupervised techniques use unlabeled data

SIMILAR

	treatment	mg	anti-psychotic	placebo	patients
olanzapine	110	86	76	75	73
clozapine	70	30	78	0	89
vinegar	15	0	0	0	0

NOT SO SIMILAR

How are these tasks achieved?

- We've seen that the central paradigm is classification—determining what, of a number of options (classes), something (the instance) is, given available information (attributes/features)
- We've seen how to conceptualize problems in this way, but how do we actually do it?
  - Attribute choice and information representation
  - Algorithm/technique choice

A Bit of Statistical NLP Philosophy

- Using simple, easily available features to find sophisticated relationships
- In the chunking task we used parts of speech from automatic parsing as well as gazetteer-based attempts at NER (from lists)
- You may also have tried the word strings themselves
- We can't, though, include a deeper understanding of the semantics of the document, that a human might use
- How well does this work? How well can it work?

Wombling and snetches

The Captain's side raked first. Tom staked. The hired sportsmen played so hard that they **wombled** too fast, and were shaky with the rakes. Tom fooled around the way he always did, and all his stakes dropped true. When it was his turn to rake he did not let Captain Najork and the hired sportsmen score a single rung, and at the end of the **snetch** he won by six ladders.

(How Tom beat Captain Najork and his hired sportsmen
Russell Hoban and Quentin Blake)

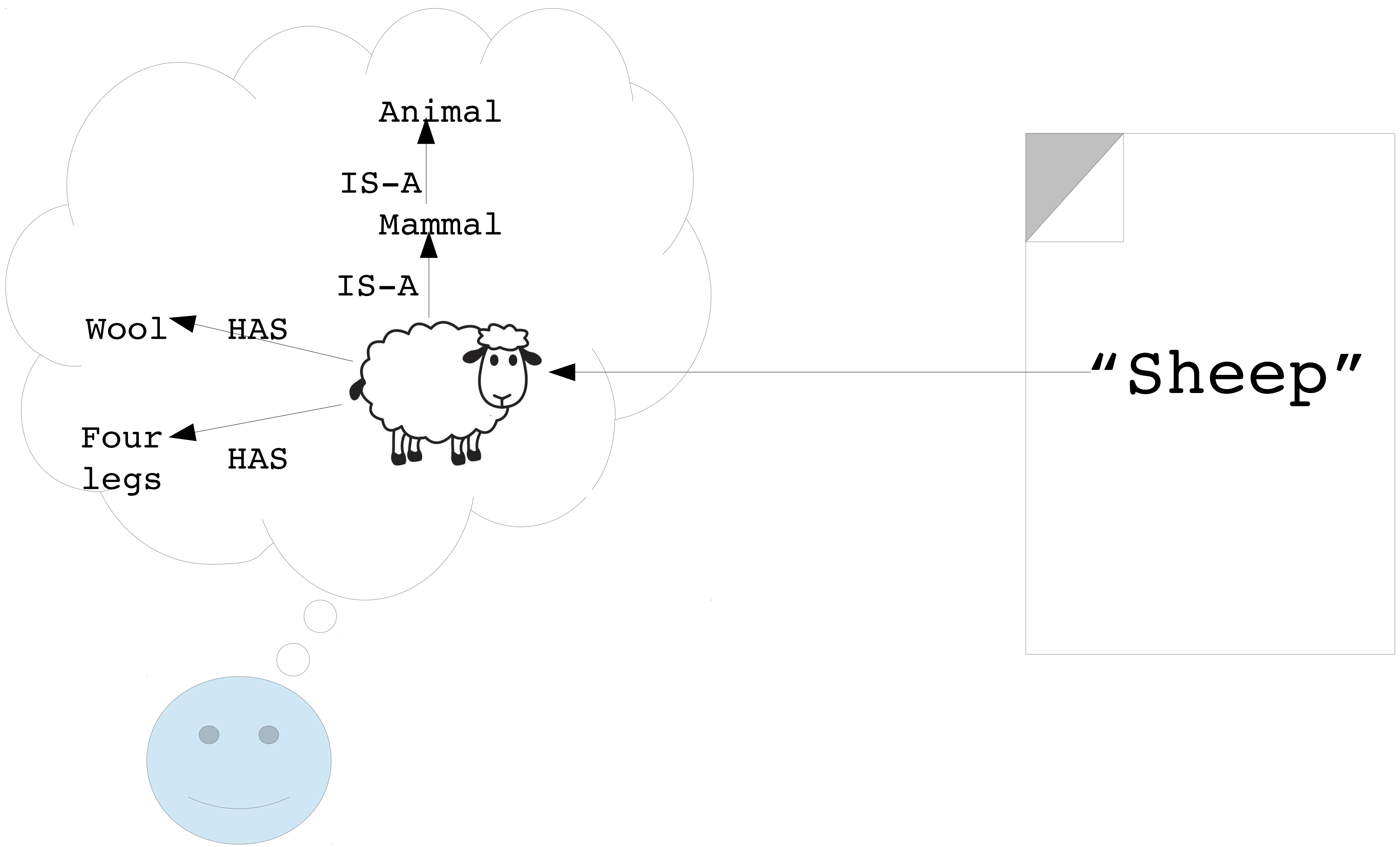
The distributional hypothesis

- “The meaning of a word is its use in language” (Wittgenstein)
- The contexts in which words appear correlate with their meaning
- We understand a word by its distribution: the set of contexts in which it is found
- “Don't think, but look!” (Wittgenstein)

Formal semantics and lexical semantics

- A contrast to distributional semantics
- Formal semantics
  - models the relationship between language and the world
  - defines meaning in terms of this model
  - defines languages in terms of formal logic
- The lexicon is defined as mappings from words to structured, conceptual knowledge

Lexical semantics



Complementary

- Distributional semantics is based on **statistics**, formal semantics on **mathematics**
- Distributional semantics is **differential**, lexical semantics is **referential**
- Distributional semantics is based on large **corpora**, lexical semantics (more often) on **structured lexicons**
- **Gathering a corpus** is easier than **building a lexicon**

Lack of grounding (after Bruni, 2013)

- Task: finding semantic features for sheep
- Generated by psychology students (McRae, 2005):
  - have four legs, say “Baah”, have wool, are white
- Generated from texts (Baroni, 2010):
  - live on farms, graze, get scrapie
- Collocates (nearby words) in Google (via WebCorp):
  - black, crc, wool, electric, industry, goats...
- Weakly supervised extraction of features from large corpora gives P=24%, R=48% over generated properties (Kelly, 2010)

Collocations

```
VE: The authors compared the efficacy of olanzapine and lithium in the prevention of mood
nd received open-label co-treatment with olanzapine and lithium for 6-12 weeks. Those meet
in Pharmacokinet. 1999 Sep;37(3):177-93. Olanzapine. Pharmacokinetic and pharmacodynamic p
patients with schizophrenia confirm that olanzapine is a novel antipsychotic agent with br
d with traditional antipsychotic agents, olanzapine causes a lower incidence of extrapyram
urbation of prolactin levels. Generally, olanzapine is well tolerated. The pharmacokinetic
okers and men have a higher clearance of olanzapine than women and nonsmokers. After admin
rred between olanzapine and alcohol, and olanzapine and imipramine, implying that patients
:485-92. doi: 10.1192/bjp.bp.107.037903. Olanzapine for the treatment of borderline person
o evaluate treatment with variably dosed olanzapine in individuals with borderline persona
double-blind trial, individuals received olanzapine (2.5-20 mg/day; n=155) or placebo (n=1
rried-forward methodology. RESULTS: Both olanzapine and placebo groups showed significant
p. CONCLUSIONS: Individuals treated with olanzapine and placebo showed significant but not
he types of adverse events observed with olanzapine treatment appeared similar to those ob
is study compared three dosage ranges of olanzapine (5 +/- 2.5 mg/day [Olz-L], 10 +/- 2.5
```

What do we know about Olanzapine from its collocations?

- Deep learning and word collocations produces the following!
  - Human - Animal = Ethics
  - Stock Market ≈ Thermometer
  - Library - Books = Hall
  - Obama + Russia - USA = Putin
  - Iraq - Violence = Jordan
  - President - Power = Prime Minister
  - Politics - Lies = Germans

  (From http://byterot.blogspot.co.uk, based on word2vec software)
- Olanzapine is most similar to: [["risperidone",0.7404874563217163], ["aripiprazole",0.7372795939445496],["quetiapine",0.7360421419143677], ["ziprasidone",0.7347999811172485],["fluoxetine",0.6984522938728333]]

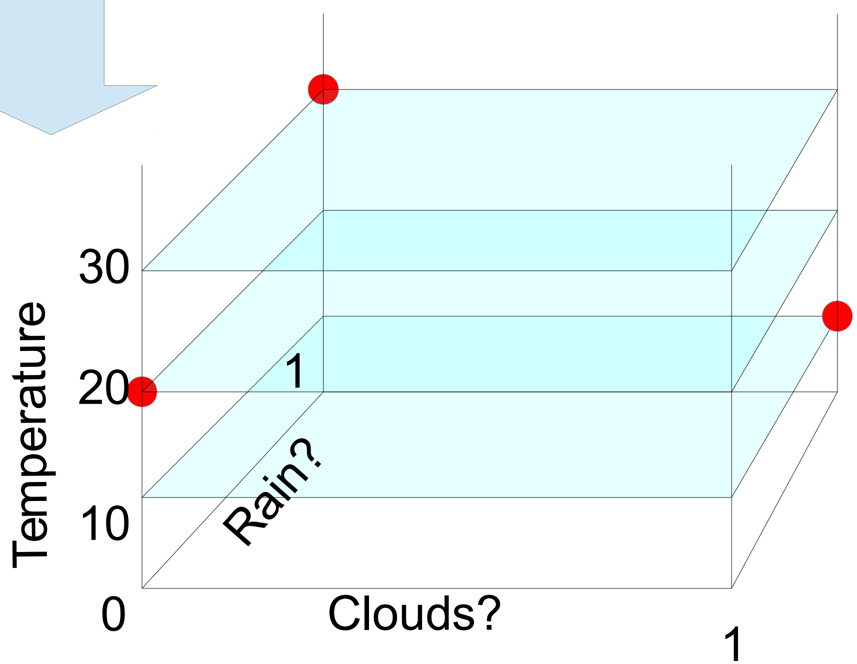
Limitations of Distributional Semantics

- May not model more complex nuances of meaning
  - Though note the deep learning example earlier uses multiple layers of abstraction to encode semantic complexity
- Simple features may mean that e.g. information inherent in word order is lost
  - This is why we often create more complex features such as explicit representations of negated expressions

All Data are a Point in Hyperspace

Clouds?	Rain?	Temperature?	Go to park?
y	y	10	n
n	n	20	y
n	y	30	n

Boolean values map to 1 or 0



All Data are a Point in Hyperspace

- To be a point in space, everything needs to be a number
- How can words be a point in hyperspace (or parts of speech, or orthographical categories)?

Bag of words:

the	cat	sat	on	mat
2	1	1	1	1

Word bigrams:

the cat	cat sat	sat on	on the	the mat
1	1	1	1	1

- “the mouse sat on the mat” → [2,0,1,1,1] (there is no dimension for mouse in the training data)
- Sparse vector format: [0:2,2:1,3:1,4:1]

What this means for ML

- Feature vectors have one dimension for every word that appeared in the corpus
- This means that NLP data often have a very large number of dimensions
  - Long training times
- Feature vectors are sparse
- Word bigrams or trigrams mean even more dimensions and even more sparse data, so use with caution!
  - These can also overfit smaller datasets leading to poor generalizability

So how would I ...

- .. find words most similar to Olanzapine? Find documents most relevant to a particular symptom?
- Not machine learning—we can do this by using vector space representations of documents and search terms intelligently

Contexts as matrices

- Build matrices of event frequencies, where events are words in documents
- Word-document matrices allow us to compare documents in terms of words that appear in them, and words in terms of their distribution over documents
  - That's just the vectors on the previous slide, one for each document, arranged in columns
- Word-word matrices tell us what words appear with what other words
  - You can get that by squaring the word-document matrix
- Word-sequence information is lost (at least in the simplest models)

Word-Word matrices

	treatment	mg	anti-psychotic	placebo	patients
olanzapine	110	86	76	75	73

- Top 5 collocates for olanzapine
- Collocates four to the left and right, from www.webcorp.org.uk
- Restricted to *.ncbi.nlm.nih.gov (i.e. mostly PubMed abstracts)
- Normalised to collocates per 1000 hits

Word-Word matrices

	treatment	mg	anti-psychotic	placebo	patients
olanzapine	110	86	76	75	73
clozapine	70	30	78	0	89

- Top 5 collocates for olanzapine
- Collocates four to the left and right, from www.webcorp.org.uk
- Restricted to *.ncbi.nlm.nih.gov (i.e. mostly PubMed abstracts)
- Normalised to collocates per 1000 hits

Word-Word matrices

	treatment	mg	anti-psychotic	placebo	patients
olanzapine	110	86	76	75	73
clozapine	70	30	78	0	89
vinegar	15	0	0	0	0

- Top 5 collocates for olanzapine
- Collocates four to the left and right, from www.webcorp.org.uk
- Restricted to *.ncbi.nlm.nih.gov (i.e. mostly PubMed abstracts)
- Normalised to collocates per 1000 hits

