Information Extraction with GATE

Leon Derczynski
University of Sheffield

9 Dec 2013

„Infocommunication technologies and the society of future (FuturICT.hu)" TÁMOP-4.2.2.C-11/1/KONV-2012-0013

About this tutorial

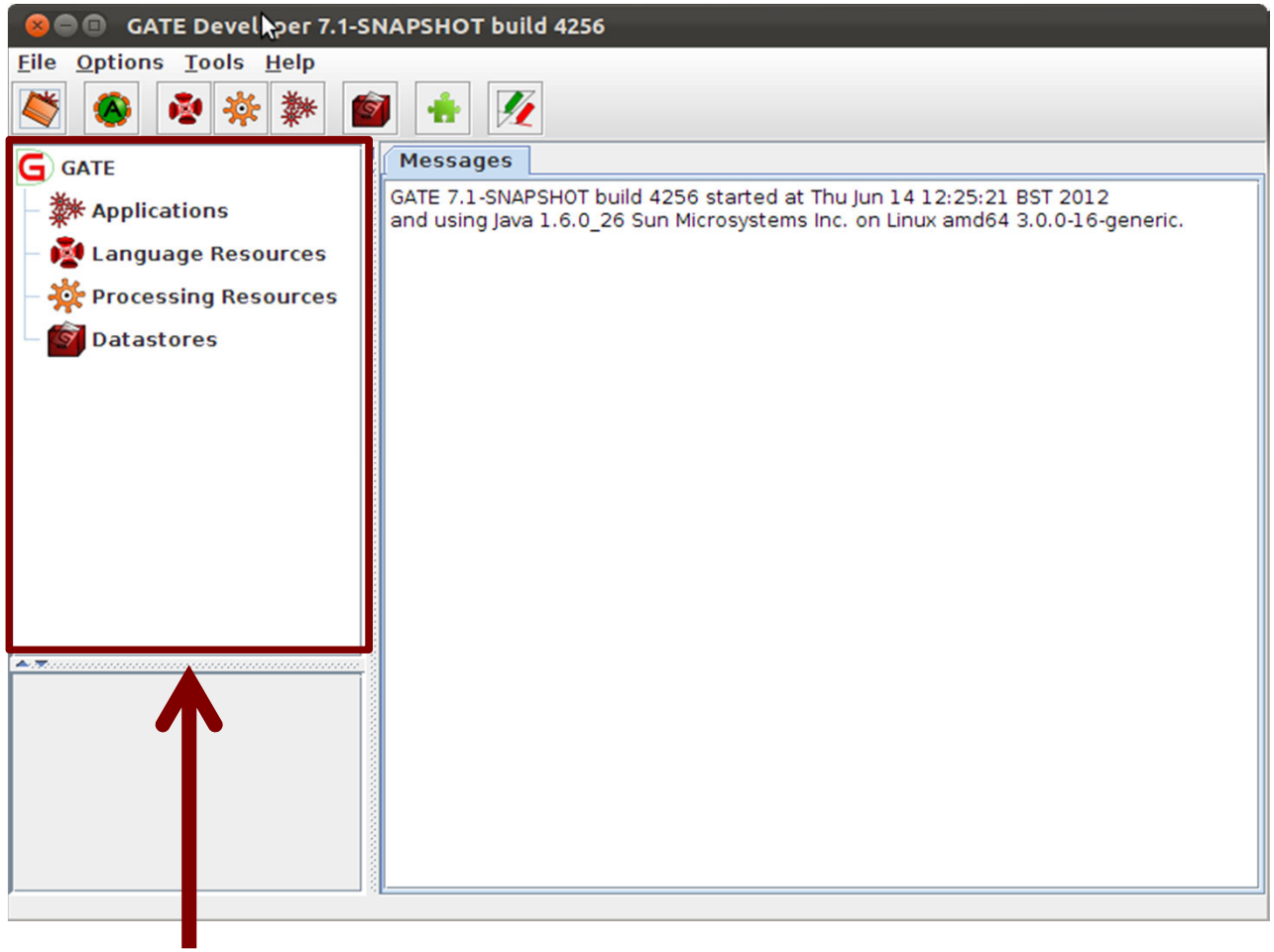
- This tutorial will be a hands on session with some explanation as you go.
- As topics are introduced, there'll be time for you to try playing with different parts of the GUI
- Things for you to try yourself are in red.
- Start GATE on your computer now (if you haven't already)
- There'll be extra time at the end to practise again, or go on to some further exercises. Please don't jump ahead: if you're already familiar with some topics, perhaps you can help your neighbour if they get stuck.
- This tutorial is about how to **use** the various components. Later, you'll learn more about the underlying functionality. So please reserve your burning questions about this for a little bit longer!

Time to get your hands dirty!

1. Finding your way around the GATE GUI

- How to navigate the GATE GUI
- How to set up the different options
- Introduction to resources and parameters

Resources Pane



Resources Pane

Resources Pane

- **Language resources** (LRs) are documents or document collections
  - a collection of documents is known as a **corpus**
- **Processing resources** (PRs) are annotation tools that operate on text within the documents
- **Data stores** are specialised files where documents are kept for future use
- **Applications** are groups of processes that run on one or more documents

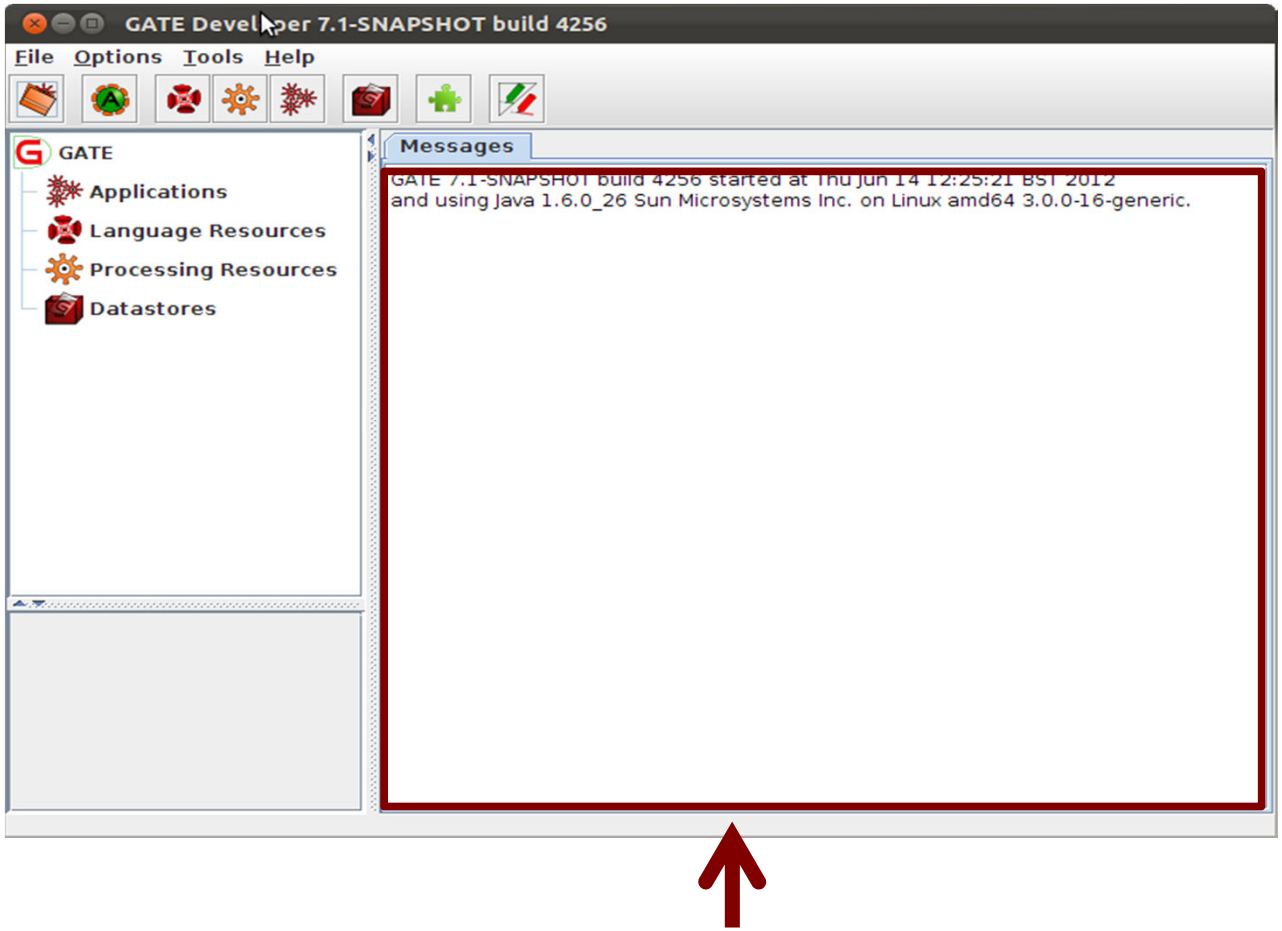
Simple operations on resources

- In general, right clicking on the name of a resource in the resource pane gives access to a menu of actions
- Double clicking on an instance of a resource enables you to view the resource
- Selecting a resource instance and pressing Delete will generally close it
- You can also right click and then select “Close”

