
Chunking—Practical Exercise

Chunking for NER

- Chunking, as we saw at the beginning, means finding parts of text
- This task is often called Named Entity Recognition (NER), in the context of finding person and organization names
- The same principle can be applied to any task that involves finding where things are located in text
 - For example, finding the noun phrases
 - Can you think of any others?

California Governor Arnold Schwarzenegger proposes deep cuts.

Person

Chunking for NER

- It's implemented as a twist on classification (everything is classification under the hood!)
- We achieve this in the Learning Framework by identifying which tokens are the beginning of a mention, which are the insides and which are the outsides (“BIO”)
 - There are other schemes; the old Batch Learning PR used BE (beginnings and ends)
- You don't need to worry about the Bs, Is and Os; the Learning Framework will take care of all that for you! You just need a corpus annotated with entities

California Governor Arnold Schwarzenegger proposes deep cuts.





Chunking—Practical Exercise

- Materials for this exercise are in the folder called “chunking-hands-on”
- You might want to start by closing any applications and corpora from the previous exercise, so we have a fresh start

Finding Person Mentions using Chunking Training and Application PRs



Load the corpus

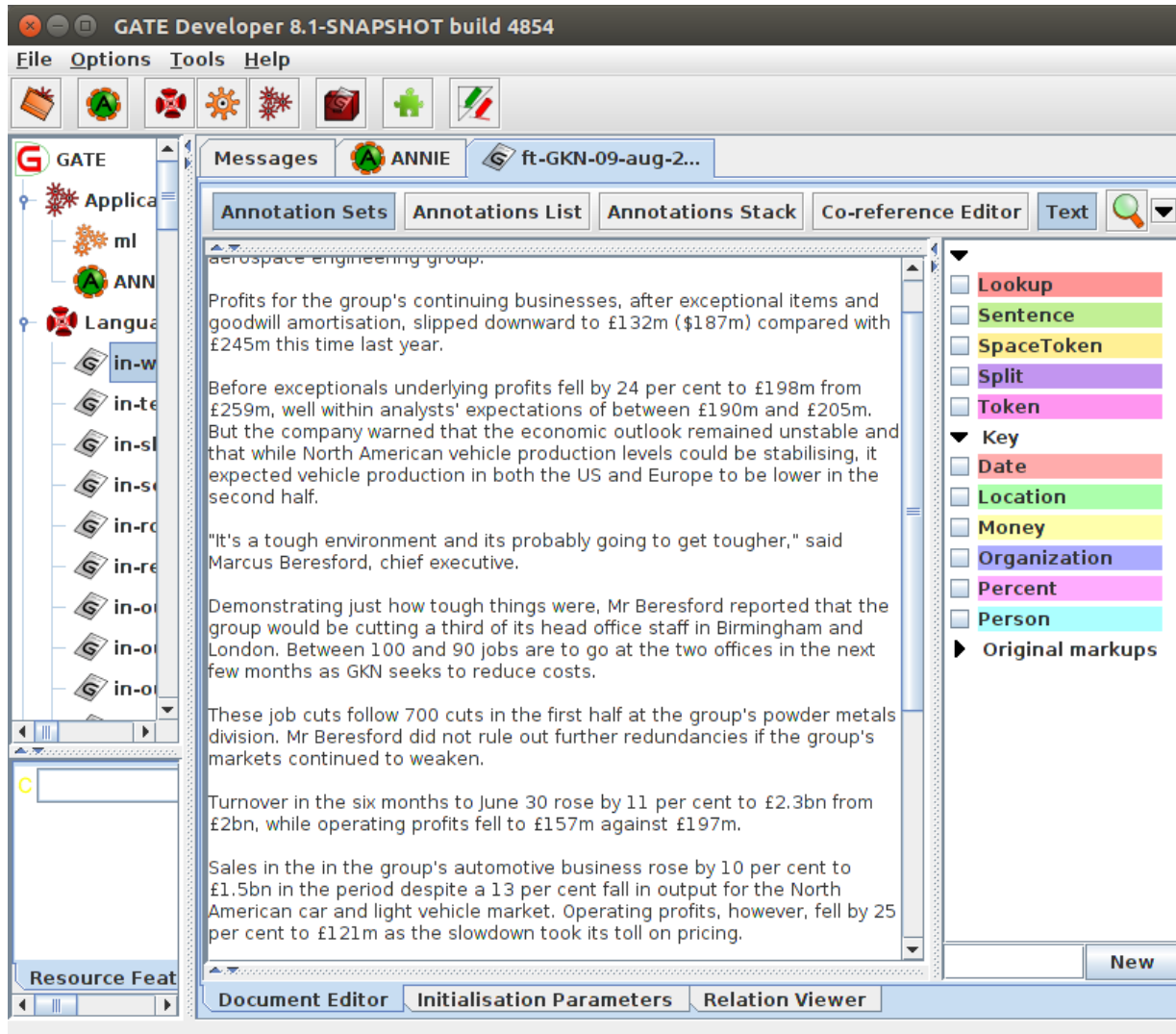
- Create corpora for training and testing, with sensible names
- Populate them from the training and testing corpora you have in your chunking hands on materials
- Open a document and examine its annotations



Examining the corpus

- The corpus contains an annotation set called “Key”, which has been manually prepared
- Within this annotation set are annotations of types “Date”, “Location”, “Money”, “Organization” and so forth

Creating the application



- As previously, if we run ANNIE on the corpus, we have more annotations to work with
- So start by loading ANNIE as the basis for your application
- Again, we don't need the NE transducer or orthomatcher

NER GATE application



GATE Developer 8.2-SNAPSHOT build 5490

File Options Tools Help

GATE

- Applications
 - ANNIE
 - Language Resources
 - Processing Resources
 - Annotation Set Transfer
 - LF_ApplyChunking 00031
 - LF_TrainChunking 00030
 - ANNIE OrthoMatcher
 - ANNIE NE Transducer
 - ANNIE POS Tagger
 - ANNIE Sentence Splitter
 - ANNIE Gazetteer
 - ANNIE English Tokeniser
 - Document Reset PR
 - Datastores

Messages ANNIE

Loaded Processing resources

Name	Type
ANNIE NE Transducer	ANNIE NE Transducer
ANNIE OrthoMatcher	ANNIE OrthoMatcher
LF_ApplyChunking 00031	LF_ApplyChunking

Selected Processing resources

Name	Type
Document Reset PR	Document Reset PR
ANNIE English Tokeniser	ANNIE English Tokeniser
ANNIE Gazetteer	ANNIE Gazetteer
ANNIE Sentence Splitter	ANNIE Sentence Splitter
ANNIE POS Tagger	ANNIE POS Tagger
Annotation Set Transfer 00036	Annotation Set Transfer
LF_TrainChunking 00030	LF_TrainChunking

Run "Annotation Set Transfer 00036"?