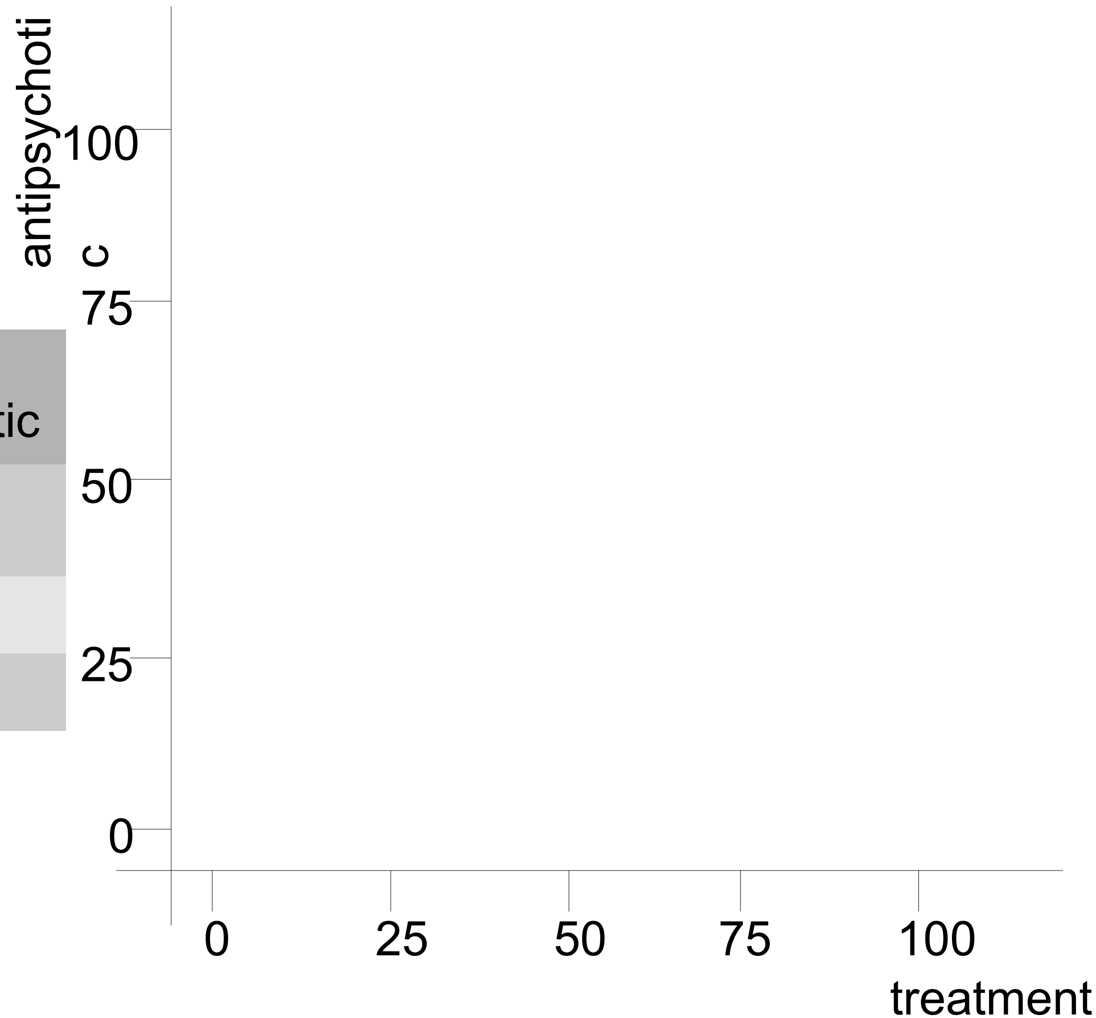
Word-Word matrices

	treatment	mg	anti-psychotic	placebo	patients	balsamic
olanzapine	110	86	76	75	73	0
clozapine	70	30	78	0	89	0
vinegar	15	0	0	0	0	109

- Top 5 collocates for olanzapine
- Collocates four to the left and right, from www.webcorp.org.uk
- Restricted to *.ncbi.nlm.nih.gov (i.e. mostly PubMed abstracts)
- Normalised to collocates per 1000 hits