Parameters

- Applications, LRs, and PRs all have various parameters which can be set either at load time (initialisation) or at run time.
- Parameters enable different settings to be used, e.g. case sensitivity
- **Initialisation Parameters** (set at load time**)** cannot be changed without reloading (these may be called “init parameters” for short)
- **Run time Parameters** can be changed between each application run
- Later you'll be able to experiment with setting parameters on resources and applications

Display Pane



Display Pane

Displaying Elements

- When you first open GATE, the Display page will typically just display any messages from the system
- It displays whatever elements you are currently working with, e.g. an application, a document or a processing resource
- Double clicking on an instance of any resource will generally display it
- Along the top of the pane may be various tabs which allow you to toggle the views of any open resources
- Clicking on a tab displays that view
  - E.g. “Messages” tab shows messages

Setting up GATE options

- You can set up different options in GATE using the Options menu.
  - Click Options → Configuration → Appearance to change the look and feel of GATE, such as menu and text fonts
  - Try a few different options.
- Clicking the Advanced tab enables you to adjust settings such as saving your options, and saving the session so that when you reopen GATE, it will remember and reload the applications you had open at the end of your previous session
- You can try this out later.

2. Loading and Viewing Documents

- Loading a document and setting its parameters
- Navigating through documents and viewing their annotations

Loading a document

- When GATE loads a document, it converts it into a special format for processing
- GATE can process documents in all kinds of formats: plain text, HTML, XML, PDF, Word etc.
- Documents have a markupAware parameter which is set to true by default: this ensures GATE will process any existing annotations such as HTML tags and present them as annotations rather than leaving them in the text.
- Documents can be exported in various formats or saved in a datastore for future processing within GATE

Loading documents

- To load a document, you can right click on Language Resources and select “New → GATE Document”
- You can also go via the File menu --> New Language Resource → GATE Document
- The sourceURL parameter enables you to specify the document to be loaded. You can type the filename or URL, or click the file browser icon to navigate to the correct document.
  - Try loading a file from your hands on materials and one from the Web – you must include the http:// part when specifying a URL
- You can also just type a string of text into the box. In this case, you need to select stringContent rather than sourceUrl, using the arrow, before typing the text.
- Try loading a document via the stringContent method

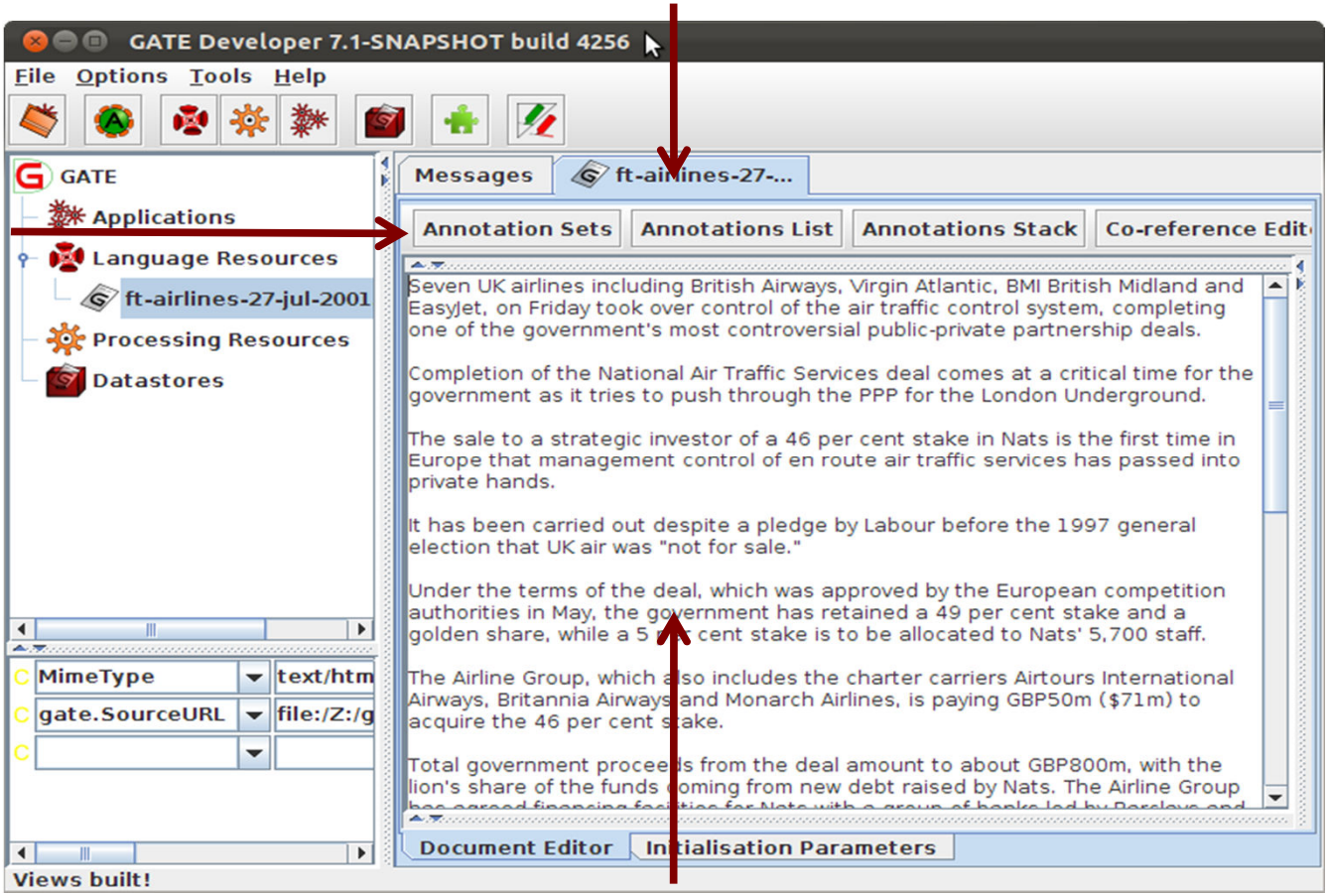
Initialisation parameters

- A document has a variety of init parameters: some compulsory and some optional
- Compulsory parameters have a tick in the “Required” box
- You can provide your own name or use the default name GATE provides (document name + a unique ID, which prevents confusion with multiple copies of the same document)
- Note that the same approach to naming applies with other kinds of resources such as PRs

Document viewer

Highlighted tab is the resource currently being viewed

Document viewer buttons



Document

Opening and closing documents

- To view a document, double click on the document name in the Resources pane
- To close a document, right click on the document name and select “Close”
- To hide a document, while leaving it loaded, right click on the document tab and select “Hide”
- The Document viewer buttons at the top of the Display pane let you select different views
- To view the annotations, you first need click “Annotation Sets”, and then select the relevant set and annotation(s) on the right
- To see a list of annotations at the bottom, click on “Annotations List”

➢ Load the “ft-airlines-27-jul-2001.xml” file from your hands-on folder

3. All about Annotations

- Introduction to annotations, annotation types and annotation sets
- Creating and viewing annotations