Yes No If value of feature is

Corpus: <none>

Runtime Parameters for the "Annotation Set Transfer 00036" Annotation Set Transfer:

Name	Type	Required	Value
annotationTypes	ArrayList		[]
copyAnnotations	Boolean	✓	false
inputASName	String		
outputASName	String		
tagASName	String		Original markups
textTagName	String		

Run this Application

Serial Application Editor Initialisation Parameters About...

Annotation Set Transfer 00036 loaded in 0.001 seconds

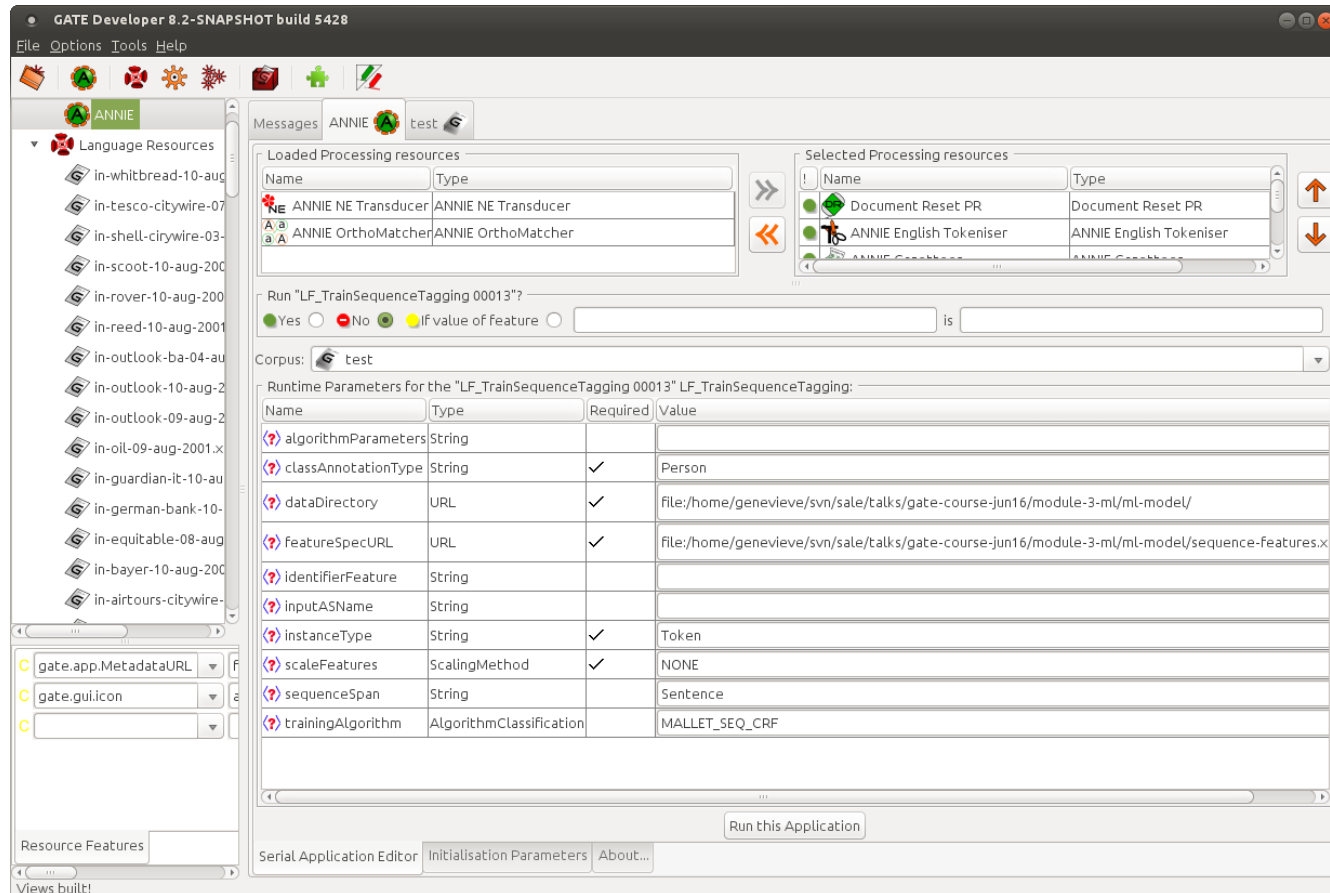
- Again, we need an Annotation Set Transfer, so create and add one
- Then create both training and application chunking PRs
- Start by just adding the training one



Annotation Set Transfer

- We'll use the annotation set transfer to copy the Person annotations up to the default annotation set, where we can learn them
- **Go ahead and set up your AST now**
- Be sure to copy them, not move them!

Chunking training parameters



Messages ANNIE test

Loaded Processing resources

Name	Type
ANNIE NE Transducer	ANNIE NE Transducer
ANNIE OrthoMatcher	ANNIE OrthoMatcher

Selected Processing resources

Name	Type
Document Reset PR	Document Reset PR
ANNIE English Tokeniser	ANNIE English Tokeniser

Run "LF_TrainSequenceTagging 00013"?

Yes No If value of feature is

Corpus: test

Runtime Parameters for the "LF_TrainSequenceTagging 00013" LF_TrainSequenceTagging:

Name	Type	Required	Value
algorithmParameters	String		
classAnnotationType	String	✓	Person
dataDirectory	URL	✓	file:/home/genevieve/svn/sale/talks/gate-course-jun16/module-3-ml/ml-model/
featureSpecURL	URL	✓	file:/home/genevieve/svn/sale/talks/gate-course-jun16/module-3-ml/ml-model/sequence-features.x
identifierFeature	String		
inputASName	String		
instanceType	String	✓	Token
scaleFeatures	ScalingMethod	✓	NONE
sequenceSpan	String		Sentence
trainingAlgorithm	AlgorithmClassification		MALLET_SEQ_CRF

Run this Application

Serial Application Editor Initialisation Parameters About...