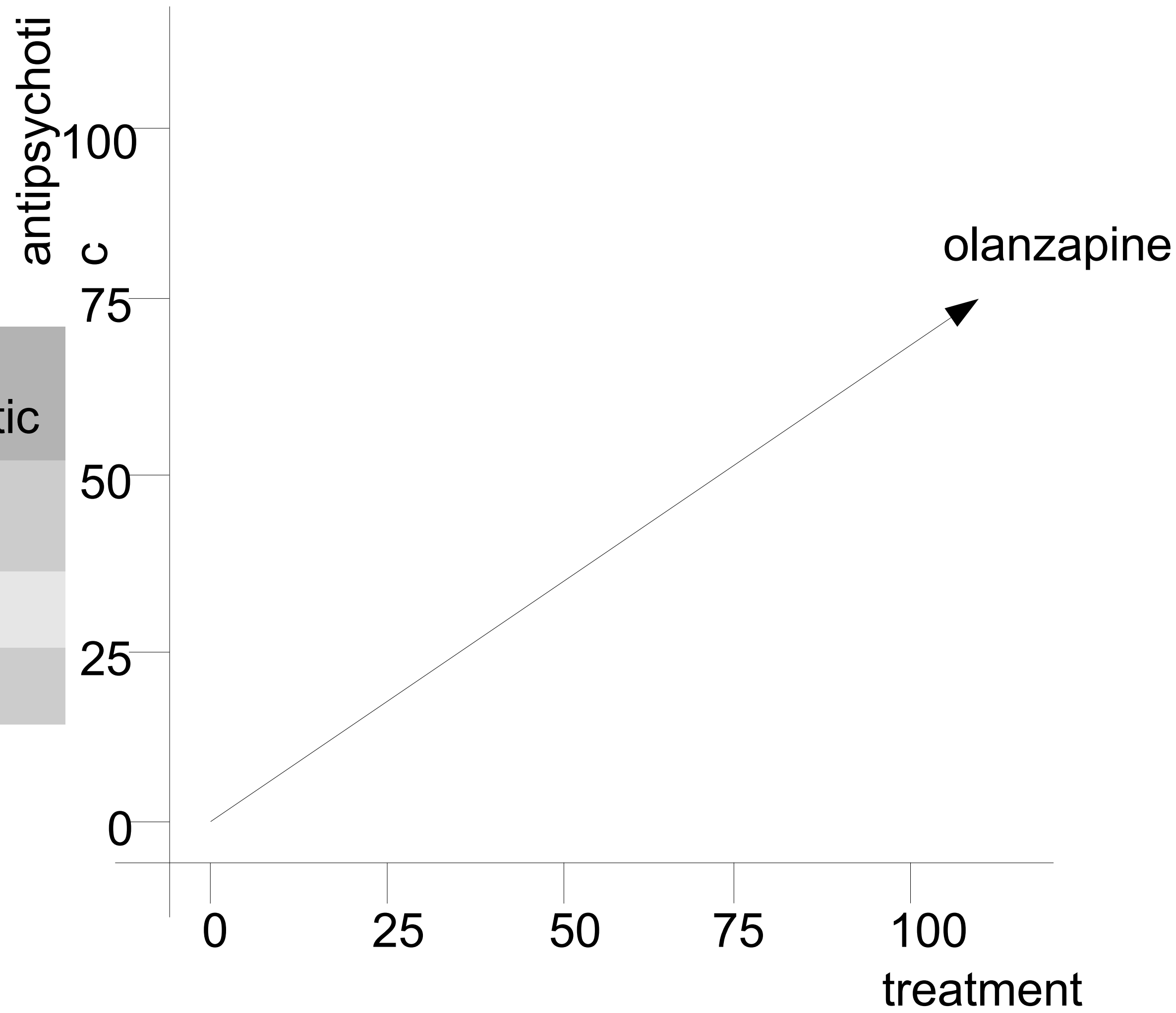
Relatedness

	treatment	anti-psychotic
olanzapine	110	76
clozapine	70	78
vinegar	15	0



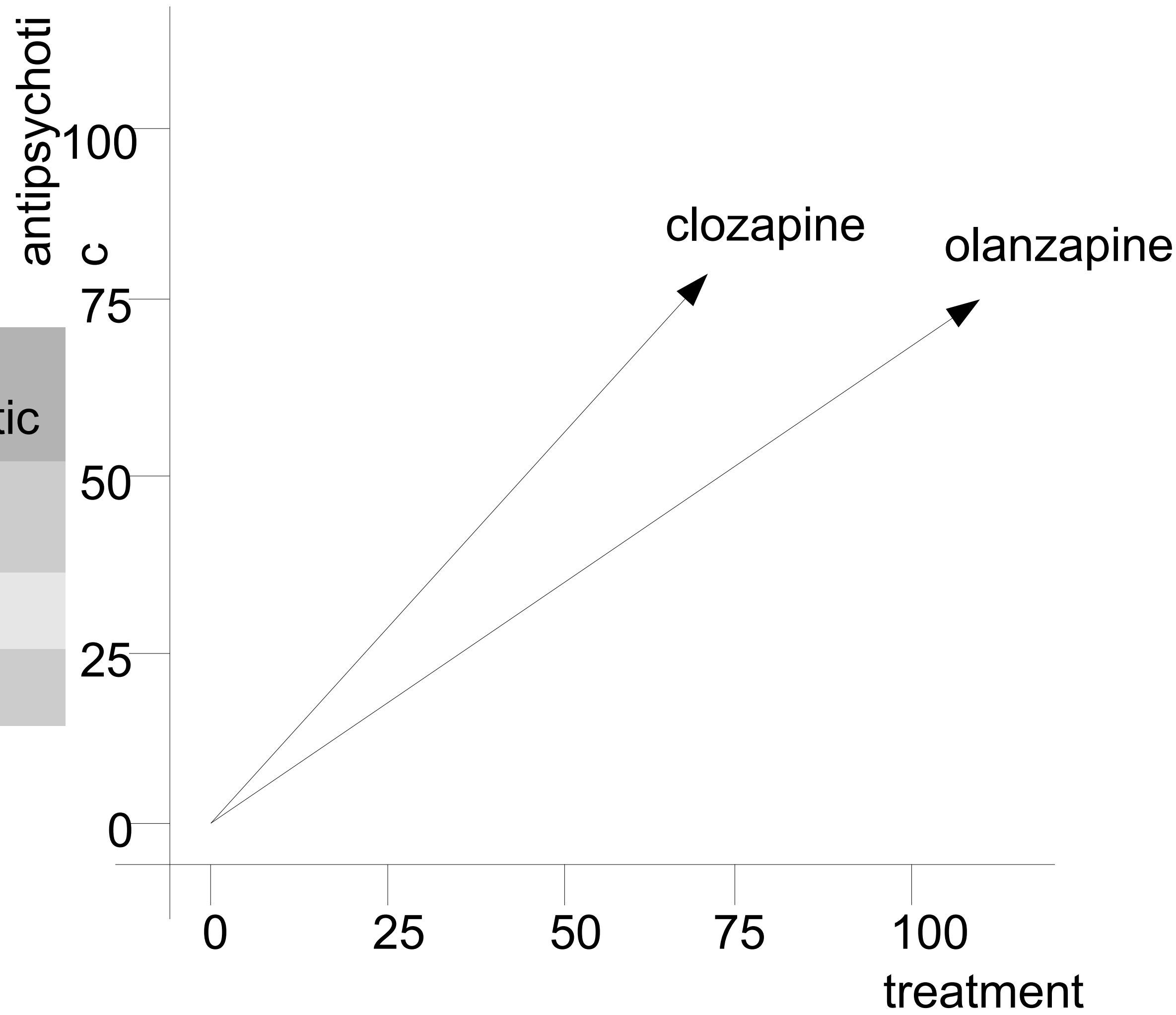
Relatedness

	treatment	anti-psychotic
olanzapine	110	76
clozapine	70	78
vinegar	15	0



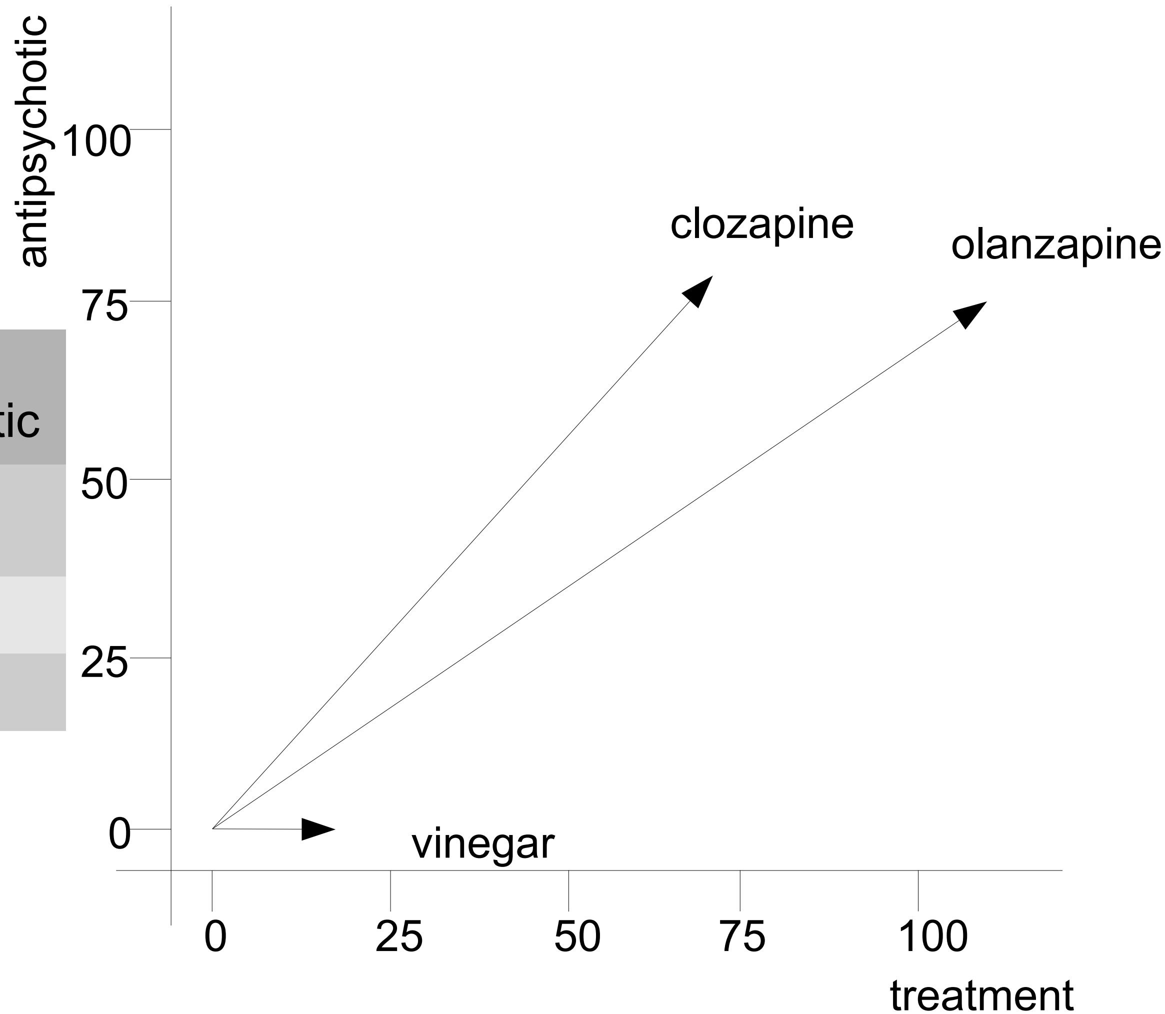
Relatedness

	treatment	anti-psychotic
olanzapine	110	76
clozapine	70	78
vinegar	15	0



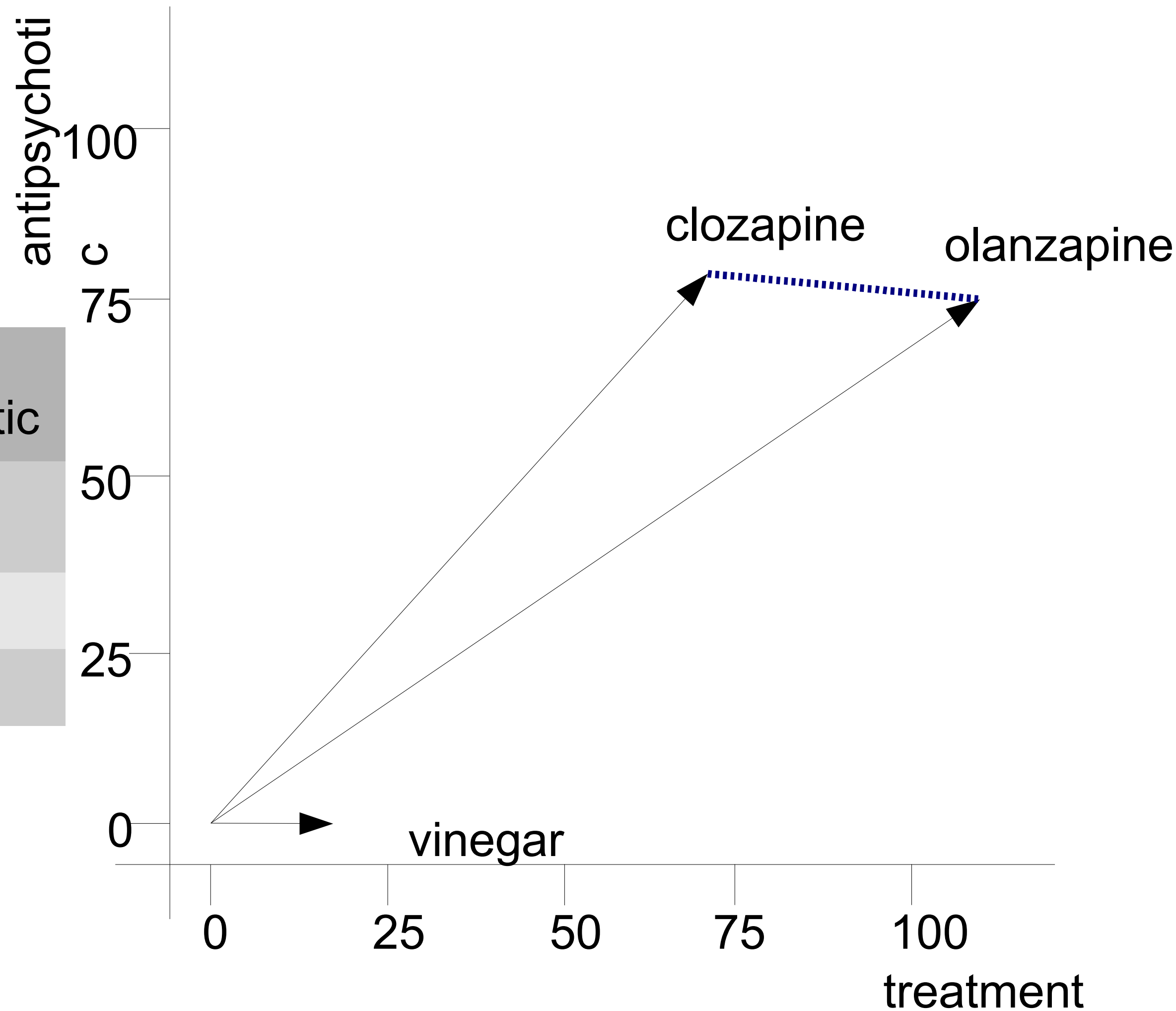
Relatedness

	treatment	anti-psychotic
olanzapine	110	76
clozapine	70	78
vinegar	15	0



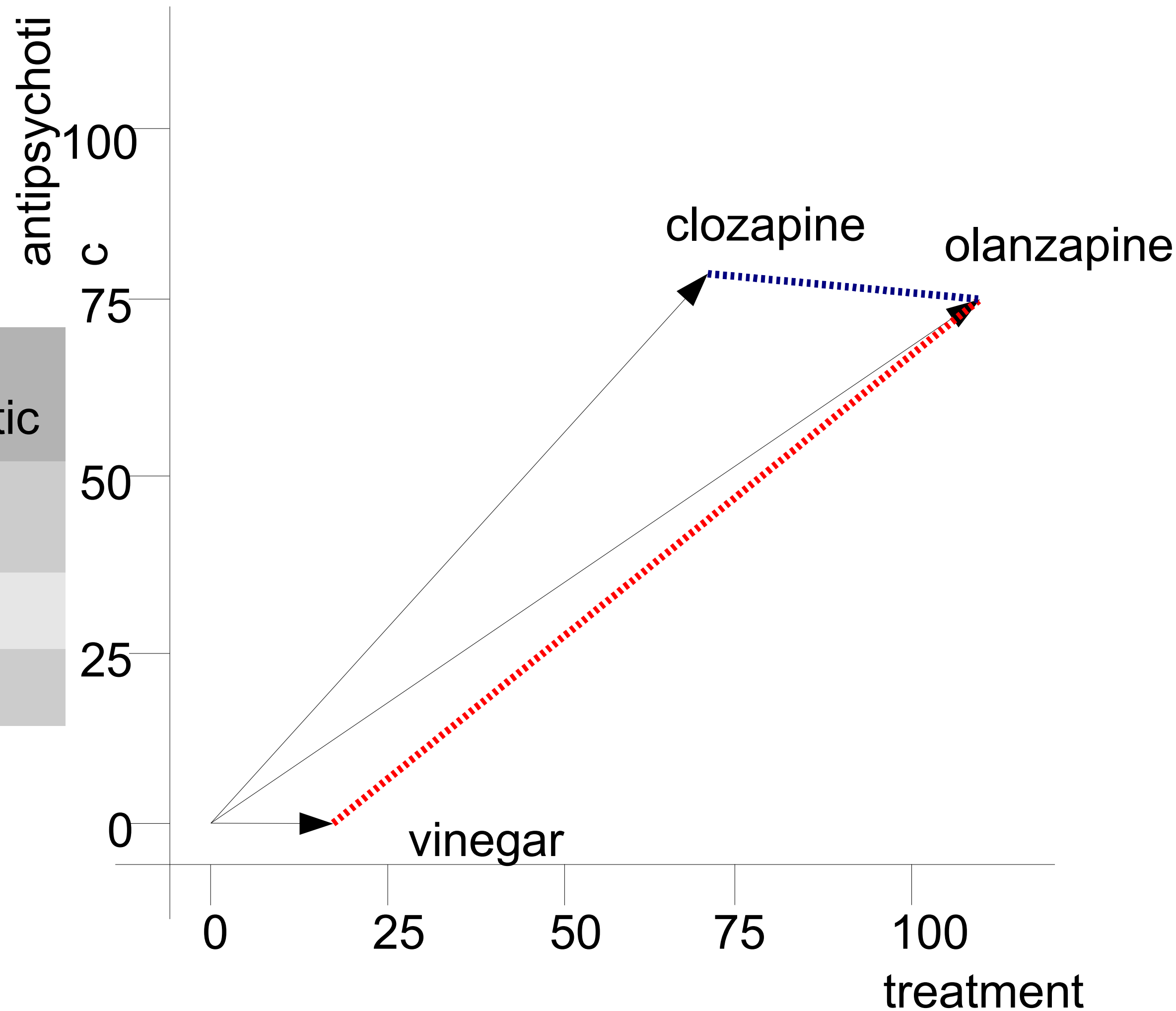
Relatedness

	treatment	anti-psychotic
olanzapine	110	76
clozapine	70	78
vinegar	15	0



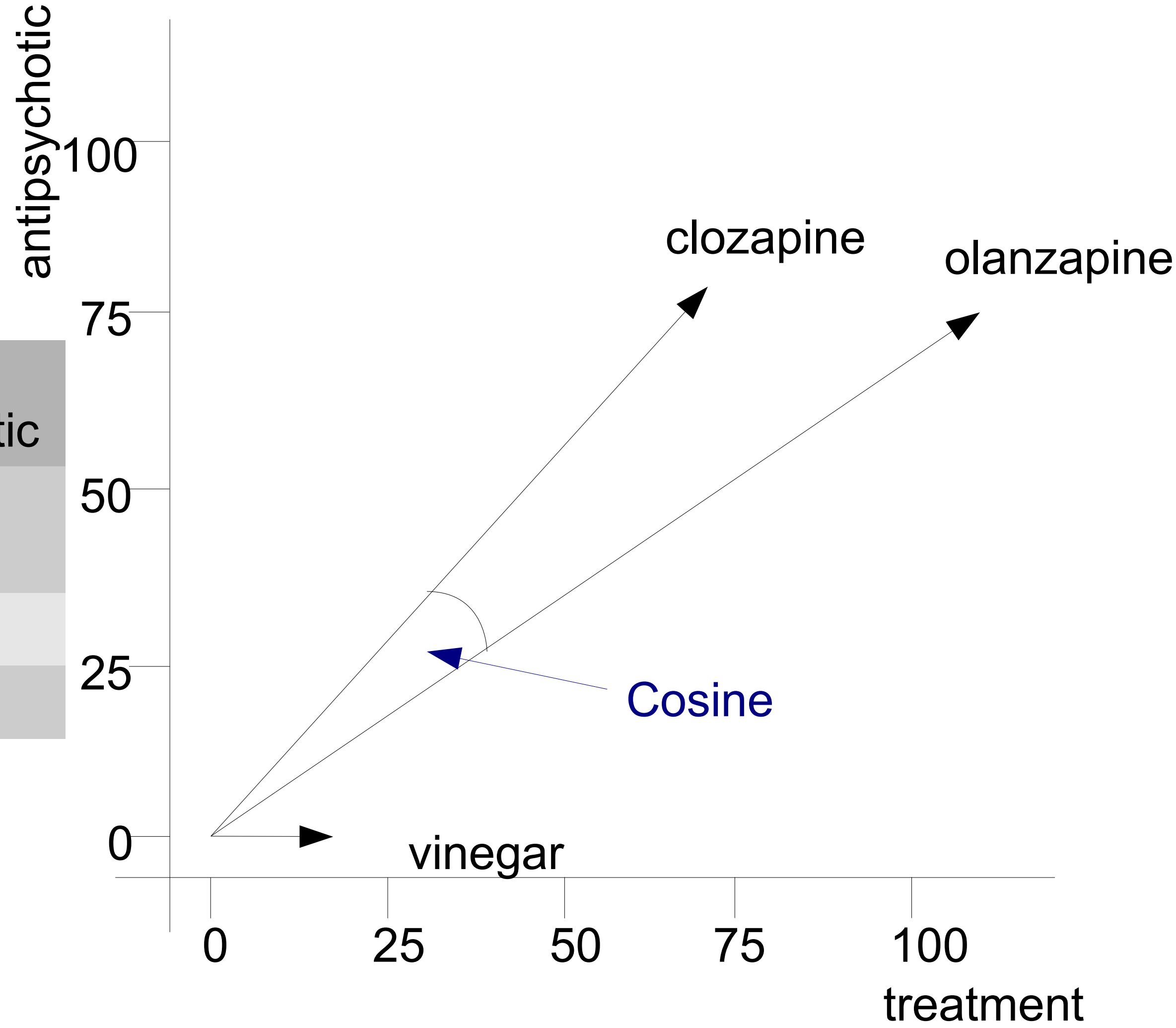
Relatedness

	treatment	anti-psychotic
olanzapine	110	76
clozapine	70	78
vinegar	15	0



Relatedness

	treatment	anti-psychotic
olanzapine	110	76
clozapine	70	78
vinegar	15	0



Word-Document Matrices

- Same approach applies to whole documents, not just words
  - Comparing documents for similarity, e.g. similarity to a prototype case that you want cases similar to
  - Using search terms to find relevant documents
  - Expanding terms with related terms to improve search etc.

	Blunted affect, poor rapport.	Good rapport, good eye contact.	Poor rapport, poor eye contact.
blunted	1	0	0
affect	1	0	0
poor	1	0	2
rapport	1	1	1
good	0	2	0
eye	0	1	1
contact	0	1	1

Practicalities

- This works better if you:
  - Intelligently select features
    - E.g. word stemming reduces sparsity, stop-word removal avoids unnecessary features
  - Weight toward important words
    - TF-IDF transform divides word counts by number of documents the word appears in, to favour discriminative words
  - Reduce dimensionality
    - Random Indexing reduces dimensions at little cost
    - Singular value decomposition generalizes
  - Choose similarity metric appropriately

How would I do this in GATE?