Annotations

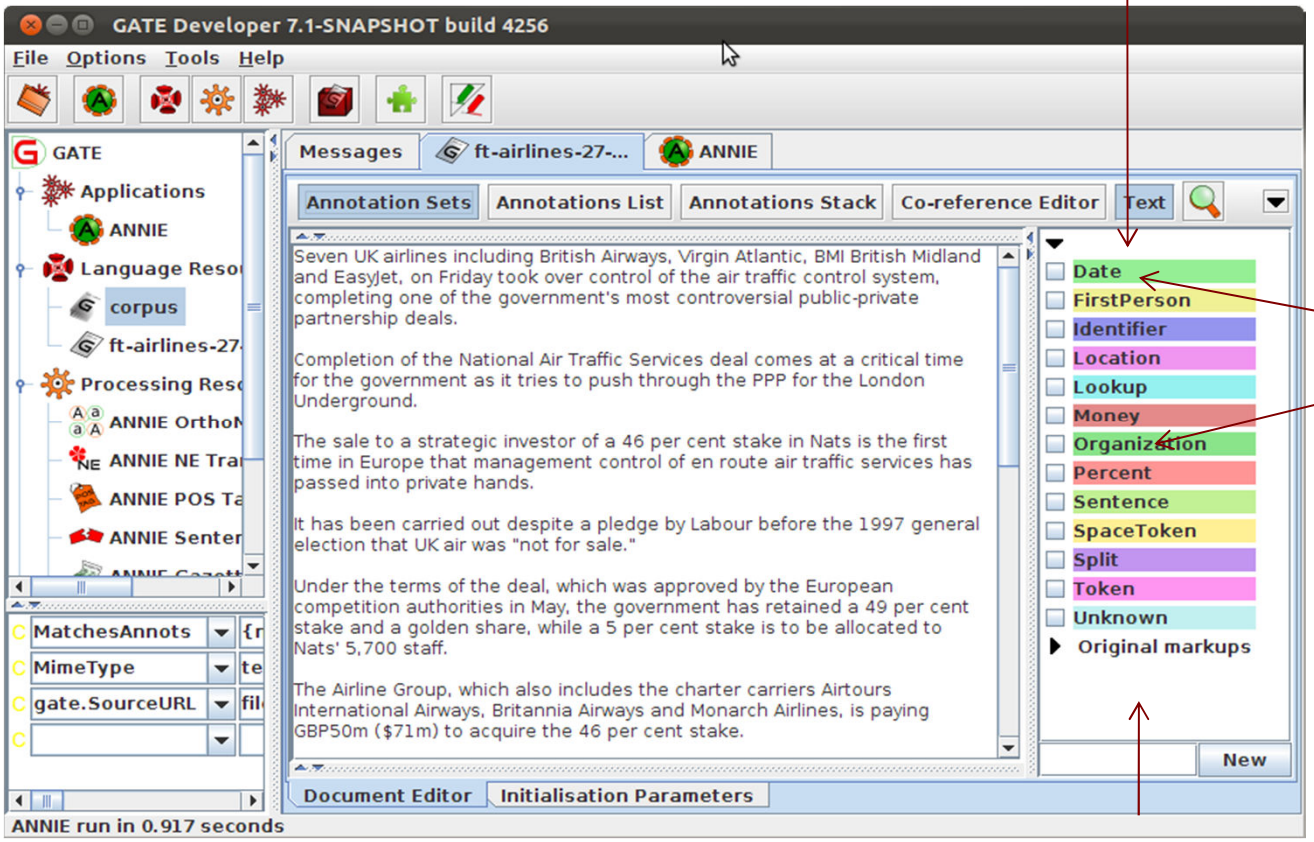
- The annotations associated with each document are a structure central to GATE.
- Each annotation consists of
- start and end offsets
- optionally a set of features associated with it
- each feature has a name and a value

Annotation Sets

- Annotations are grouped into sets, e.g. Default, Original Markups
- Each set can contain a number of annotation types, e.g. Person, Location etc.
- You can create and organise your annotation sets as you wish.
- It's useful to keep different sets for different tasks you may perform on a document, e.g. to separate the original HTML tags from your new annotations
- It's important to understand the distinction between annotation set, annotation type, and annotation
- This is best explained by looking at them in the GUI

Annotation Sets

Default annotation set



Annotation types

Original Markups annotation set

Viewing annotations

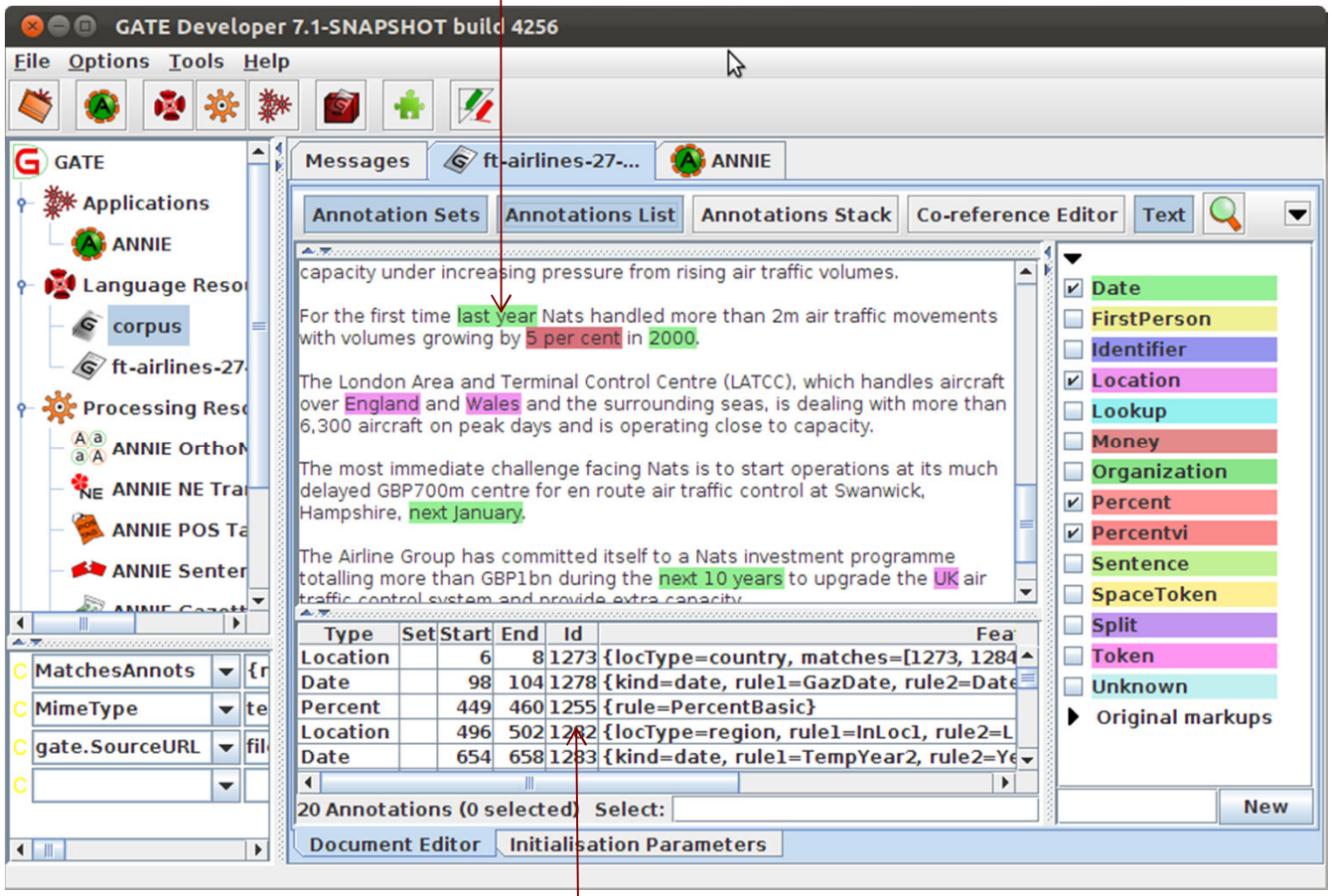
- Double click on your document to view it
- Click on the Annotation Sets button to open a new pane on the right hand side (Annotation Sets view)
- Default (unnamed) set contains some examples of annotations
- Click on the arrow to display the annotation types belonging to that set
- You should see types such as Location, Date, Person etc.
- Select an annotation type to view all the annotations of that type in the document

A closer look at the annotations

- Select the Annotations List button from the menu above the Display pane
- For each annotation type selected in the Annotation sets view, all annotations corresponding to that type will be shown in the table
- Table shows annotation type, offsets, annotation set, features and values
- Select a row in the table to highlight the annotation in the text
- Click on a column heading to sort according to the header
- There are also other annotation views possible such as the AnnotationStack and Coreference Editor: we'll look at these later