- Let's look at the parameters for the training PR
- Instead of targetFeature, we have classAnnotationType

Chunking training parameters

- For classification, the class to learn is in a feature on the instance, is specified to the PR in the targetFeature parameter
- For chunking, the class to learn takes the form of an annotation type. In our case, our corpus is annotated with Person annotations that we are going to learn to locate
- This type to learn is indicated in the classAnnotationType parameter



Chunking training parameters

- Set the `classAnnotationType` now
- Set the `dataDirectory` to where you want to save your model, and set the `featureSpecURL` (there's a feature spec to get you started in the hands on materials)
- Set `instanceType`. What do you think it should be?

Sequence Spans

- sequenceSpan is only relevant when using sequence learners
- Sequence learners classify each instance in the span by making use of the others
- For example, a noun phrase might be more likely to follow a determiner than a preposition, or a person name might be more likely to follow the word “Mrs”
- The Learning Framework offers the Conditional Random Fields sequence learner
- It might be good for finding Persons, so let's use it!
 - You don't have to use a sequence learner for chunking though
- What do you think would be a good sequence span?

Sequence Spans

- Sequence spans should be spans within which instance classes follow patterns
 - For example, grammatical rules apply to sequences of parts of speech
 - However, sentiment classifications of individual customer reviews don't form a meaningful sequence
- A sequence span shouldn't be longer than necessary
- Sentence would be a good span for our task
- Fortunately, ANNIE creates sentence annotations for us, so those are available to use
- **Set sequenceSpan to "Sentence"**

Feature Specification

```
<ML-CONFIG>
```

```
<ATTRIBUTE>
<TYPE>Token</TYPE>
<FEATURE>category</FEATURE>
<DATATYPE>nominal</DATATYPE>
</ATTRIBUTE>
```

```
<ATTRIBUTE>
<TYPE>Token</TYPE>
<FEATURE>kind</FEATURE>
<DATATYPE>nominal</DATATYPE>
</ATTRIBUTE>
```

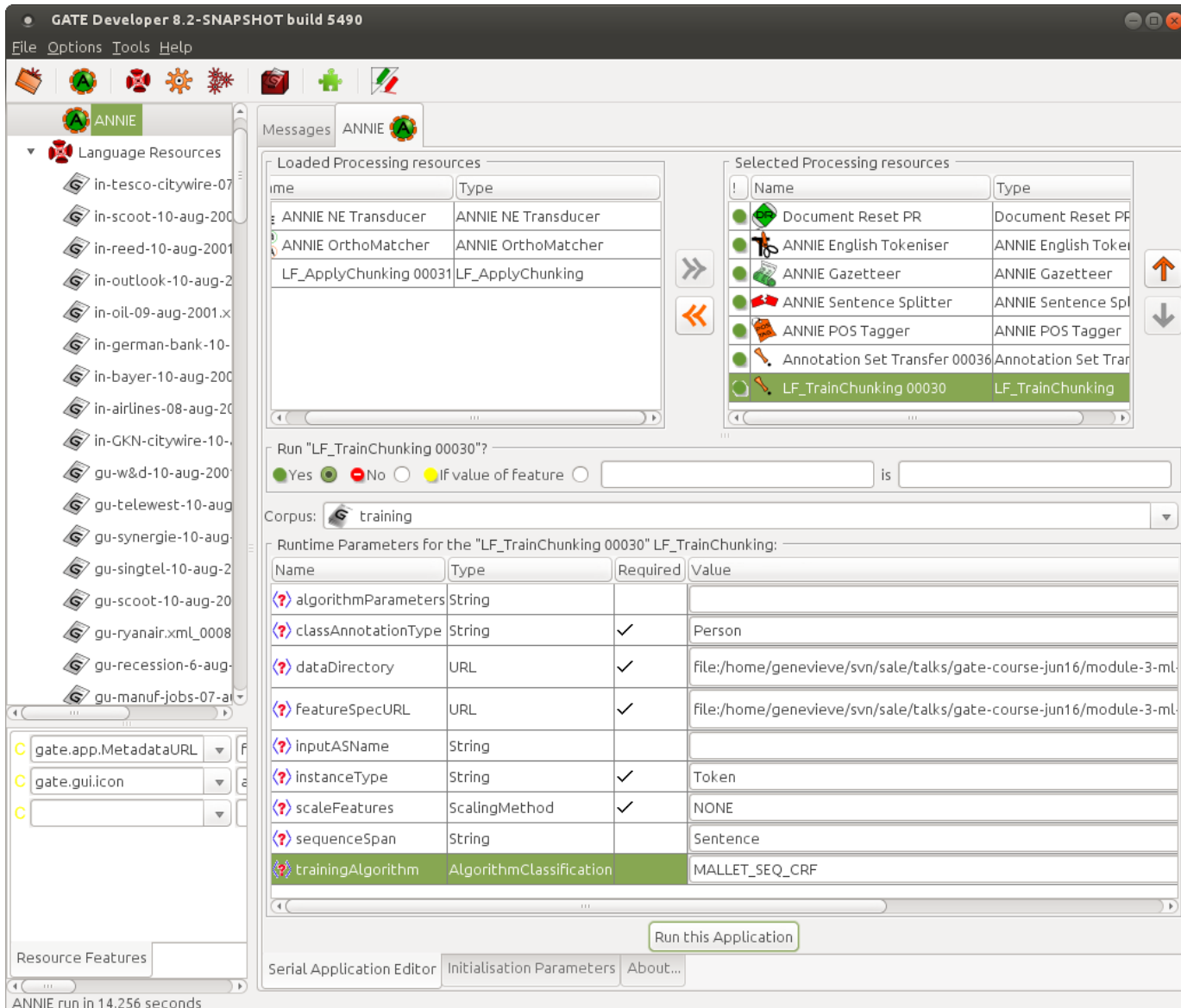
```
<ATTRIBUTE>
<TYPE>Token</TYPE>
<FEATURE>length</FEATURE>
<DATATYPE>numeric</DATATYPE>
</ATTRIBUTE>
```

```
<ATTRIBUTE>
<TYPE>Token</TYPE>
<FEATURE>orth</FEATURE>
<DATATYPE>nominal</DATATYPE>
</ATTRIBUTE>
```

```
<ATTRIBUTE>
<TYPE>Token</TYPE>
<FEATURE>string</FEATURE>
<DATATYPE>nominal</DATATYPE>
</ATTRIBUTE>
```

```
</ML-CONFIG>
```

- For this task, we are using attribute features
- These allow us to take features from the instance annotations or others that are co-located with them
- We specify type, feature and datatype
- Attribute features also can be taken from instances nearby
- That's a bit less useful with a sequence learner though—why?