- Currently working on support for more sophisticated aided semantic modelling
- The simple case could be achieved using a Groovy script (more about Groovy later in the week!)

How would I ...

- ... classify patient records?
- We have labelled documents to train on

Document content

Class

Features

	“Blunted affect, poor rapport.”	“Good rapport, good eye contact.”	“Poor rapport, poor eye contact.”
	Not doing well	Doing well	Not doing well
blunted	1	0	0
affect	1	0	0
poor	1	0	2
rapport	1	1	1
good	0	2	0
eye	0	1	1
contact	0	1	1

Decision Trees

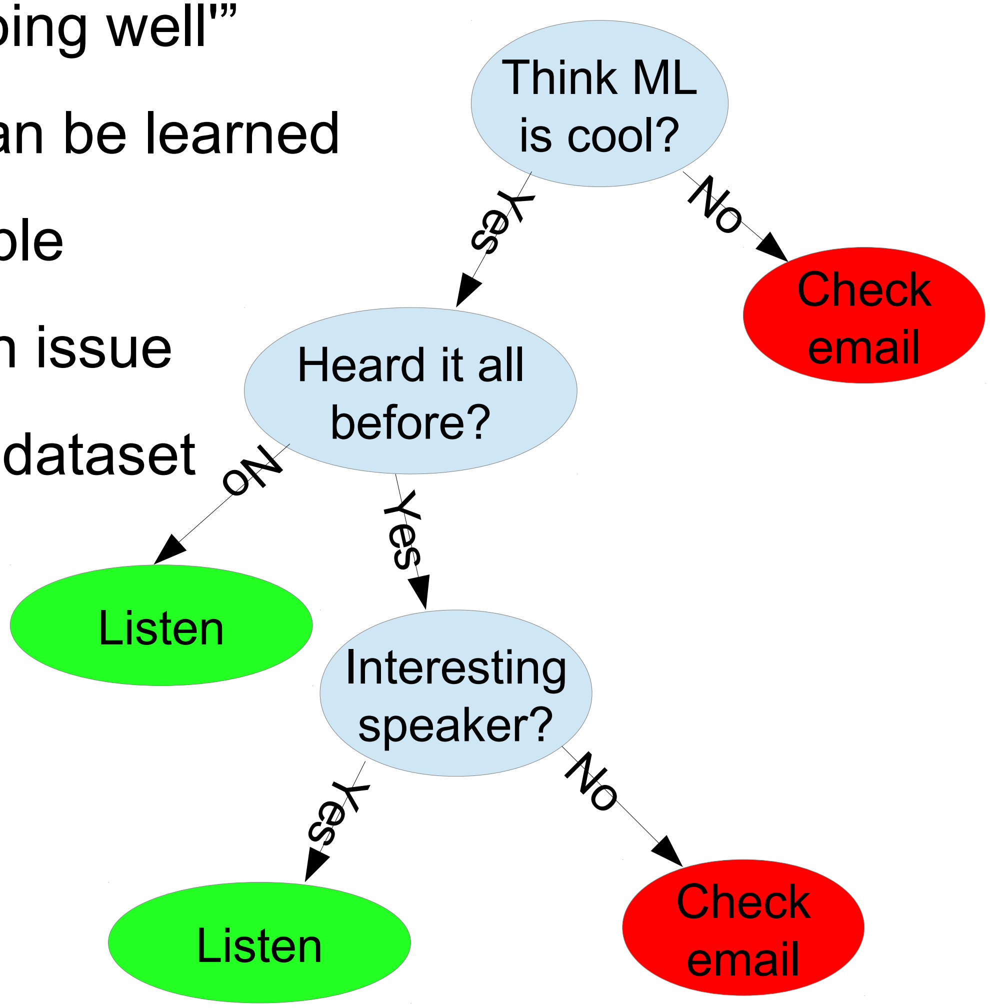
- What look like good features for this task?

	"Blunted affect, poor rapport."	"Good rapport, good eye contact."	"Poor rapport, poor eye contact."
	Not doing well	Doing well	Not doing well
blunted	1	0	0
affect	1	0	0
poor	1	0	2
rapport	1	1	1
good	0	2	0
eye	0	1	1
contact	0	1	1

- Do you think we could learn a rule to succeed at this task?

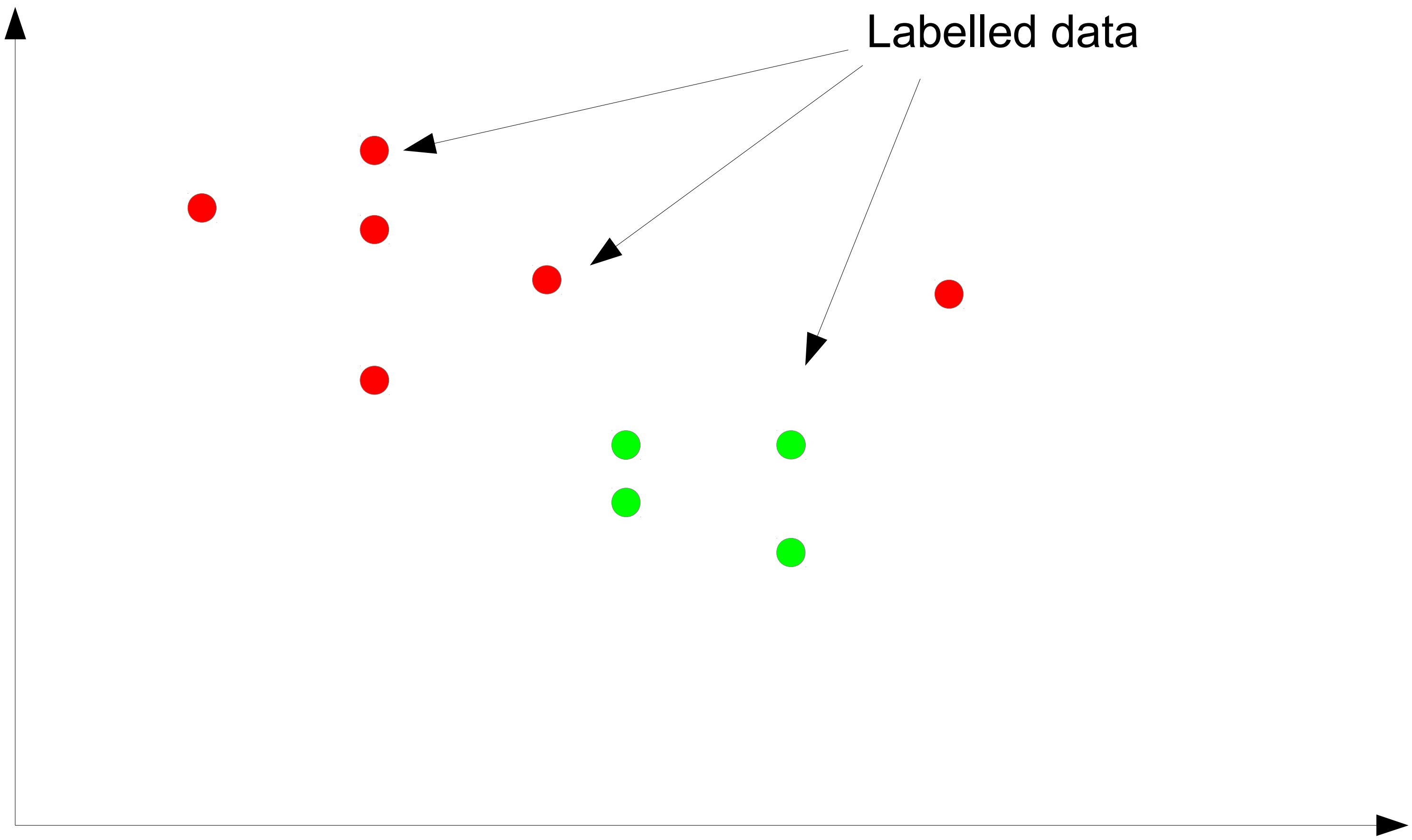
Decision Trees

- Decision tree approaches learn rules such as “if the word 'good' appears, classify as 'doing well'”
- Complex sequences can be learned
- Model is human-readable
- Data sparsity can be an issue
  - Every rule splits the dataset



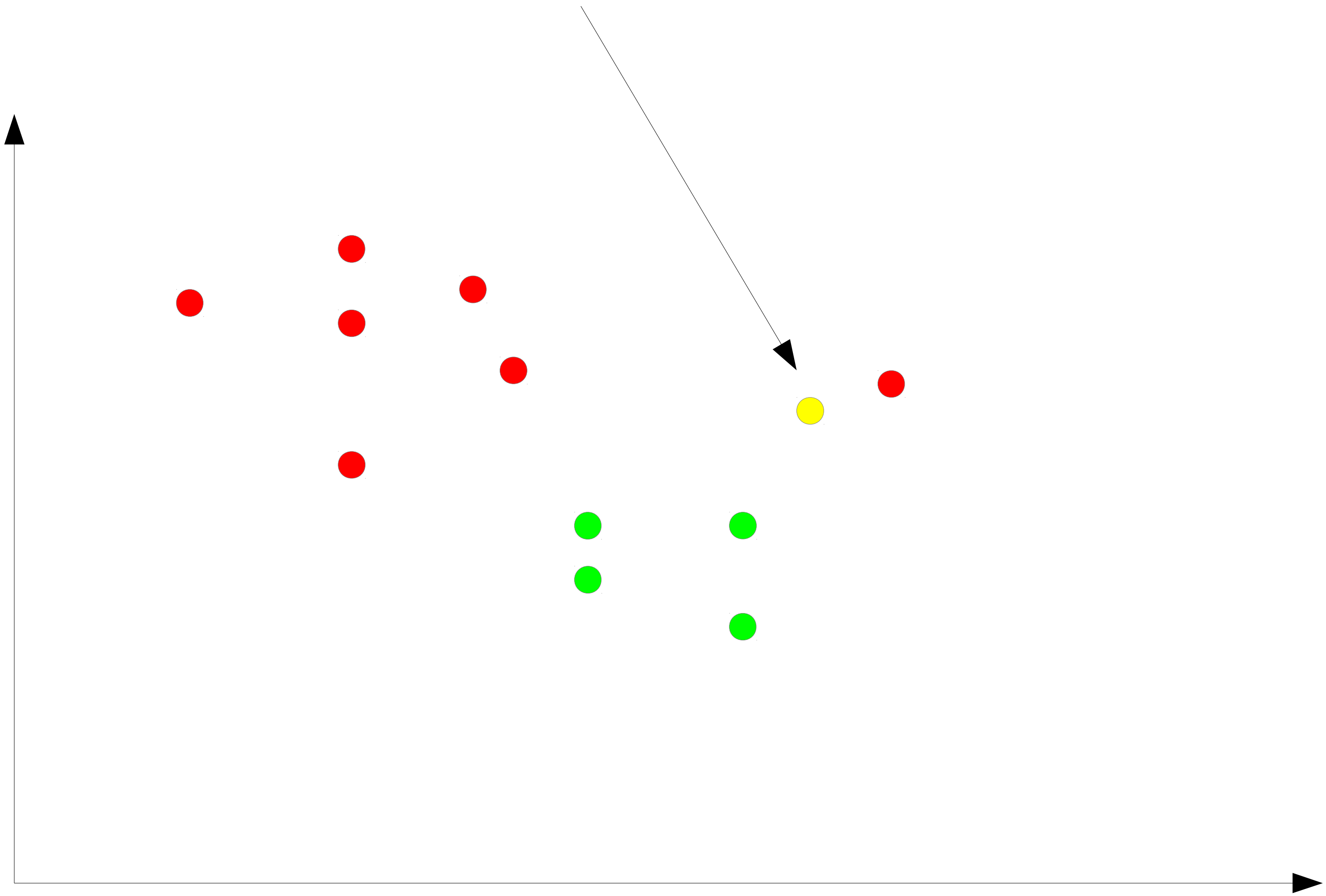
Vector Space Approaches

- Recall that all data are points in hyperspace



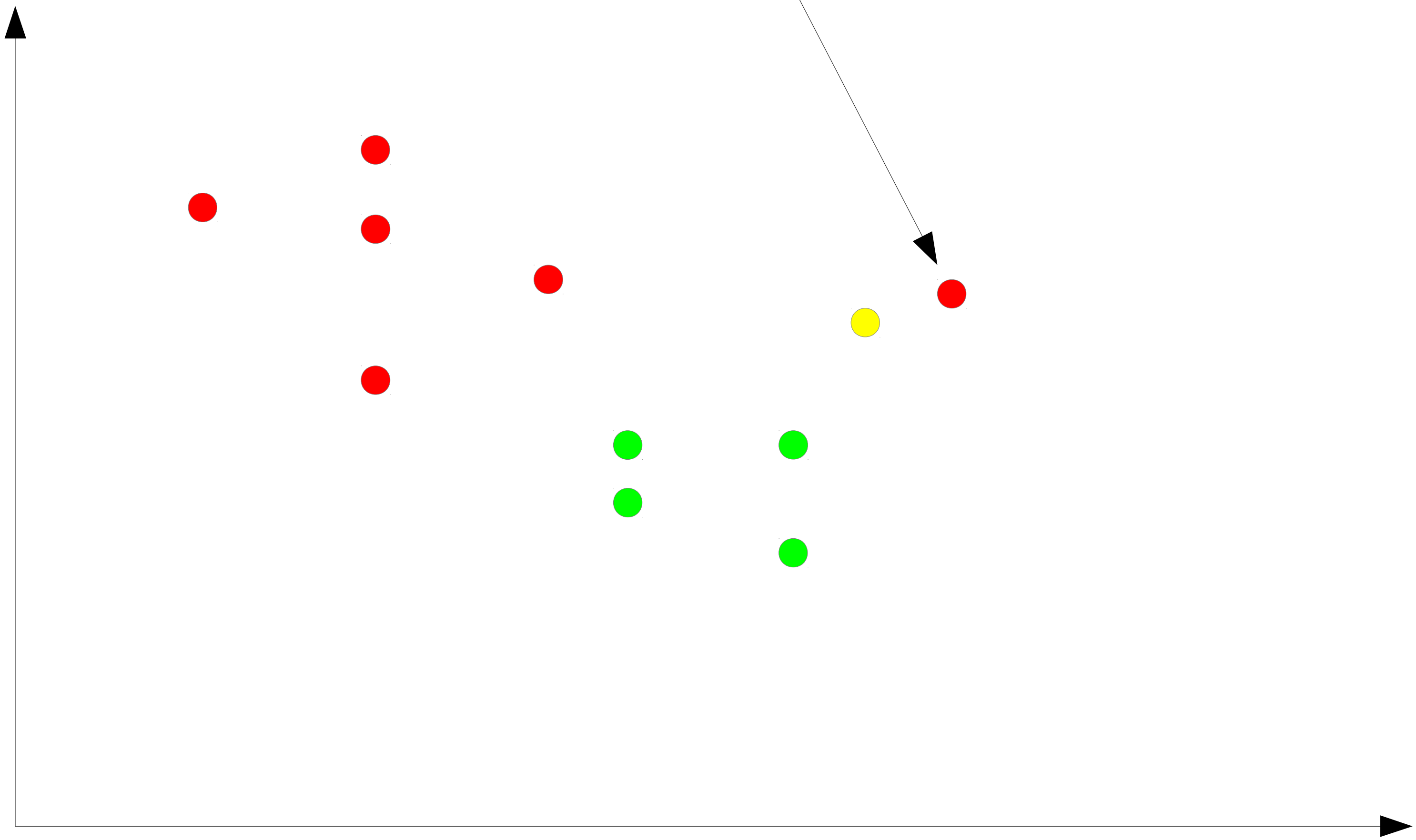
Vector Space Approaches

- How would you classify this point?