Annotations

Date annotation

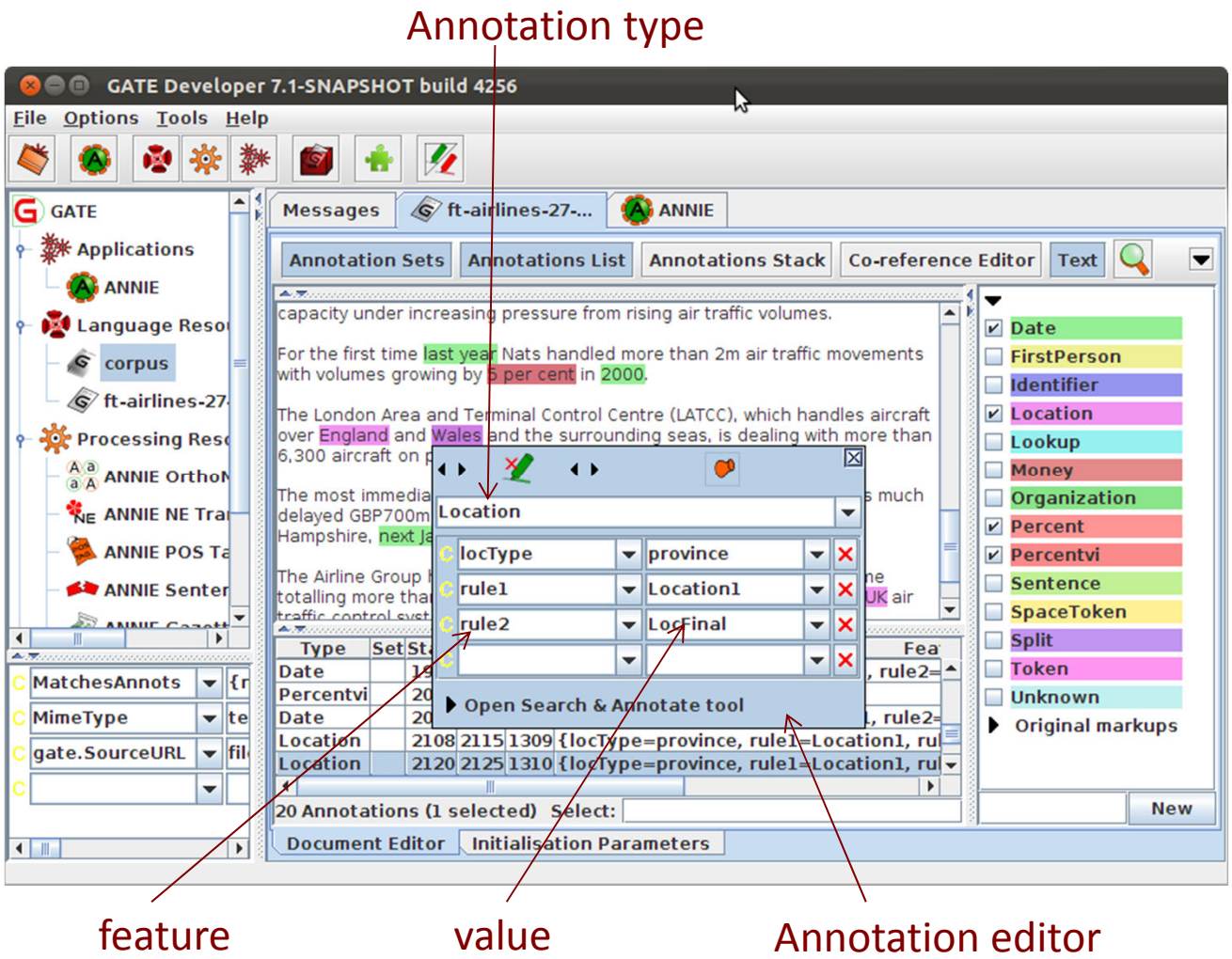


Annotations table

Editing existing annotations

- Select an annotation type from the Annotation Sets view and hover over a highlighted annotation in the text
- A popup window displays more information about it: this is the annotation editor
- Click the drawing pin symbol at the top of the editor. This will “pin” the window open (you can still move the window around on your screen if you wish)
- Try editing the annotation: you can change the annotation type, feature names and values, the span of the annotation (clicking left and right arrows at the top of the box) or delete the annotation or its features (red Xs)
- Close the annotation editor by clicking the X in the top right corner, then view your edited annotation in the Annotation List

Annotation editor



Creating new annotations

- To create a new annotation, select the portion of text you want to annotate and hover over it with the mouse.
- The annotation editor will appear: this will automatically create a new annotation.
- It will create an annotation of the same type as your last annotation: if this is your first annotation it will default to "_New_". You can change this by simply editing the text.
- You can edit this annotation as before.
- You can delete the annotation by clicking on the red cross/green crayon icon
- The new annotations will appear in the currently selected annotation set. To change this, simply select a different set.
- To create a new annotation set, enter a name in the text field at the bottom of the annotation sets view and click "New".

➢ Try creating some new annotations in your text.

Creating a Corpus

- A corpus is a collection of documents.
- For most GATE applications, it is easier to work with a corpus rather than an individual document, even if that corpus only contains one document.
  - Right click Language Resources → New → GATE Corpus
  - Click the edit button [add icon] and add your document to the corpus
- As with the documents, you can name your corpus or use the default GATE name.
  - Double click on the corpus name in the Resources pane to view the corpus.
  - Double click the document listed there to view it.

Another way to add documents to a corpus

- You can also create an empty corpus and then add documents to it, if these documents are already loaded in GATE
  - Create another corpus as before, but do not select any documents to add to it
  - Open the corpus and use the + button to add documents, or drag them from the Resources pane

Removing documents

- To remove documents from a corpus, use the X button in the corpus editor
- Note that this does not remove the document from GATE, just from the corpus
  - The document is available to be added to other corpora. Indeed a document can belong to several corpora
- If you do remove the document from GATE, it will also remove it from the corpus
  - But if you remove the corpus, it doesn't remove the document!
- Try experimenting with adding and removing documents

More about corpora

- You can use the up and down arrows to rearrange documents in a corpus
- Click on the tab at the bottom to view the initialisation parameters of the corpus

Populating a Corpus (1)

- Usually, a corpus will consist of more than one document. Sometimes there could be hundreds of documents in a corpus.
- Using the populate function means you don't have to preload the documents in GATE first, and allows you to load all the documents into the corpus in one go
- To do this, let's first tidy up a bit
- It's best to keep GATE GUI clutter-free by removing any unwanted resources and documents, or it can get a bit confusing
  - Close all open documents and corpora

Populating a Corpus (2)

- Create a new corpus as before, but don't add any documents to it yet
- Right click on the corpus name in the Resources pane and select Populate
- Use the file browser icon to select the name of the directory with your documents
- The Extensions parameter lets you select only documents of a certain type.
  - Press the edit button to see a list of allowed types
  - Type “xml” in the box (without the quotes)
  - Press “Add” and then “OK”
- “Encoding” lets you choose the right encoding for the documents. The wrong encoding can cause characters to be incorrectly displayed
  - Enter “UTF-8” here
- “Recurse directories will also load documents in any subdirectories
  - Deselect the “Recurse directories” box
- As if by magic, all the documents will be loaded in one go
- View the contents of the corpus as before.

Cheat's tip for quick corpus creation

- If you're just testing something on one document, there's a quick way to create a new corpus and add the document to it.
- Right click on the document loaded in GATE and select “New corpus with this document”.
- This does everything in one go.
- Try it on any document you have loaded.
- Note that a document can belong to more than one corpus at the same time, but it can get confusing if you do this!