The screenshot shows the GATE Developer interface with the following configuration:

- Language Resources:** A list of various corpora is shown on the left, including 'in-tesco-citywire-07', 'in-scoot-10-aug-200', 'in-reed-10-aug-2001', 'in-outlook-10-aug-2', 'in-oil-09-aug-2001.x', 'in-german-bank-10-', 'in-bayer-10-aug-200', 'in-airlines-08-aug-20', 'in-GKN-citywire-10-', 'gu-w&d-10-aug-200', 'gu-telewest-10-aug', 'gu-synergie-10-aug', 'gu-singtel-10-aug-2', 'gu-scoot-10-aug-20', 'gu-ryanair.xml_0008', 'gu-recession-6-aug', and 'gu-manuf-jobs-07-a'.
- Loaded Processing resources:**

Name	Type
ANNIE NE Transducer	ANNIE NE Transducer
ANNIE OrthoMatcher	ANNIE OrthoMatcher
LF_ApplyChunking 00031	LF_ApplyChunking
- Selected Processing resources:**

Name	Type
Document Reset PR	Document Reset PR
ANNIE English Tokeniser	ANNIE English Tokeniser
ANNIE Gazetteer	ANNIE Gazetteer
ANNIE Sentence Splitter	ANNIE Sentence Splitter
ANNIE POS Tagger	ANNIE POS Tagger
Annotation Set Transfer 00036	Annotation Set Transfer
LF_TrainChunking 00030	LF_TrainChunking
- Run "LF_TrainChunking 00030"?:**
 - Yes
 - No
 - If value of feature is
- Corpus:** training
- Runtime Parameters for the "LF_TrainChunking 00030" LF_TrainChunking:**

Name	Type	Required	Value
algorithmParameters	String		
classAnnotationType	String	✓	Person
dataDirectory	URL	✓	file:/home/genevieve/svn/sale/talks/gate-course-jun16/module-3-ml
featureSpecURL	URL	✓	file:/home/genevieve/svn/sale/talks/gate-course-jun16/module-3-ml
inputASName	String		
instanceType	String	✓	Token
scaleFeatures	ScalingMethod	✓	NONE
sequenceSpan	String		Sentence
trainingAlgorithm	AlgorithmClassification		MALLET_SEQ_CRF
- Buttons:** Run this Application, Serial Application Editor, Initialisation Parameters, About...
- Status:** ANNIE run in 14.256 seconds

- Make sure you have selected the training corpus
- Run the application!

Chunking application parameters

- Now move the training PR out of the application and add the application PR
- You can take the annotation set transfer out too
- It doesn't have a targetFeature parameter like the classification application PR did
- You don't need to tell it what type to create because the model knows it from training!
- Set dataDirectory to the location where you told the training PR to put the model
- Set the sequence span



Applying

The screenshot shows the GATE Developer interface with the following components:

- Left Panel:** A tree view of Language Resources including files like 'in-tesco-citywire-07', 'in-scoot-10-aug-200...', 'in-reed-10-aug-2001...', 'in-outlook-10-aug-2...', 'in-oil-09-aug-2001.x', 'in-german-bank-10-', 'in-bayer-10-aug-200...', 'in-airlines-08-aug-20...', 'in-GKN-citywire-10-', 'gu-w&d-10-aug-200...', 'gu-telewest-10-aug-', 'gu-synergie-10-aug-', 'gu-singtel-10-aug-2...', 'gu-scoot-10-aug-20...', and 'gu-ryanair.xml_0008'.
- Messages Panel:** Shows 'Loaded Processing resources' and 'Selected Processing resources'.

Name	Type
ANNIE NE Transducer	ANNIE NE Transducer
ANNIE OrthoMatcher	ANNIE OrthoMatcher
Annotation Set Transfer 00036	Annotation Set Trans
LF_TrainChunking 00030	LF_TrainChunking

Name	Type
Document Reset PR	Document Reset PR
ANNIE English Tokeniser	ANNIE English Tokeniser
ANNIE Gazetteer	ANNIE Gazetteer
ANNIE Sentence Splitter	ANNIE Sentence Splitter
ANNIE POS Tagger	ANNIE POS Tagger
LF_ApplyChunking 00031	LF_ApplyChunking
- Configuration Panel:** A dialog box titled 'Run "LF_ApplyChunking 00031"?' with radio buttons for 'Yes', 'No', and 'If value of feature'. Below it, a 'Corpus:' dropdown is set to 'training'.

Name	Type	Required	Value
algorithmParameters	String		
confidenceThreshold	Double	✓	0.0
dataDirectory	URL	✓	file:/home/genevieve/svn/sale/talks/gate-course-jun16/module-3-ml-barbour/chunkir
inputASName	String		
instanceType	String	✓	Token
outputASName	String		LearningFramework
sequenceSpan	String		Sentence
- Bottom Panel:** Includes a 'Run this Application' button and tabs for 'Serial Application Editor', 'Initialisation Parameters', and 'About...'. A status bar at the bottom left indicates 'ANNIE run in 14.256 seconds'.

- Now run this on the test corpus

Chunking—Evaluation using Corpus QA

Chunking Evaluation

- For classification, each response is simply right or wrong
- For NER, there are more ways to be wrong
 - Fewer or more mentions than there really are, or you can overlap
- So we need different metrics

What are precision, recall and F1?