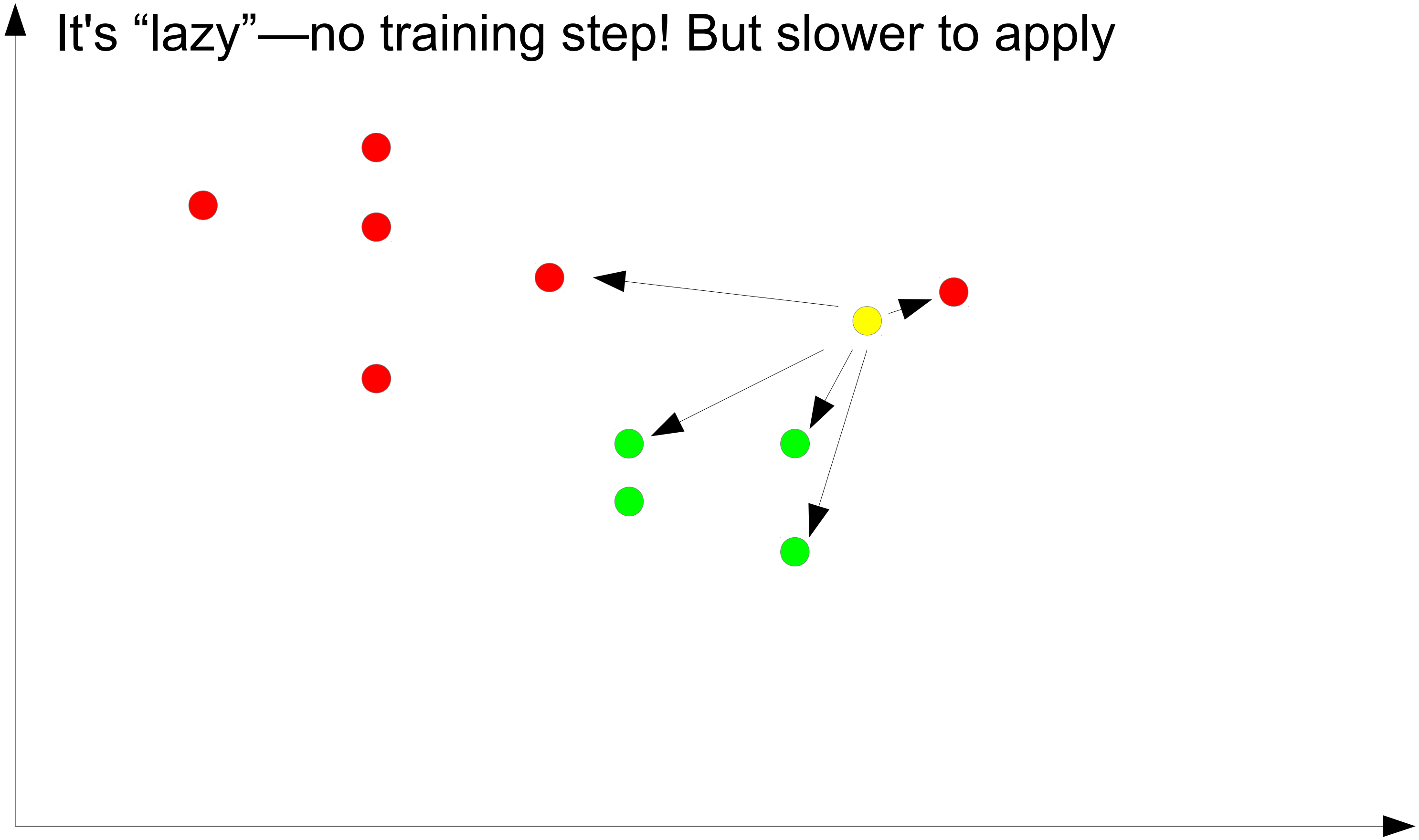
Vector Space Approaches

- It's nearest to the red one



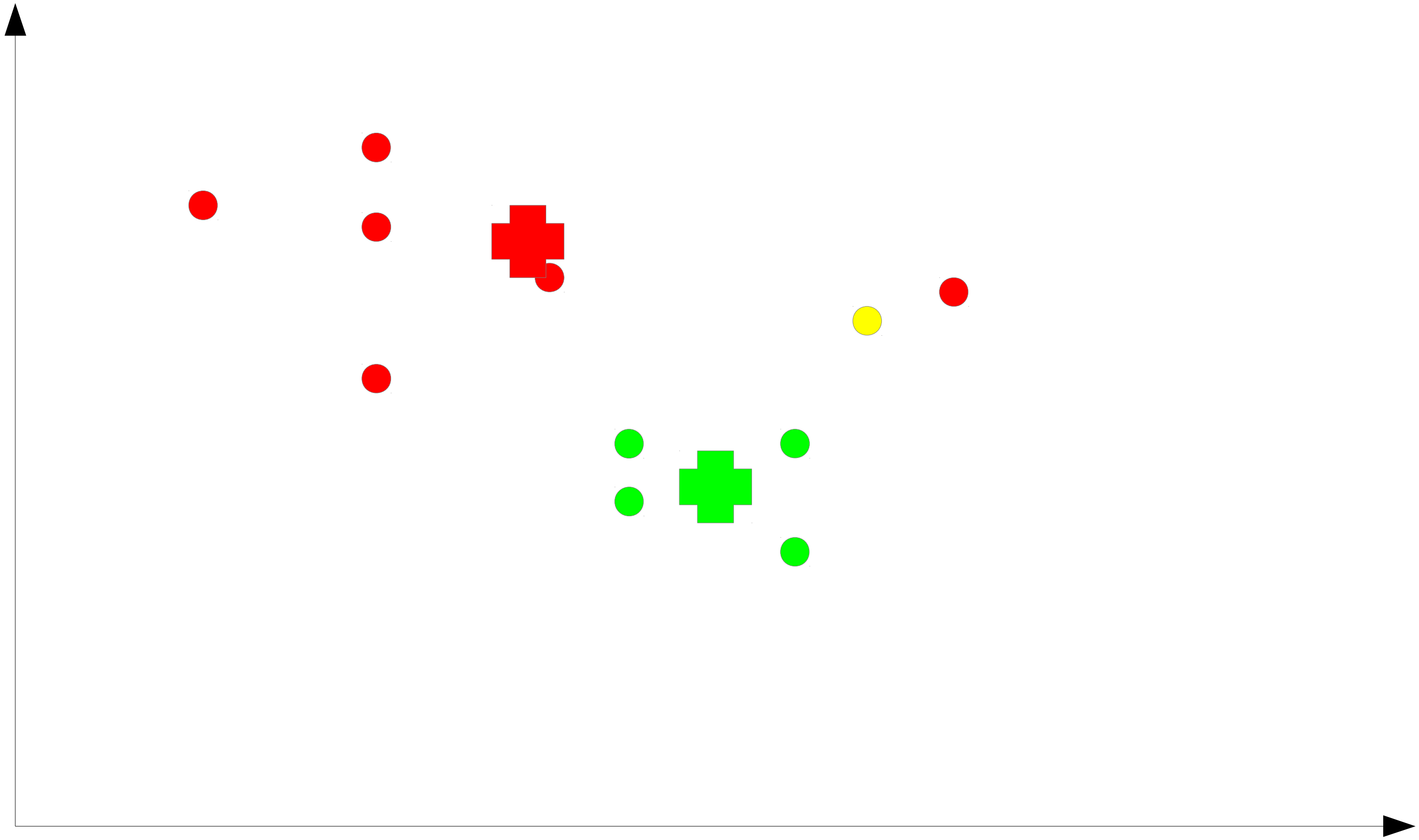
Vector Space Approaches

- But of its 5 nearest neighbours, three are green
- This is the k-nearest neighbours approach
- It's “lazy”—no training step! But slower to apply



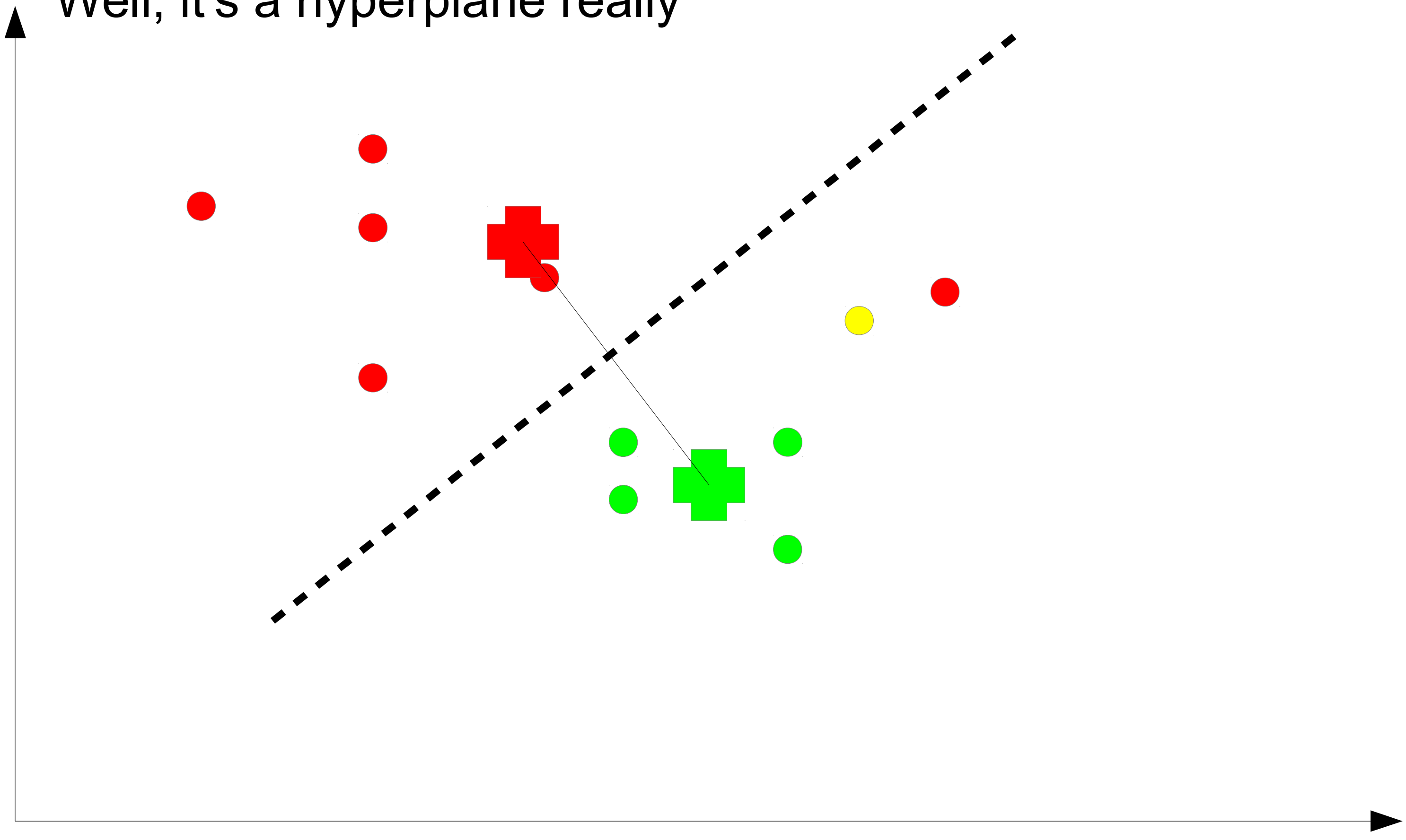
Vector Space Approaches

- Find centroids?