5. Processing Resources and Plugins

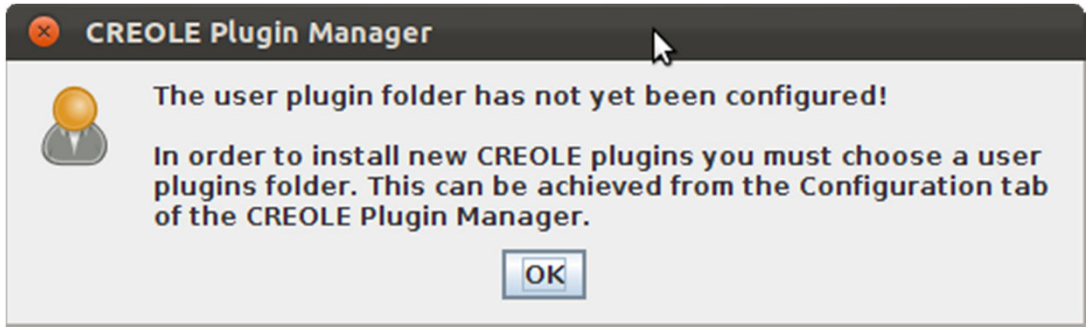
- Loading processing resources and managing plugins

Processing Resources and Plugins

- Processing resources (PRs) are the tools that enable annotation of text. They implement algorithms. Typically this means creating or modifying annotations on the text.
- An application consists of any number of PRs, run sequentially over a corpus of documents
- A plugin is a collection of one or more PRs, bundled together. For example, all the PRs needed for IE in Arabic are found in the Lang_Arabic plugin.
  - A plugin may also contain language or visual resources, but you don't need to worry about that now!
- An application can contain PRs from one or more different plugins.
- In order to access new PRs, you need to load the relevant plugin

Plugins

- Click the icon on the top GATE menu to open the Plugin Manager [or go via File->Manage CREOLE Plugins]
- Depending on your version of GATE, you may see a popup box:



- User plugin folder is a folder on your hard-drive where plugins other than those provided by GATE are stored

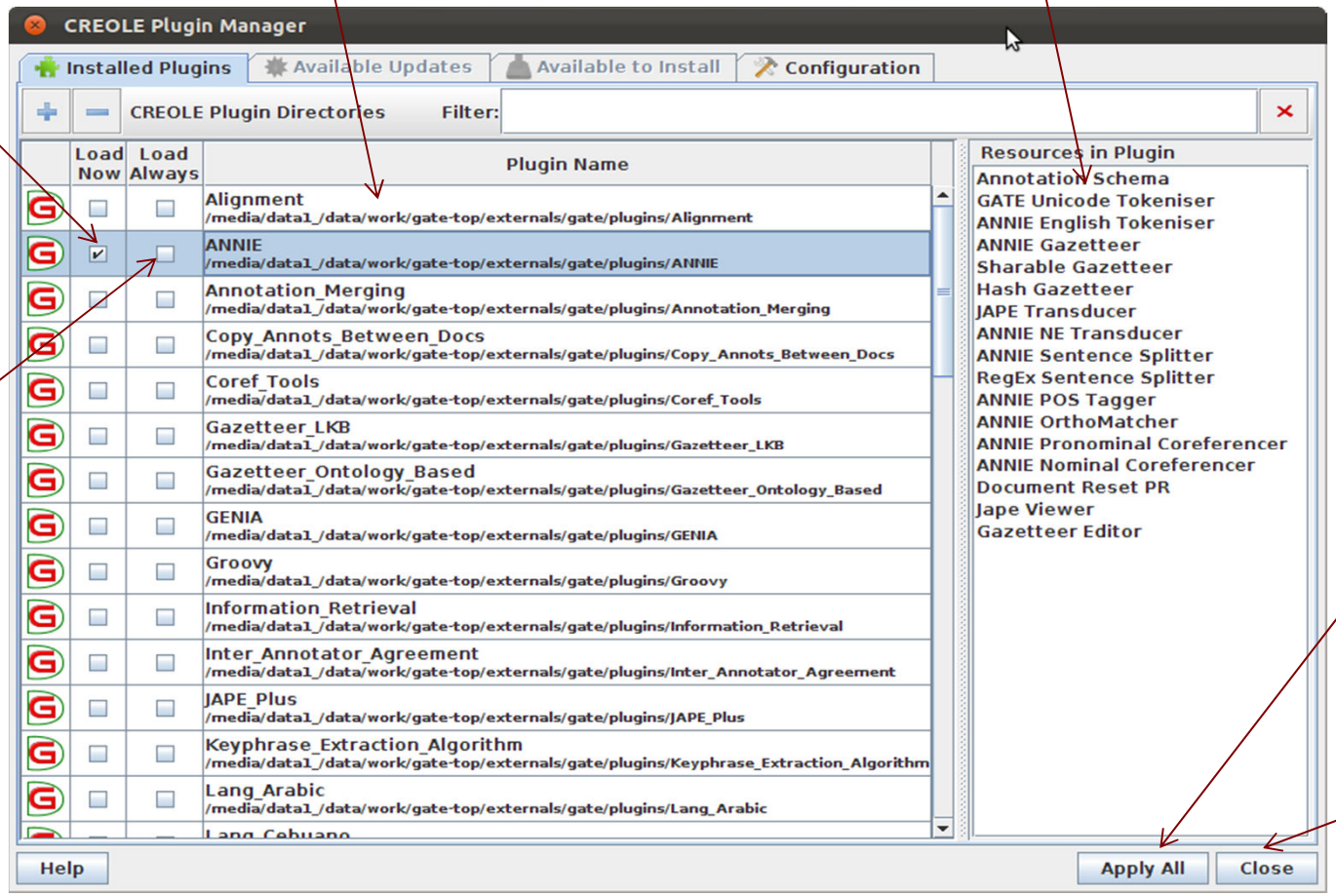
Plugins

Load the plugin for this session only

List of available plugins

Resources in the selected plugin

Load the plugin everytime GATE starts



Apply all the settings

Close the plugins manager

Plugins

- Select a plugin to see (on the RHS) the names of the resources it contains
- Check the relevant “Load Now” box to load a plugin of your choice
- Click “Apply All” to load the selected plugin
- Click “Close”
- Right click on Processing Resources to see which new PRs are now available

6. Applications

- Loading and running ANNIE and pre-existing applications
- Creating a new application

Here's one I made earlier: ANNIE

- ANNIE is a readymade collection of PRs that performs IE on unstructured text. For those who grew up in the UK, you can think of it as a Blue Peter-style “here's one we made earlier”.
- We'll use ANNE as an example GATE application.
- Later, we'll show you how to make your own application from scratch.

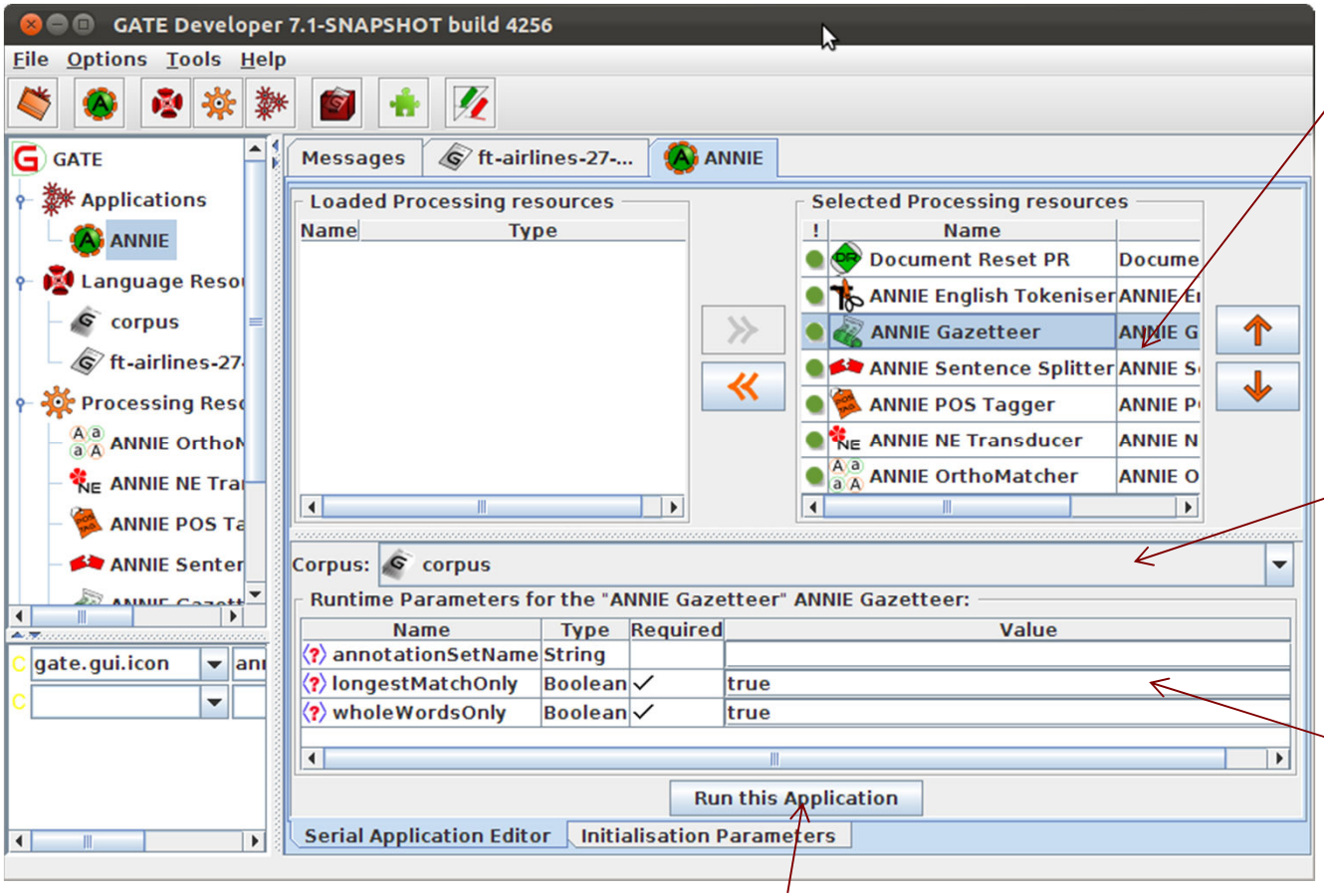
➢ Click the icon from the top GATE menu OR Select File → Load ANNIE system