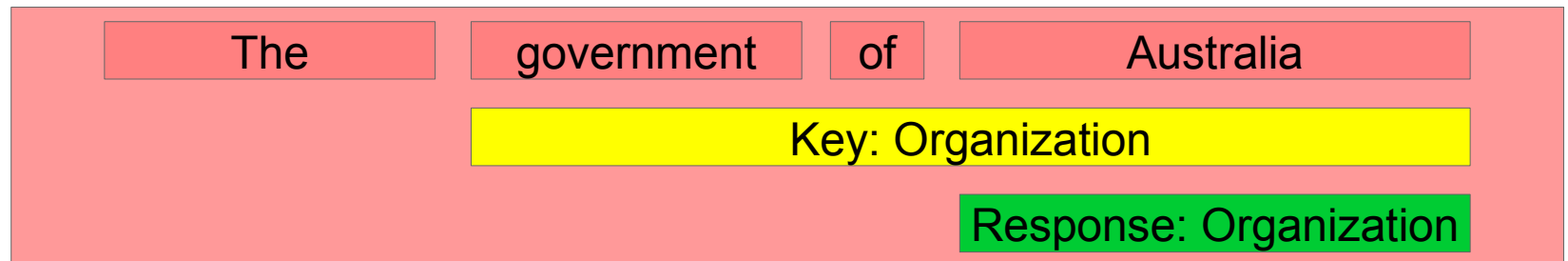
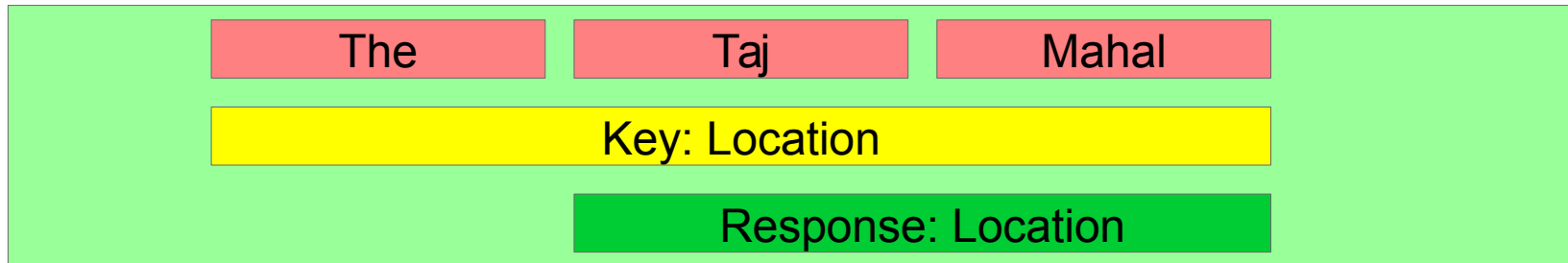
- Precision: what proportion of our automatic annotations were correct?
- Recall: what proportion of the correct annotations did our automatic tool create?
- $P = \text{correct} / (\text{correct} + \text{spurious}) = \text{tp} / (\text{tp} + \text{fp})$
- $R = \text{correct} / (\text{correct} + \text{missing}) = \text{tp} / (\text{tp} + \text{fn})$
- where tp = true positives, fp = false positives, fn = false negatives

What are precision, recall and F1?

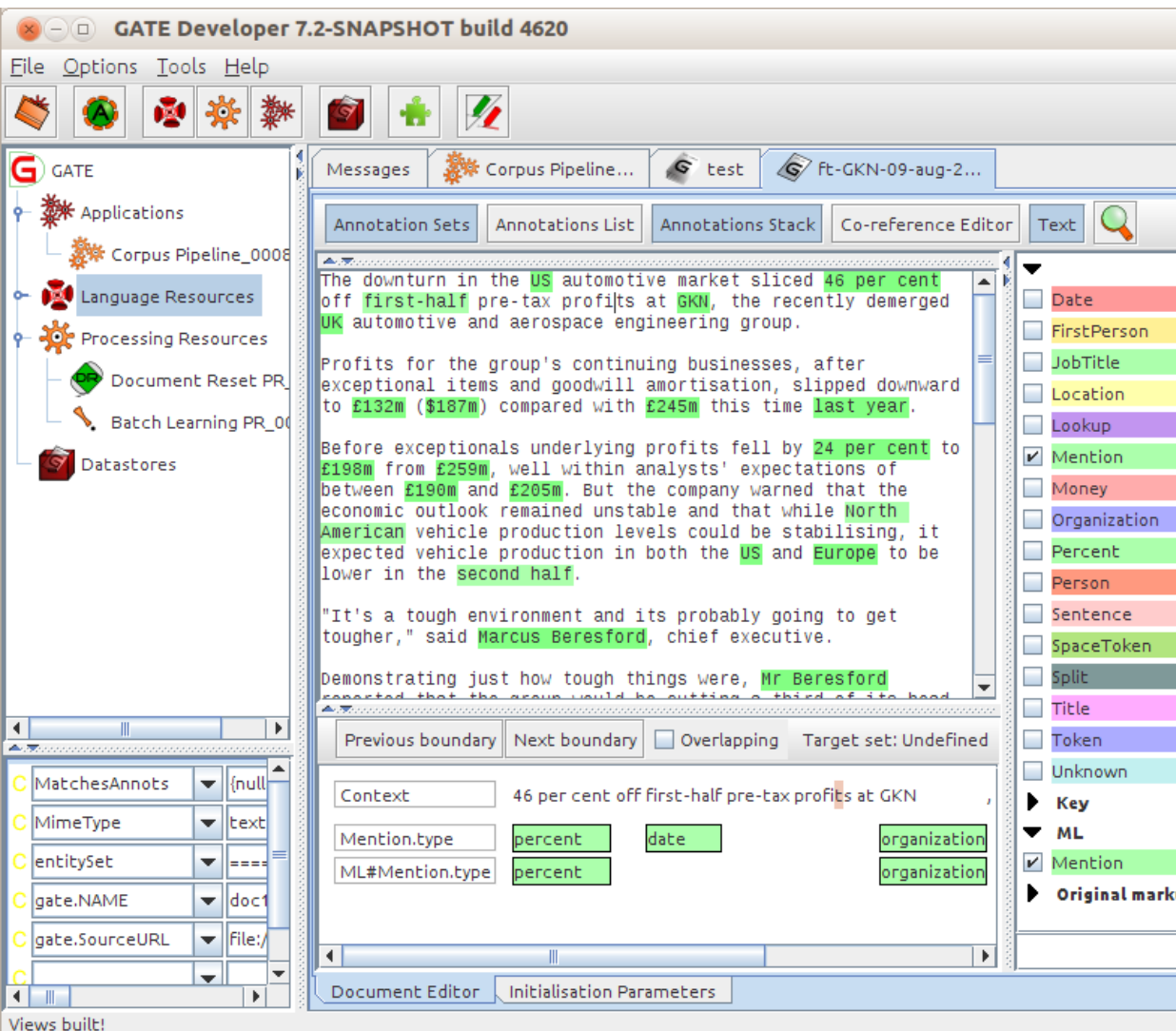
- F-score is an amalgam of the two measures
- $F_{\beta} = (1+\beta^2)PR / (\beta^2 P + R)$
 - The equally balanced F1 ($\beta = 1$) is the most common F-measure
 - $F1 = 2PR / (P + R)$