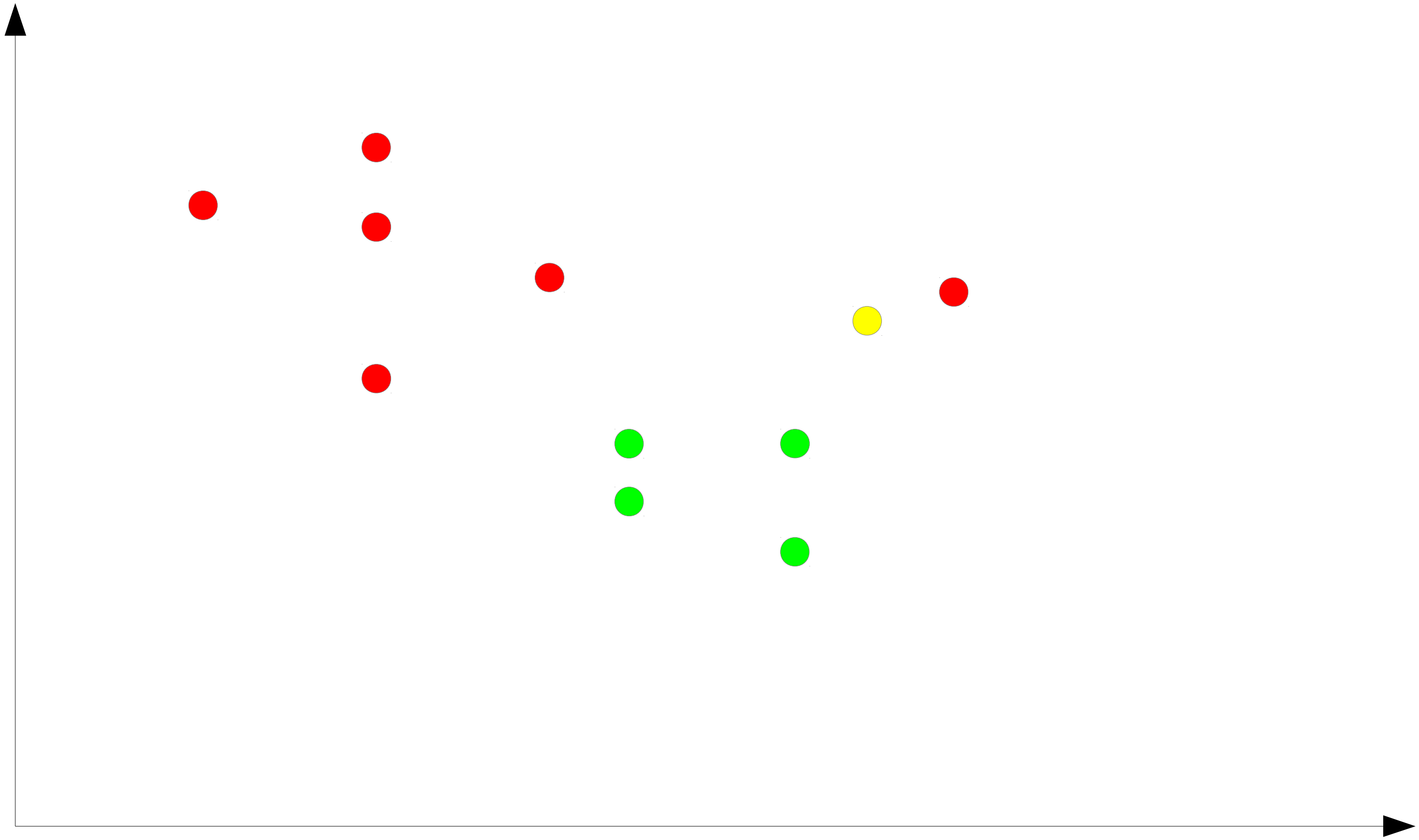
Vector Space Approaches

- This is equivalent to drawing a line across the space and using that to decide
- Well, it's a hyperplane really



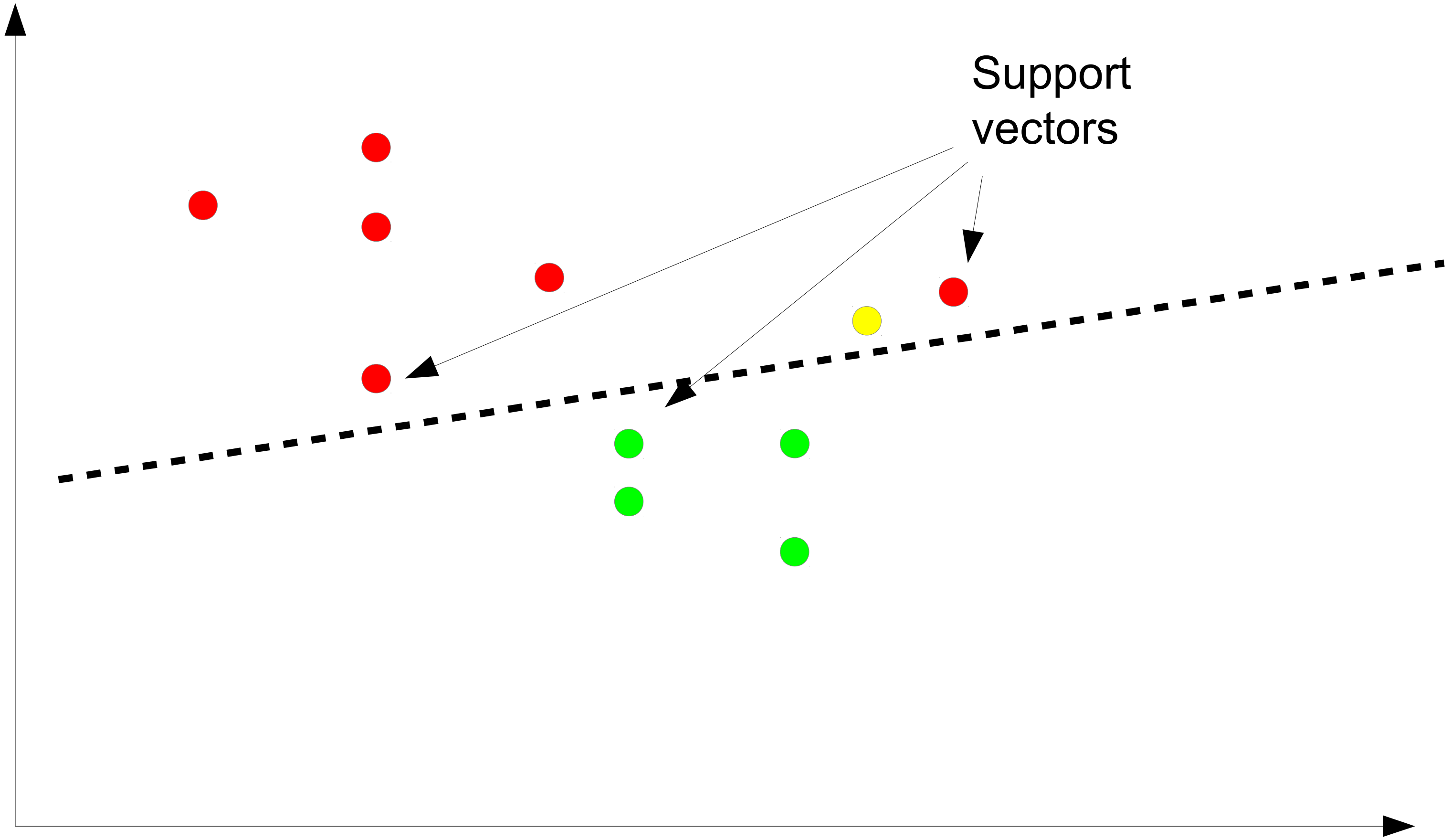
Vector Space Approaches

- How about focusing on the edges of the group rather than the middle?



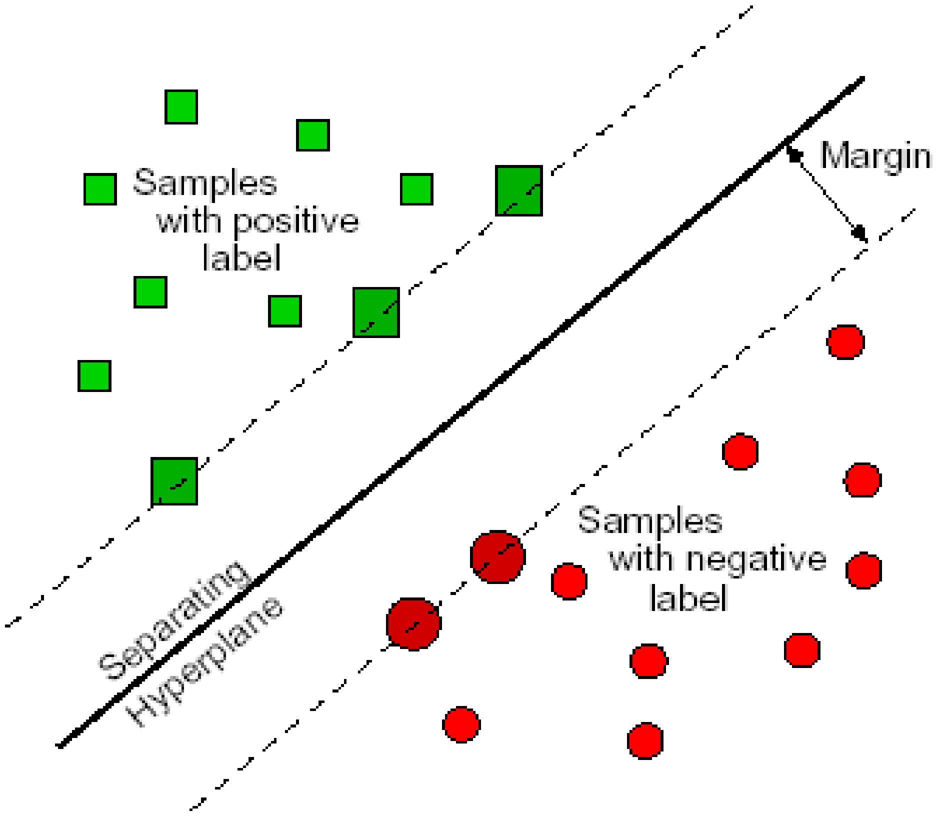
Vector Space Approaches

- Let's try to make the line as far away as possible from the ones at the edge
- That's a support vector machine!



Support Vector Machines

- Attempt to find a hyperplane that separates data
- Goal: maximize margin separating two classes
- Wider margin = greater generalisation



Support Vector Machines

- What if data doesn't split?
- Data may not split because it's noisy—a perfect solution isn't possible given available information
  - e.g. classifying people as male or female based on height isn't going to work
- Data may not split because a linear separator is unsuitable
  - e.g. classifying people as living or not living in Sheffield based on latitude and longitude of address won't work because Sheffield is a globule

Kernel Trick

- Map data into different dimensionality
- http://www.youtube.com/watch?v=3liCbRZPrZA
- As shown in the video, due to polynomial kernel elliptical separators can be created, not just straight lines.
- Different kernels lead to wide variety of complex separators

Kernel Trick in GATE and NLP

- Linear and polynomial kernels are implemented in Batch Learning PR's SVM
- Learning Framework PR includes all LibSVM kernels, including linear, polynomial and RBF
- However for many NLP problems a linear kernel is perfectly adequate, and complex kernels may overfit
  - Technically you could find a perfect solution that draws little rings round all your positive points, but it would not generalize!
- Cost parameter allows for misclassification and avoids overfitting

Summary

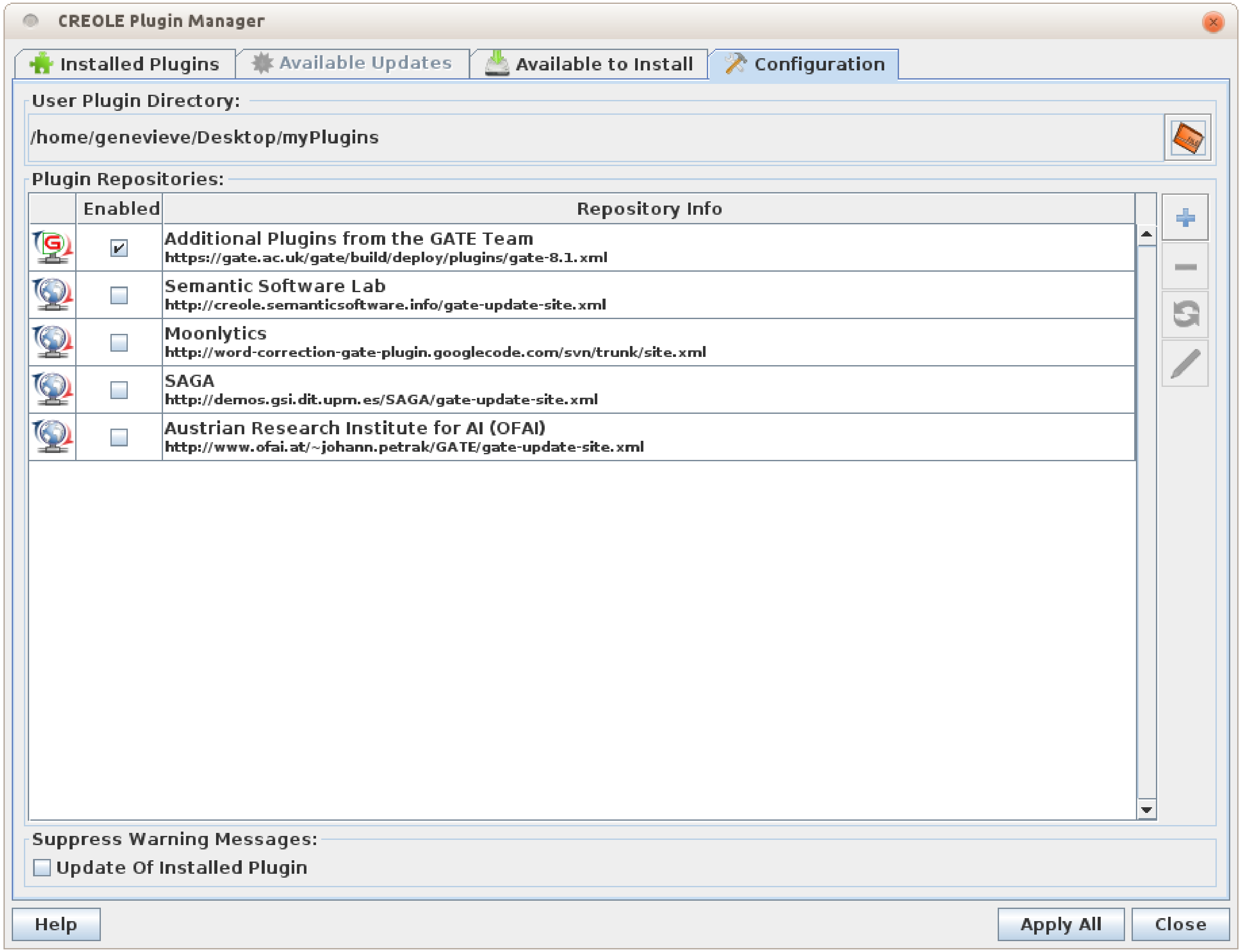
- We have seen a number of tasks, in addition to NER presented earlier in the day, that statistical techniques and ML can help with
- We have considered the potential inherent in distributional information about text in achieving different tasks
- We have learned to conceptualize data as points in high-dimensional space
- We have grasped the principles underlying a few common approaches and algorithms
- But getting a feel for what works best in different tasks comes from experience
- So now for some more hands-on!