➢ Select “with defaults”

➢ Load any document from the hands-on material and add it to a corpus

Running an application

- View the ANNIE application by double clicking on it



PRs selected in application (in order of their execution)

Corpus on which the application is executed

Runtime parameters of the selected PR

Execute the application

Viewing the results

- When a message appears in the bottom left corner of your GATE window saying something like “ANNIE run in 1.3 seconds”, the application has finished.
  - Double click on the document to view it
  - View the annotations by selecting Annotation Sets and clicking on any Annotation types in the Default (unnamed) set
  - If you want, you can view the annotations table too.
- Remember that not all the results will be perfect! Later in the course, you'll learn more about the causes of these errors.

Input and output annotation sets

- Some PRs use the results of previous PRs in the application. For example, the sentence splitter makes use of Token annotations produced by the tokeniser.
- The inputAS (annotation set) for the sentence splitter is the name of the annotation set where it will find the Token annotations
- The outputAS is the name of the set where it will produce the results of the sentence annotations.
- In ANNIE, the inputAS and outputAS are always the same. Later, we'll look at examples where you might want these to be different.
- Some PRs just have a parameter “annotationSetName” instead. This is because the inputAS and outputAS must be the same for that PR (usually because the PR adds information to an existing annotation rather than creating a new one)

Changing runtime parameters

- Now we're going to change the name of the annotation set, so that all ANNIE annotations appear in a new set called ANNIEresult
- The annotation set where the results are stored is one of the runtime parameters of the PRs

➢ Double click on ANNIE to view the application and PRs.

➢ For each PR listed, click on it and check whether it has any parameters labelled “annotationSetName”, “inputASName” or “outputASName”

➢ Edit all of these by typing “ANNIEresult” in the box.

➢ Double check that you haven't missed any. This is really important, otherwise your application may not work.

➢ Now run the application again and view the results.

Adding new PRs (1)

- Let's add a Verb Phrase Chunker PR to ANNIE.
- First, we have to load the plugin that contains it, and then load the PR into GATE, before we can add it to the application.
  - Use the plugins manager to load the Tools plugin.
  - Right click on Processing Resources and select “New” → “ANNIE VP Chunker”
  - Leave all the default parameters set and click “OK”.
  - To find out more about the VP Chunker, right click and select “Help”.

Adding new PRs (2)

- Now we need to add the new PR to the application.
  - Double click on ANNIE.
- You'll see the VP chunker is in the list of loaded PRs. This means it's available in GATE, but isn't yet contained in the application.
  - Add it to the application by selecting it and using the right arrow to transfer it.
  - Now use the up arrow to move it to the right place in the application. It should go after (below) the POS tagger but before (above) the NE transducer.
  - Change the inputASName and outputASName parameters to ANNIEresult.
  - Run the application and view the results on the document.
- You should see a new annotation type “VG”.

7. Saving documents

- Using datastores
- Saving documents for use outside GATE

Types of datastores

There are 2 types of datastore:

- Serial datastores store data directly in a directory
- Lucene datastores provide a searchable repository with Lucene-based indexing

For now, we'll look at serial datastores. We will not look at Lucene (searchable) datastores today.

Create a new serial datastore

- Right click “Datastores” from the Resources pane and select “Create Datastore”
- Select “Serial Datastore”
- Create a new empty directory by clicking the “Create New Folder” icon and give your new directory a name
- Select this directory and click “Open”
- Now your datastore is ready to store your documents

Save documents to the datastore

- Right click on your corpus and select “Save to Datastore”
- Select the datastore that you just created
- Now close the corpus and document
- Double click on the name of the datastore in the Resources pane
- You should see the corpus and document
- Double click on them to load them back into GATE and view them
- They should contain the annotations you created previously
- You can remove things from the datastore by right clicking on their name in the datastore and selecting “Delete”
- You can add several corpora to the same datastore

If you have lots of documents..

- A datastore is the best way to store them, because it uses less memory in GATE when processing
  - Delete all corpora and documents in your datastore
  - Load a new corpus (Language Resources → New → GATE Corpus)
  - Create a new datastore and save the (empty) corpus to the datastore
  - Now populate your corpus (right click on corpus → Populate)
- You should see the documents appear in your datastore
- As if by magic, your documents will be loaded into the datastore and saved automatically.
  - Close and reopen your datastore to check they really were saved!

Saving documents outside GATE

- Datastores can only be used inside GATE, because they use some special GATE-specific format
- If you want to use your documents outside GATE, you can save them in 2 ways:
  - as standoff markup, in a special GATE representation
  - as inline annotations (preserving the original format)
- Both formats are XML-based. However “save as xml” refers to the first option, while “save preserving format” refers to the second option.

Saving as XML

- Load any document from the hands-on material into GATE, then right click on it in the Resources pane
- Select “Save as XML” and select a filename.
- In this format, all annotations are appended to the end of the document and the location for each annotation is marked by a tag in the body of the document
- Each annotation has a unique ID
- If you’re curious, load the document into your favourite text editor and have a look at it!

Save preserving format

- This option will save the document with all the original annotations from HTML or XML documents, and any new annotations that you currently have selected in the document editor
- This can be useful for saving only selected annotation types
- Annotations are saved using standard XML tags, with the annotation type as the tag name
- Partially overlapping annotations can not be saved
- Right click on a document and select “Save preserving format”
- If the Advanced Option in GATE “Include annotation features for save preserving format” has been checked, then selected features will be saved as well as annotations, in this mode.
- You can play with this option on your own later.

Summary

- This tutorial has given you a guided tour of the GATE GUI
- Looked at language resources, datastores, applications and processing resources
- There are lots of other tools and options you can play with: see the User guide for more info
- Tomorrow we'll look at the topic of Information Extraction, and ANNIE, GATE's default IE system

Extra exercises

If you have some spare time, you can try some more exercises:

- Load an HTML or XML document with the markupAware parameter set to false and see the difference
- Investigate the AnnotationStack
- Play with Advanced Options
- Run an application over documents in a datastore

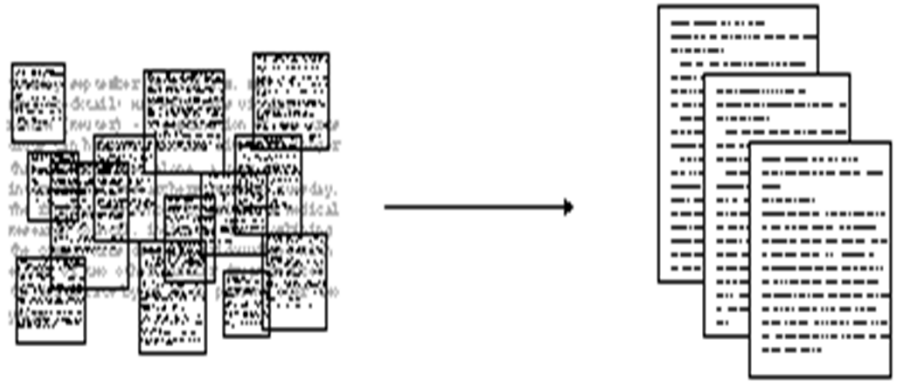
Information Extraction with GATE

„Infocommunication technologies and the society of future (FuturICT.hu)" TÁMOP-4.2.2.C-11/1/KONV-2012-0013

What is information extraction?

IE is not IR

- IR pulls **documents** from large text collections (usually the Web) in response to specific keywords or queries. You analyse the **documents**.



- IE pulls **facts** and **structured information** from the content of large text collections. You analyse the **facts**.