Strict and Lenient

- “Strict” means we count an annotation as correct only if it has the same span as the gold standard annotation
- Lenient means we allow an annotation that overlaps to be correct, even if it isn't a perfect span match
- Which do you think is the right way to do it?



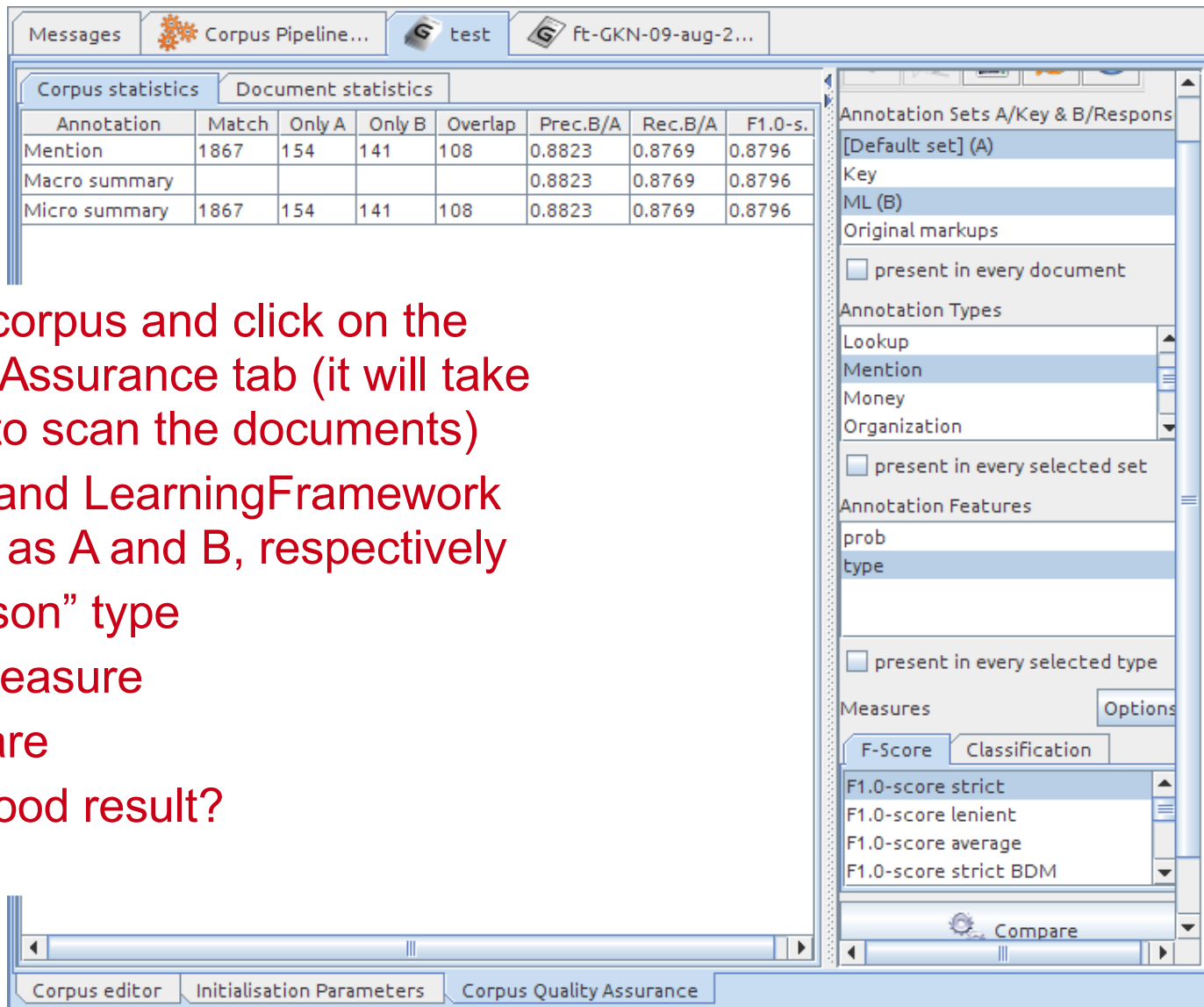
Examining the results of application



The screenshot shows the GATE Developer interface. The main window displays a document with several paragraphs of text. The text is annotated with various entities and mentions. For example, "US" is annotated as a location, "46 per cent" as a percent, "first-half" as a date, "GKN" as an organization, and "UK" as a location. The "Annotations Stack" panel on the right shows a list of annotation sets, including "Date", "FirstPerson", "JobTitle", "Location", "Lookup", "Mention", "Money", "Organization", "Percent", "Person", "Sentence", "SpaceToken", "Split", "Title", "Token", "Unknown", "Key", "ML", and "Original marku". The "Mention" set is checked. The "Context" panel at the bottom shows the current context: "46 per cent off first-half pre-tax profits at GKN".

- **Examine a document from the test corpus**
- You should have a new “LearningFramework” AS with Person annotations
- The original Person annotations (in the Key AS) are similar but not always identical!
- The Annotations Stack is good for comparing them
- How similar do they appear to be? Do you think you will get a good result?