Classification Exercise using Learning Framework Plugin

Classification tasks

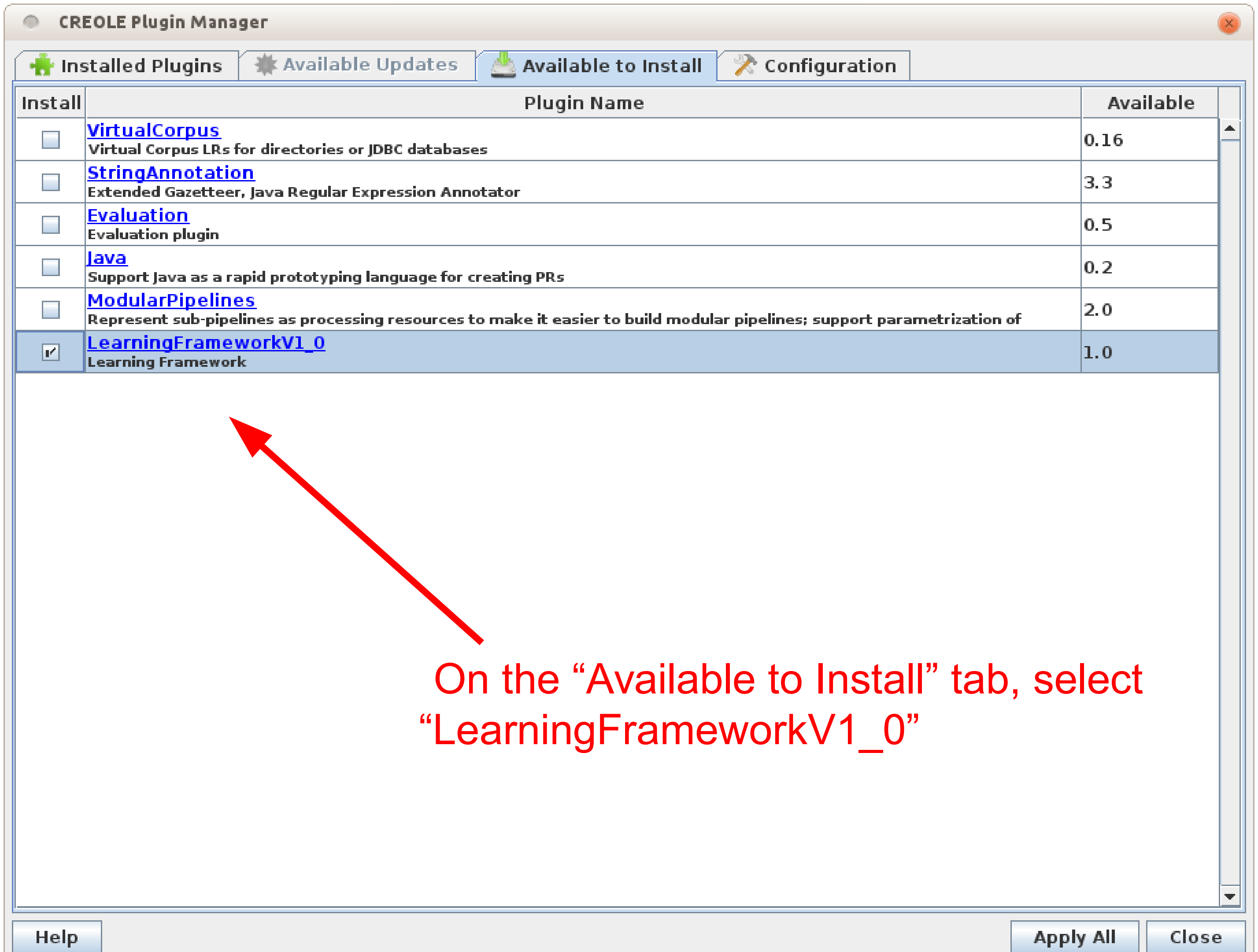
- Opinion mining
  - Example: the documents contain spans of text (such as individual sentences or longer consumer reviews) which you want to classify as positive, neutral, or negative
- Genre detection: classify each document or section as a type of news
- Author identification
- Today we will classify documents according to topic

Getting the Learning Framework Plugin



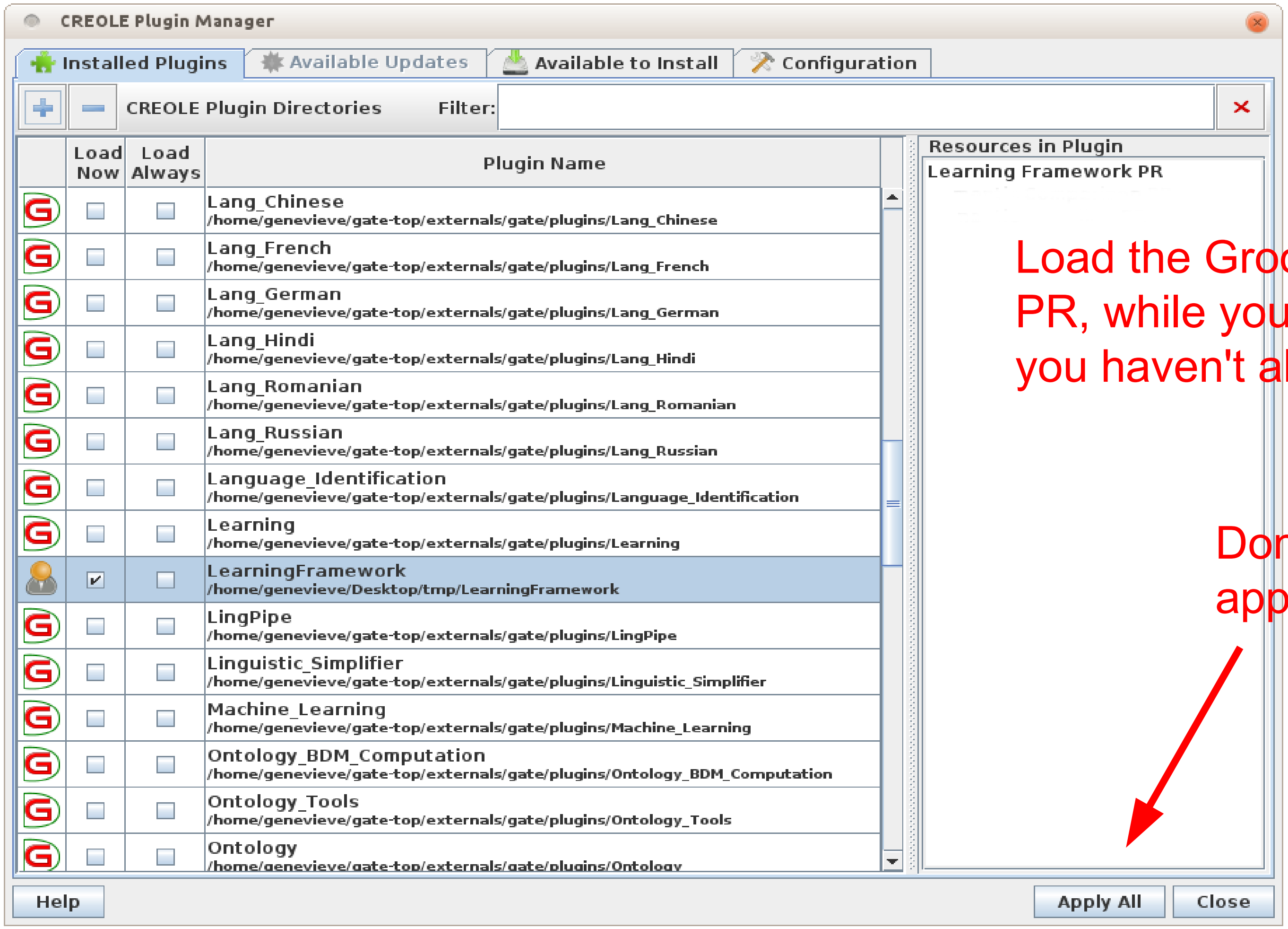
- Make a directory somewhere sensible for your plugins
- In the plugin manager, select your plugin directory as User Plugin Directory
- Enable “Additional Plugins from the GATE Team”

Getting the Learning Framework Plugin



On the “Available to Install” tab, select “LearningFrameworkV1_0”

Getting the Learning Framework Plugin



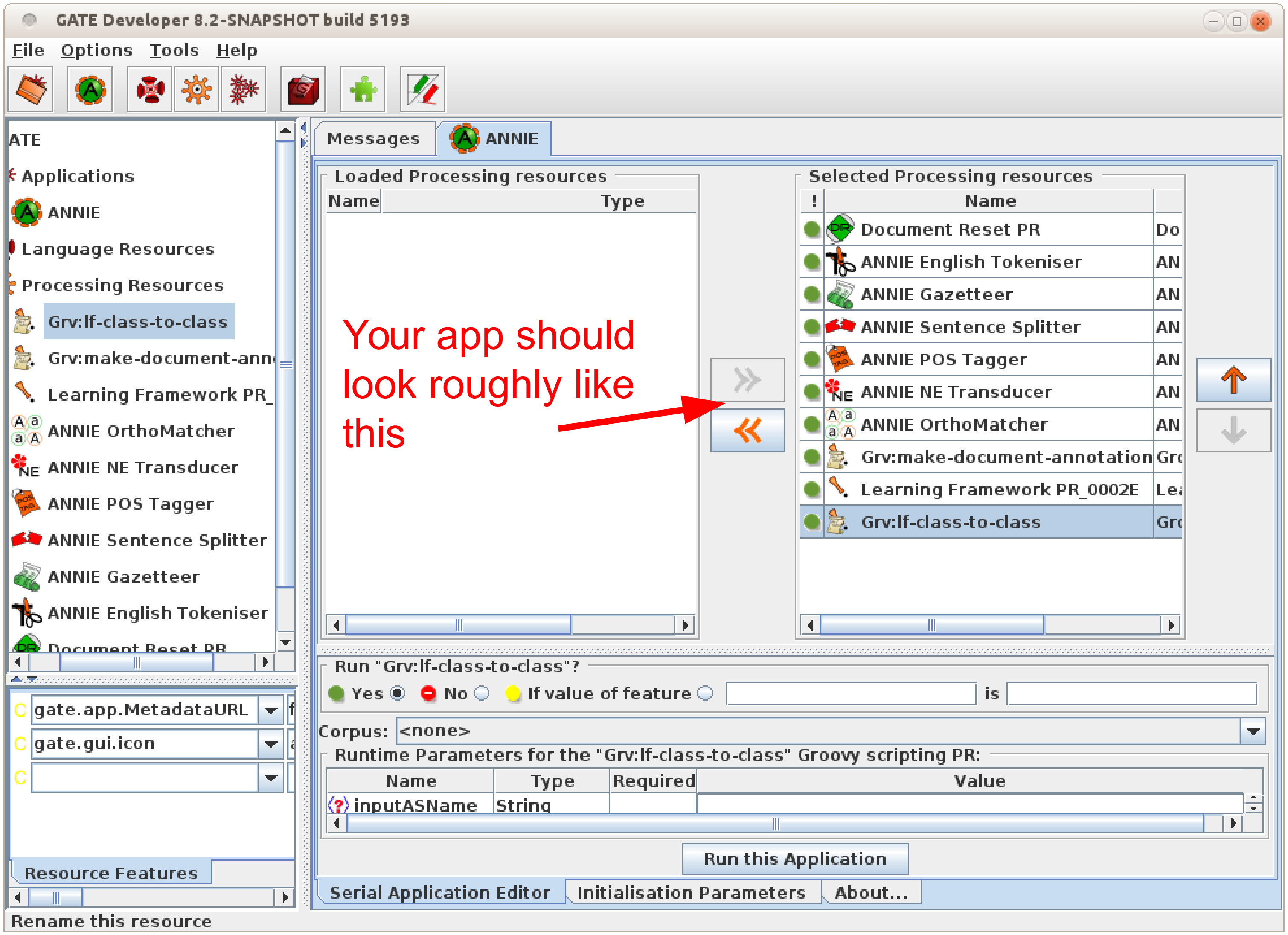
- Now you should be able to load the plugin
- If you can't, download this zip file from here and unzip into your user plugin directory:

https://github.com/GenevieveGorrell/gateplugin-LearningFramework/releases

Making the Application

- Load ANNIE with defaults
- Make an instance of the Learning Framework PR (you won't need any init time parameters)
- Make two Groovy Scripting PRs, with the following scripts as init time parameters:
  - hands-on/resources/make-document-annotation.groovy
  - hands-on/resources/lf_class_to_class.groovy

Making the Application



- Rename your Groovy scripts so you know which is which
- Add “make-document-annotation”
- Then add the Learning Framework PR
- Finally add “lf_class_to_class”

Load Corpora

- In the hands-on directory, you will find directories called “test” and “training”. These hold the test and training corpora.
- Make and populate these two corpora in GATE now.
- Documents contain some existing annotations, from previous annotation work:
    - Annotator sets on the document: alter the Document Reset PR to keep “ConsensusAuto”—it might be useful later

Overview of the Task

- The first Groovy script is going to put the “class” feature from the document into a “Document” annotation in the default annotation set that spans the whole document content
- We're learning to classify these “Document” annotations, so our learning instance annotation type is “Document”
- The feature containing the classification will be “class”
- Have a look at a few documents and see what class values there are. Do you think it will be an easy task to learn?
- We will use annotations on the document to provide attributes to learn from

Using the Learning Framework PR

- Unlike the Batch Learning PR, the Learning Framework PR takes the config file as a runtime parameter
- The config file only specifies features—everything else is specified as a runtime parameter
- That's a lot of runtime parameters!
- It makes it easier to try different things in the GUI though.