IE for Document Access

- With traditional query engines, getting the facts can be hard and slow
- Where has the Queen visited in the last year?
- Which airports are currently closed due to the volcanic ash?
- Which search terms would you use to get these?
- How can you specify you want to see someone's home page?
- IE returns information in a structured way
- IR returns documents containing the relevant information somewhere

IE as an alternative to IR

- IE returns knowledge at a much deeper level than traditional IR
- It allows you to specify your query in a more structured way
- Constructing a database through IE and linking it back to the documents can provide a valuable alternative search tool
- Even if results are not always accurate, they can be valuable if linked back to the original text

What is IE used for?

- IE is an enabling technology for many other applications:
- Text Mining
- Semantic Annotation
- Question Answering
- Opinion Mining
- Decision Support
- Rich information retrieval and exploration
- and so on..

Two main types of IE systems

Knowledge Engineering

- rule based
- developed by experienced language engineers
- make use of human intuition
- require only small amount of training data
- development can be very time consuming
- some changes may be hard to accommodate

Learning Systems

use statistics or other machine learning

developers do not need IE expertise

require large amounts of annotated training data

some changes may require re-annotation of the entire training corpus

Named Entity Recognition: the cornerstone of IE

Traditionally, NE is the identification of proper names in texts, and their classification into a set of predefined categories of interest

- Person
- Organisation (companies, government organisations, committees, etc)
- Location (cities, countries, rivers, etc)
- Date and time expressions

Various other types are frequently added, as appropriate to the application, e.g. newspapers, ships, monetary amounts, percentages.

Why is NE important?

- NE provides a foundation from which to build more complex IE systems
- Relations between NEs can provide tracking, ontological information and scenario building
- Tracking (co-reference): “Dr Smith”, “John Smith”, “John”, “he”
- Ontologies: “Athens, Georgia” vs “Athens, Greece”

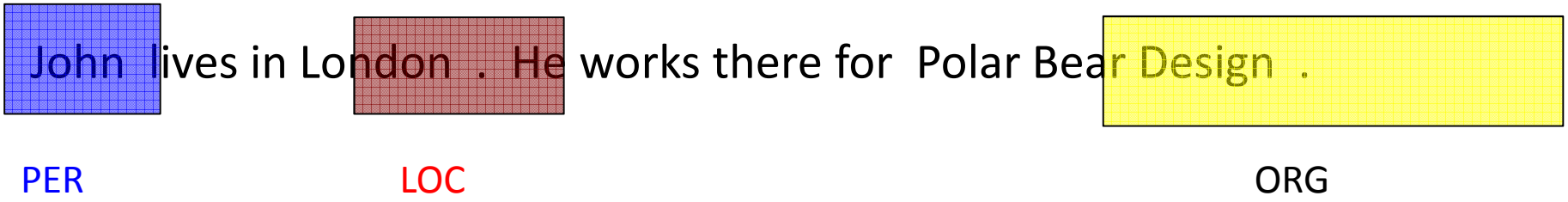
Typical NE pipeline

- Pre-processing (tokenisation, sentence splitting, morphological analysis, POS tagging)
- Entity finding (gazetteer lookup, NE grammars)
- Coreference (alias finding, orthographic coreference etc.)
- Export to database / XML / ontology

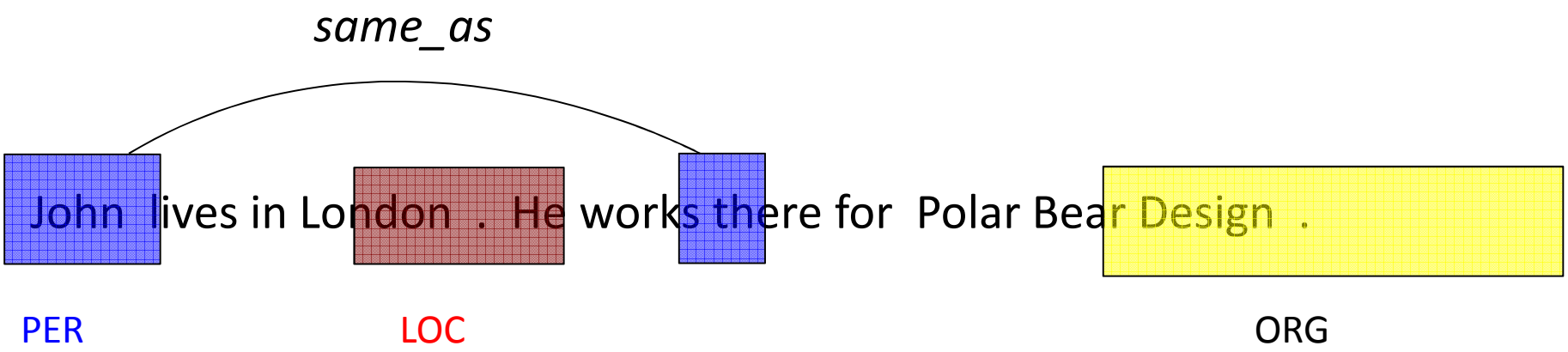
Example of IE

John lives in London . He works there for Polar Bear Design .