Comparing the Sets with Corpus QA



The screenshot shows the GATE Corpus Quality Assurance interface. The main window displays a table of statistics for the 'test' corpus. The table has columns for Annotation, Match, Only A, Only B, Overlap, Prec.B/A, Rec.B/A, and F1.0-s. The rows are Mention, Macro summary, and Micro summary. The F1.0-s column shows a value of 0.8796 for all rows.

Annotation	Match	Only A	Only B	Overlap	Prec.B/A	Rec.B/A	F1.0-s.
Mention	1867	154	141	108	0.8823	0.8769	0.8796
Macro summary					0.8823	0.8769	0.8796
Micro summary	1867	154	141	108	0.8823	0.8769	0.8796

The right-hand pane shows the configuration for the 'Key' annotation set. The 'Annotation Types' list includes 'Mention', which is selected. The 'Annotation Features' list includes 'prob' and 'type'. The 'Measures' section shows 'F-Score' selected, with 'F1.0-score strict' chosen from the dropdown menu. The 'Compare' button is visible at the bottom of the pane.

- Select the test corpus and click on the Corpus Quality Assurance tab (it will take a few seconds to scan the documents)
- Select the Key and LearningFramework annotation sets as A and B, respectively
- Select the “Person” type
- Choose an F-measure
- Click on Compare
- Did you get a good result?



Using Annotation Diff to examine performance

Annotation Difference

Key doc: ft-BT-briefing-02-a... Key set: [Default set] Type: Mention Weight:

Resp. doc: ft-BT-briefing-02-a... Resp. set: ML-results Features: all some none 1.0

Start	End	Key	Features	=?	Start	End	Response	Features
1517	1519	BT	{class=organization}	=	1517	1519	BT	{class=organization, prob=1.0}
171	173	2p	{class=money}	=	171	173	2p	{class=money, prob=1.0}
1956	1972	Deutsche · Telekom	{class=organization}	=	1956	1972	Deutsche · Telekom	{class=organization, prob=1.0}
46	55	yesterday	{class=date}	=	46	55	yesterday	{class=date, prob=1.0}
1322	1327	Oftel	{class=organization}	=	1322	1327	Oftel	{class=organization, prob=1.0}
867	882	January · 22 · 2001	{class=date}	=	867	882	January · 22 · 2001	{class=date, prob=1.0}
1198	1203	Scoot	{class=organization}	=	1198	1203	Scoot	{class=organization, prob=1.0}
514	524	Amazon.com	{class=organization}	~	514	520	Amazon	{class=organization, prob=1.0}
1753	1761	Scoot · UK	{class=organization}	-?				
1181	1195	late · last · year	{class=date}	-?				
1007	1017	Air · Canada	{class=organization}	-?				
1924	1926	DT	{class=organization}	-?				
				?-	1499	1511	0800 · 192 · 192	{class=money, prob=1.0}
482	488	Amazon	{class=organization}	<>	482	488	Amazon	{class=location, prob=0.99999946}
800	806	Amazon	{class=organization}	<>	800	806	Amazon	{class=location, prob=0.99999905}
756	762	Amazon	{class=organization}	<>	756	762	Amazon	{class=location, prob=1.0}

Correct: 36 Recall Precision F-measure
 Partially correct: 1 Strict: 0.82 0.88 0.85
 Missing: 7 Lenient: 0.84 0.90 0.87
 False positives: 4 Average: 0.83 0.89 0.86

93 documents loaded

- Switch to the “Document statistics” tab
- Choose a document
- Click on the Annotation Diff icon
- What kind of mistakes did your application make?

Using Annotation Diff...

- “Correct”: the response annotation has the right feature and span
- “Partially correct”: response has the right feature and overlapping but not exactly matched span; this counts as correct in the “lenient” scoring
- “Missing”: key annotation+feature is missing from the response (a.k.a. “false negative”)
- “False positive”: response annotation+feature shouldn't be there (a.k.a. “spurious”)



Classification Evaluation PR for Chunking?

- We didn't use a Learning Framework evaluation PR for this chunking task
- What do you think would happen if you used the Classification Evaluation PR to do a chunking problem?
- It would work! It would evaluate the accuracy of the system in correctly identifying beginnings, insides and outsides
- However, it wouldn't tell you much about how well you did finding named entities
 - There are so many outsides that you can get a high score just by saying everything is an outside!
- You could use it to tune parameters if you wanted, though



Exercise—Improving the result

- Again, see if you can improve your result
- Try different features and algorithms



Exercise 2

- Try to learn different entity types

Exporting Feature Data



Exporting feature data

- A GATE ML PR serves a number of functions
 - Scraping features off the documents and formulating them as ML training sets
 - Sending the training sets to ML libraries to train a model
 - Creating instances (without class) at apply time to send to a trained model to be classified and writing the resulting class back onto the application instance
- We have integrated quite a few algorithms and some ML facilitation technology, so many ML tasks can be accomplished entirely in GATE

Exporting feature data

- However, GATE isn't an ML tool—its forte and contribution is complex linguistic features. There is a limit to what we will include in the way of ML innovations.
- For example, the Learning Framework;
 - doesn't include feature selection technologies
 - includes only limited feature scaling
 - doesn't integrate all algorithm variants



Exporting feature data

- For more advanced needs, there are other ways to work
- You can export your training set and use it to train a model outside of GATE
 - The Learning Framework will allow you to use a model trained outside of GATE to create an application
- Exporting data and working in e.g. Weka can also provide a faster way to tune parameters
 - When you change parameters in the LF it starts over again scraping the features off the documents, which takes a long time on a big corpus
- You could use e.g. Weka's feature selection technology and bring what you learned back into GATE by editing your feature spec
- It can also be a good sanity check to see your data in export format



Export the data as ARFF

- Create an Export PR and add it to the application
- You can remove the other Learning Framework PRs
- Annotation Set Transfer needs to stay though

Export Parameters

- `classAnnotationType` is as for training, and its presence indicates that we are exporting a CHUNKING dataset—set it to `Person`
- `dataDirectory`, `featureSpecURL`, `inputASName` and `instanceType` you are familiar with by now—set them
- For exporter, choose `EXPORTER_ARFF_CLASS*`
- Don't set target feature! This would indicate that we want to export a classification dataset!
- Don't set `sequenceSpan`—this would indicate that we want to export data in a format suitable for training a sequence learner. This isn't supported yet.