Learning Framework Runtime Params

- Set some parameters! I have highlighted the important ones for now
- Learning Framework PR can get quite annoyed if you don't set the feature spec URL and the save directory!

Runtime Parameters for the "Learning Framework PR_0002E" Learning Framework PR:

Name	Type	Required	Value
classFeature	String		class
classType	String	✓	Document
confidenceThreshold	Double	✓	0.0
featureSpecURL	URL	✓	/hands-on/resources/feature-spec.xml
foldsForXVal	Integer		3
identifierFeature	String		
inputASName	String		
instanceName	String	✓	Document
learnerParams	String		
mode	Mode	✓	CLASSIFICATION
operation	Operation	✓	TRAIN
outputASName	String		LearningFramework
saveDirectory	URL	✓	/hands-on/resources/
sequenceSpan	String		
trainingAlgo	Algorithm		<none>
trainingProportion	Float		0.5

Learning Framework Runtime Params

- Once we're set up and ready to work, the important parameters are mode, operation and trainingAlgo

Runtime Parameters for the "Learning Framework PR_0002E" Learning Framework PR:

Name	Type	Required	Value
classFeature	String		class
classType	String	✓	Document
confidenceThreshold	Double	✓	0.0
featureSpecURL	URL	✓	/hands-on/resources/feature-spec.xml
foldsForXVal	Integer		3
identifierFeature	String		
inputASName	String		
instanceName	String	✓	Document
learnerParams	String		
mode	Mode	✓	CLASSIFICATION
operation	Operation	✓	TRAIN
outputASName	String		LearningFramework
saveDirectory	URL	✓	/hands-on/resources/
sequenceSpan	String		
trainingAlgo	Algorithm		<none>
trainingProportion	Float		0.5

Understanding modes

- The mode parameter can be set to CLASSIFICATION or NAMED_ENTITY_RECOGNITION
- The NAMED_ENTITY_RECOGNITION allows you to do chunking tasks
- We're going to use CLASSIFICATION for this task though

Algorithms

- Under trainingAlgo you can see a number of helpfully named algorithms are available
- Three libraries are integrated
- Names begin with the library they are from
- After that, “CL” indicates that it's a classification algorithm and “SEQ” indicates a sequence learner
  - Sequence learners need a span specified, and are good for NER tasks
- After that, a “NUM” indicates that this learner learns a continuous numeric class
  - Our task is to learn a nominal class

Understanding operations

- There are quite a few operations, but the important ones are TRAIN and APPLY_CURRENT_MODEL
- TRAIN trains a new model and places it in the save directory
- APPLY_CURRENT_MODEL applies whatever model it finds in the save directory
  - It doesn't train first, so it doesn't care what the parameters say or whether you changed the feature spec
- EXPORT_ARFF exports the data in ARFF format, allowing you to explore it in Weka or check it looks how you expect

More operations—Evaluation

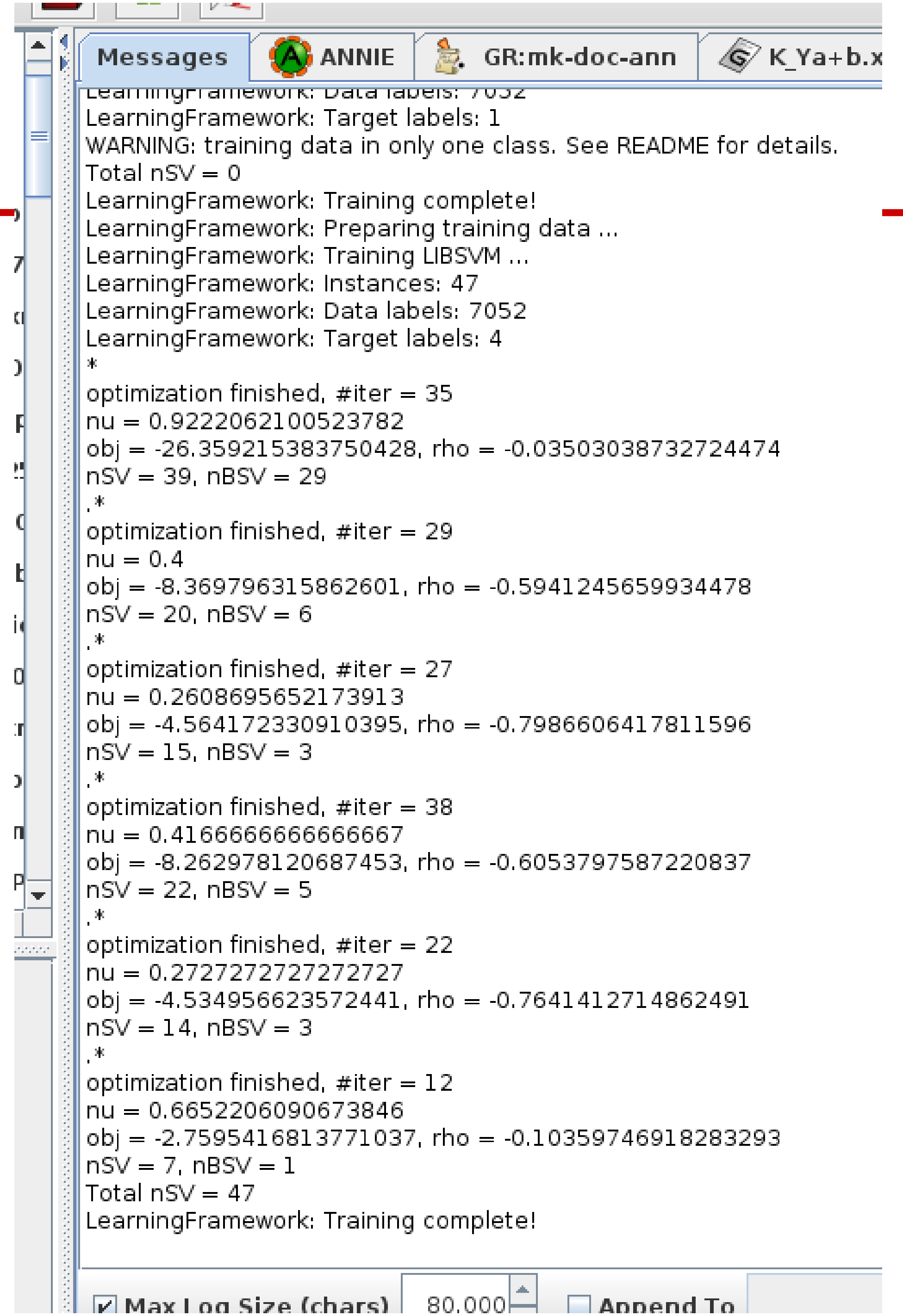
- Two evaluation modes are provided; EVALUATE_X_FOLD and EVALUATE_HOLDOUT
- These wrap the evaluation implementation provided by the machine learning library for that algorithm
- They only provide a classification evaluation, not an NER evaluation
    - Recall that the Batch Learning PR provides an NER evaluation

Export the data as ARFF

- Choose EXPORT_ARFF as mode, and run over one of the corpora
- In the resources directory, you should now see a directory called exportedCorpora
- It contains a file called output.arff
- Examine it now
- At the top are a list of attributes. Are they as expected?
- The last attribute is the class attribute. Do you see it?
- After that come feature vectors in sparse format. How can you tell that they are in sparse format? What would this file look like if they were written out in full?

Training

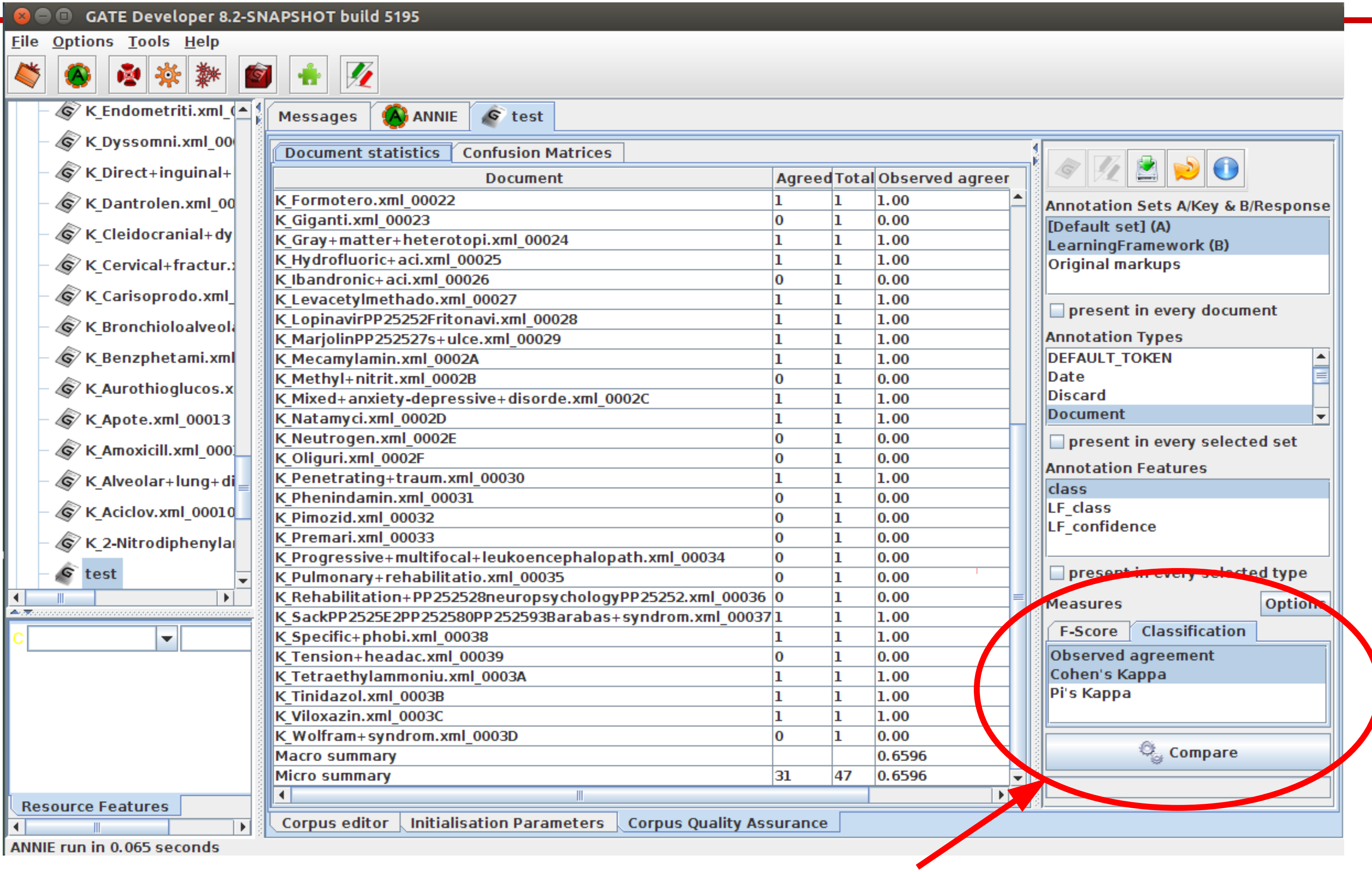
- Now select the TRAIN operation
- Try LIBSVM algorithm
  - Set learnerParams to “-b 1” because due to bug, it won't apply unless it is probabilistic! Oops!
- Remember to choose the training corpus!
- Do you get something like this?
- Is the number of instances as expected? Data labels (what are they?) Target labels?



Application

- Change to application mode
- Don't forget to use the test corpus!
- Run it!
- Did it run without error? Check what annotations have appeared in the LearningFramework annotation set

Corpus QA for Classification Tasks



Classification measures

Classification Evaluation

- For NER, there are more ways to be wrong
  - Fewer or more mentions than there really are, or you can overlap
- For classification, each response is simply right or wrong
- Therefore, “accuracy” presents the proportion of answers that were right
- But … what if your corpus is imbalanced? What if 90% of your corpus is sheep and 10% is goats?
- Kappa statistics calculate how improbable it is that your result is statistically independent of the actual

Confusion Matrices

- Another way to figure out how clever your model really is
- What do you notice about the misclassifications?

Messages | ANNIE | test | K_Dantrolen.xml...

Document statistics | Confusion Matrices

Whole corpus				
A \ B	condition	other	substance	treatment
condition	16	0	5	0
other	0	0	1	0
substance	8	0	15	0
treatment	1	0	1	0
K_2-Nitrodiphenylami...				
A \ B	substance			
substance	1			
K_Aciclov.xml_00010				

Exercises—Improving the Result

- Now see if you can improve your result
- Suggestions:
  - Try different algorithms
  - Look up the LibSVM parameters online and see if anything looks worth trying
    - Hint: try a higher cost!

Parameter Tuning in Weka—Demo

- Parameter tuning can be faster outside of GATE
- Some algorithms work better if you select features carefully
- Even if the performance doesn't degrade with extra features, it is faster with as few as possible
- Weka can help
  - So long as you use algorithms that are integrated in the PR, and use the same parameters, the result should transfer