* "CLASS" means classification—why are we exporting a classification dataset for a chunking problem? Because they're all classification behind the scenes. GATE turns the chunking problem into a classification problem for training and then turns it back again!

GATE Developer 8.2-SNAPSHOT build 5490

File Options Tools Help

Messages ANNIE test

Language Resources

- in-tesco-citywire-07
- in-scoot-10-aug-200
- in-reed-10-aug-2001
- in-outlook-10-aug-2
- in-oil-09-aug-2001.x
- in-german-bank-10-
- in-bayer-10-aug-200
- in-airlines-08-aug-20
- in-GKN-citywire-10-
- gu-w&d-10-aug-200
- gu-telewest-10-aug-
- gu-synergie-10-aug-
- gu-singtel-10-aug-2
- gu-scoot-10-aug-20
- gu-ryanair.xml_0008
- gu-recession-6-aug-
- gu-manuf-jobs-07-a
- qu-m&s-10-aug-200

Loaded Processing resources

Name	Type
ANNIE NE Transducer	ANNIE NE Transducer
ANNIE OrthoMatcher	ANNIE OrthoMatcher
LF_ApplyChunking 00031	LF_ApplyChunking
LF_TrainChunking 00030	LF_TrainChunking

Selected Processing resources

Name	Type
Document Reset PR	Document Reset
ANNIE English Tokeniser	ANNIE English Tok
ANNIE Gazetteer	ANNIE Gazetteer
ANNIE Sentence Splitter	ANNIE Sentence S
ANNIE POS Tagger	ANNIE POS Tagge
Annotation Set Transfer 00036	Annotation Set Tr
LF_Export 00099	LF_Export

Run "LF_Export 00099"?

Yes No If value of feature is

Corpus: test

Runtime Parameters for the "LF_Export 00099" LF_Export:

Name	Type	Required	Value
algorithmParameters	String		
classAnnotationType	String		Person
dataDirectory	URL	✓	file:/home/genevieve/svn/sale/talks/gate-course-jun16/module-3-ml-barbour
exporter	Exporter	✓	EXPORTER_ARFF_CLASS
featureSpecURL	URL	✓	file:/home/genevieve/svn/sale/talks/gate-course-jun16/module-3-ml-barbour
inputASName	String		
instanceType	String	✓	Token
scaleFeatures	ScalingMethod	✓	NONE
sequenceSpan	String		
targetFeature	String		
targetType	TargetType	✓	NOMINAL

Run this Application

Serial Application Editor Initialisation Parameters About...

ANNIE run in 2.811 seconds

- Set targetType to nominal, because beginnings, insides and outsides are nominal classes
- Go ahead and export the data!

Examining the ARFF



```
data.arff (~/.svn/sale/talks/gate-course-jun16/module-3-ml/chunking-hands-on) - gedit
File Edit View Search Tools Documents Help
data.arff x
@attribute A:Token:string=eyes numeric
@attribute A:Token:string=gyms numeric
@attribute A:Token:string=contributes numeric
@attribute A:Token:string=Like-for-like numeric
@attribute A:Token:string=645 numeric
@attribute A:Token:string=Separately numeric
@attribute A:Token:string=small-cap numeric
@attribute A:Token:string=Espress numeric
@attribute A:Token:string=Top numeric
@attribute A:Token:string=Notch numeric
@attribute class {0,B,I}

@data
{0 1,1 1,2 4,3 1,4 1}
{1 1,2 2,5 1}
{1 1,2 7,3 1}
{1 1,2 18,3 1}
{1 1,2 7}
{1 1,2 3}
{1 1,2 8}
{1 1,2 3}
{1 1,2 6}
{2 1}
{1 1,2 1}
{1 1,2 4}
{1 1,2 2,5 1}
Plain Text Tab Width: 8 Ln 1, Col 1 INS
```

- You'll find your exported ARFF in your dataDirectory, called data.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?

Working with Weka

Why would I want to use Weka?

- As noted previously, Weka can be faster and better for playing around with parameters to get the best result
 - Now that you have exported your data, you can try loading it into Weka in your own time, and see what you can do there
- But then you need to bring that result back into GATE! So you need to run the Weka algorithm in GATE
- Weka has some good algorithms that might be better for your task
 - Though note that Mallet's CRF is often the best for chunking, and LibSVM is often the best for most things, and you don't need Weka for those
- However, due to licensing incompatibility, we can't integrate Weka into GATE as seamlessly as we integrated LibSVM and Mallet

What you need

- Weka integration comes as a separate project, but it's easy to do!
- You need to get the Weka wrapper from here (downloading the zip is easiest):

<https://github.com/GateNLP/weka-wrapper/>

- You need to tell your application where to find the Weka wrapper
 - Use the environment variable `WEKA_WRAPPER_HOME`
 - Or use the java property `gate.plugin.learningframework.wekawrapper.home`
 - Or the setting `wekawrapper.home` in a file `weka.yaml` in the data directory used

Using Weka in the GATE GUI

- Then you can go ahead and use Weka for classification and chunking by:
 - Creating a training PR
 - Selecting WEKA_CL_WRAPPER for trainingAlgorithm
 - Giving the full class name of the Weka algorithm as the first algorithmParameters argument
 - For example “weka.classifiers.trees.RandomForest”
 - A model will be created in the specified directory as before
 - At apply time, you simply indicate this model as usual
- (Weka in the evaluation PR isn't supported—try using Weka to evaluate!)

Where to find documentation about ...

- Getting the Weka wrapper and using it to train models outside of GATE:
 - <https://github.com/GateNLP/weka-wrapper>
- Using Weka inside of GATE:
 - <https://github.com/GateNLP/gateplugin-LearningFramework/wiki/UsingWeka>
- What Weka algorithms' full class names are:
 - Weka's Javadoc, e.g.
<http://weka.sourceforge.net/doc.dev/weka/classifiers/Classifier.html>
- Note that the Weka wrapper is very new code! Let us know if you find any problems with it!