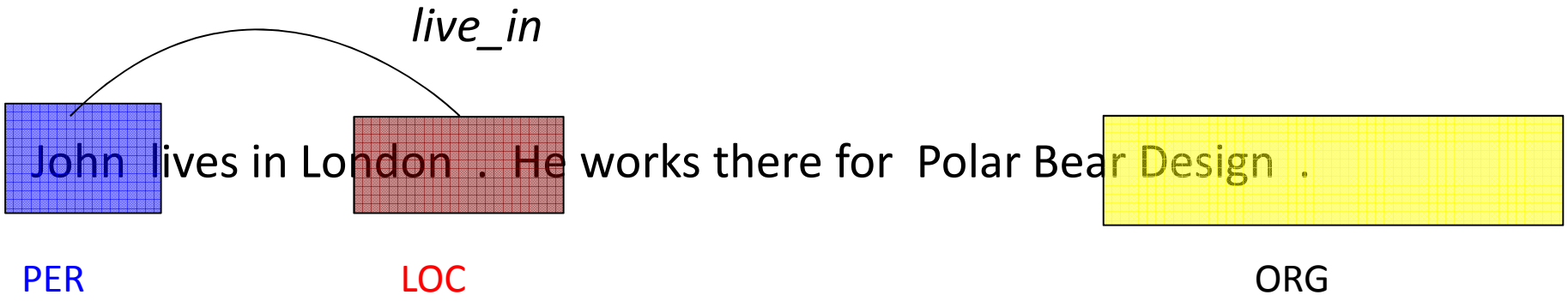
Basic NE Recognition



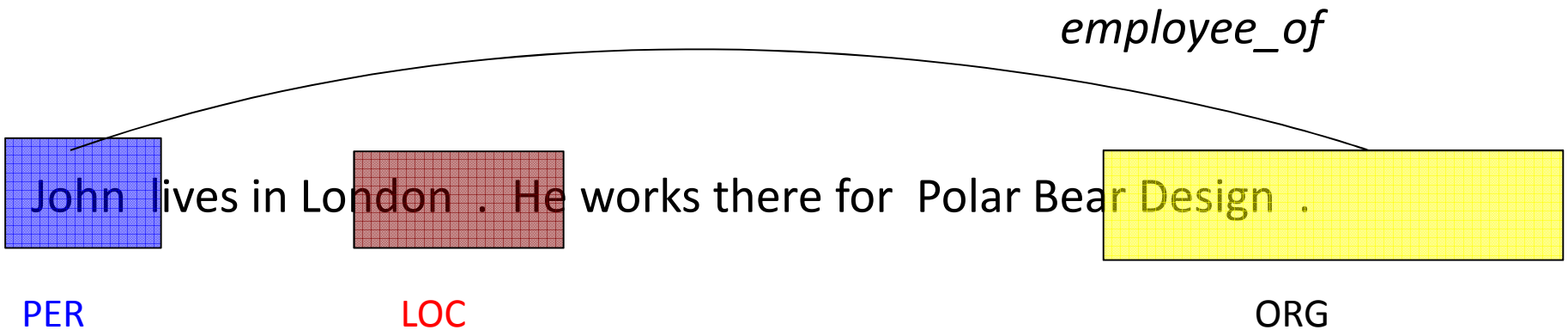
Co-reference



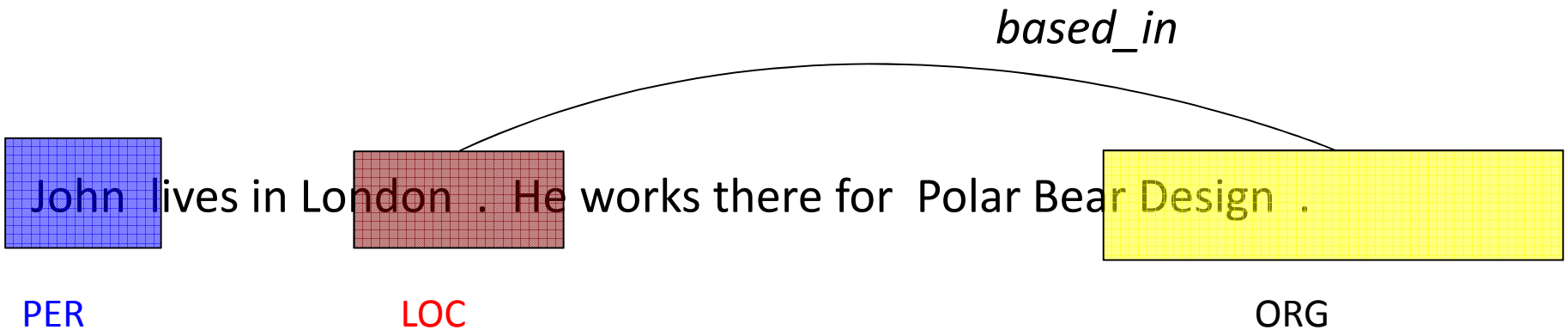
Relations


live_in
John lives in London . He works there for Polar Bear Design .
PER
LOC
ORG

Relations (2)


employee_of
John lives in London . He works there for Polar Bear Design .
PER
LOC
ORG

Relations (3)


based_in
John lives in London . He works there for Polar Bear Design .
PER
LOC
ORG

Examples of IE systems

HaSIE

- Health and Safety Information Extraction
- Application developed with GATE, which aims to find out how companies report about health and safety information
- Answers questions such as:
- “How many members of staff died or had accidents in the last year?”
- “Is there anyone responsible for health and safety?”
- IR returns whole documents

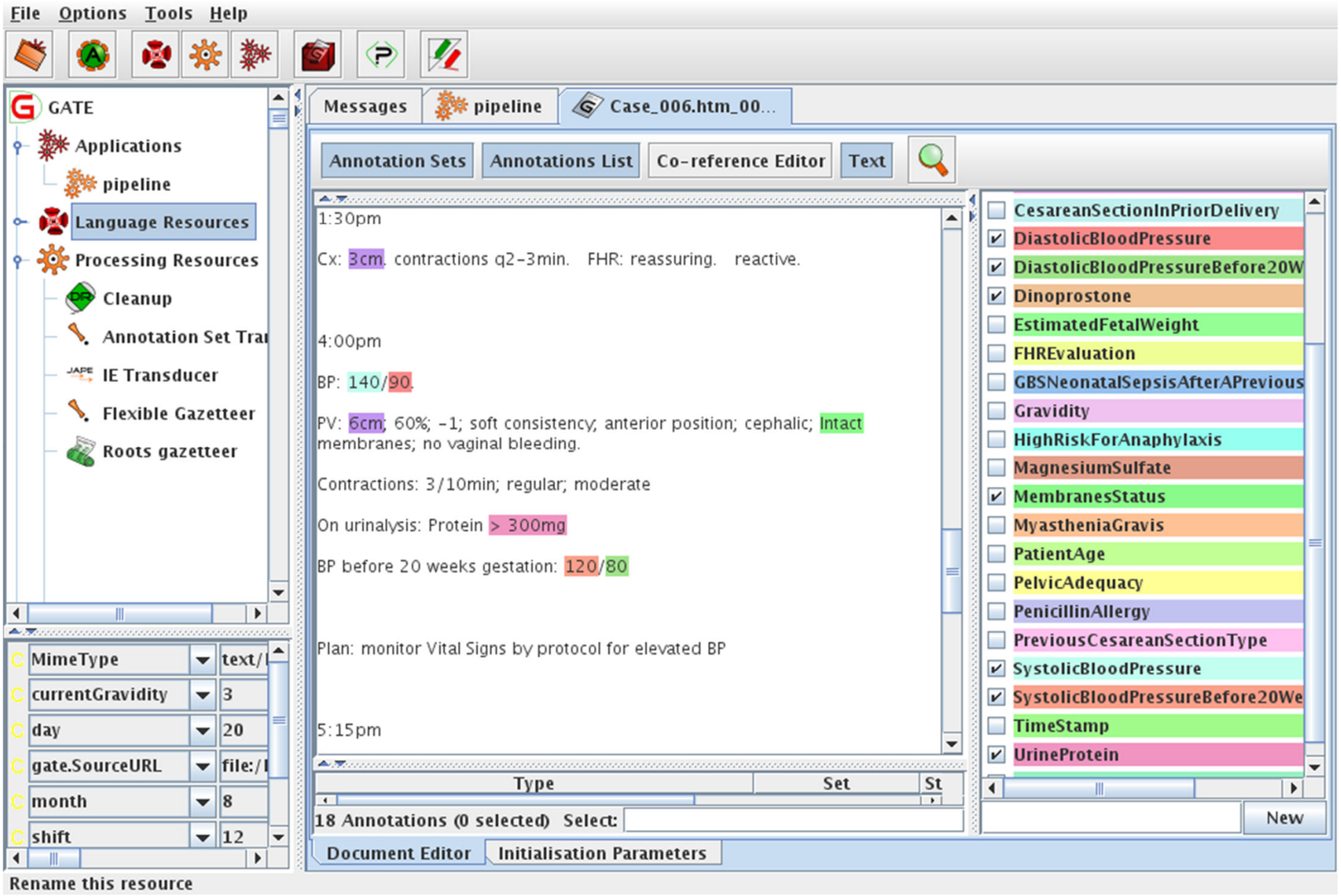
Hse

CompanyName	BAA
HSEParagraphs	sustainability management system. ... BAA has received a RoSPA gold award for occupational safety for the fourth year running. The award is given only if a consistently good or continuously improving performance can be demonstrated over a four-year period. The accident frequency ratio for construction projects was 0.4 (0.49) per 100,000 hours worked, less than one third of the national accident frequency rate in the construction sector. The company is running a ?One in a Million? campaign to raise safety consciousness and standards in construction and reduce the accident frequency rate still further to one for every million man hours worked. ... We have no higher priority than the safety and security of the passengers, staff and organisations that use our airports. In order to ensure that our systems and practices are continually assessed and upgraded, we work
Awards	BAA has received a RoSPA gold award
Accidents	The accident frequency ratio for construction projects was 0.4 (0.49) per 100,000 hours worked, less than one third of the national accident frequency rate in the construction sector.

Record: 1 of 36

Obstetrics records

- Streamed entity recognition during note taking
- Interventions, investigations, etc.
- Based entirely on gazetteers and JAPE
- Has to cope with terse, ambiguous text and distinguish past events from present
- Used upstream for decision support and warnings


File Options Tools Help
GATE
Applications
pipeline
Language Resources
Processing Resources
Cleanup
Annotation Set Trai
IE Transducer
Flexible Gazetteer
Roots gazetteer
MimeType
text/
currentGravidity
3
day
20
gate.SourceURL
file:/
month
8
shift
12
Rename this resource
Messages
pipeline
Case_006.htm_00...
Annotation Sets
Annotations List
Co-reference Editor
Text
1:30pm
Cx: 3cm. contractions q2-3min. FHR: reassuring. reactive.
4:00pm
BP: 140/90.
PV: 6cm; 60%; -1; soft consistency; anterior position; cephalic; Intact membranes; no vaginal bleeding.
Contractions: 3/10min; regular; moderate
On urinalysis: Protein > 300mg
BP before 20 weeks gestation: 120/80
Plan: monitor Vital Signs by protocol for elevated BP
5:15pm
Type
Set
St
18 Annotations (0 selected) Select:
CesareanSectionInPriorDelivery
DiastolicBloodPressure
DiastolicBloodPressureBefore20W
Dinoprostone
EstimatedFetalWeight
FHREvaluation
GBSNeonatalSepsisAfterAPrevious
Gravidity
HighRiskForAnaphylaxis
MagnesiumSulfate
MembranesStatus
MyastheniaGravis
PatientAge
PelvicAdequacy
PenicillinAllergy
PreviousCesareanSectionType
SystolicBloodPressure
SystolicBloodPressureBefore20We
TimeStamp
UrineProtein
New
Document Editor
Initialisation Parameters

Multiflora

- IE system in the botanical domain
- Finds information about different plants: size, leaf span, colour etc
- Collates information from different sources: these often refer to plant features in slightly different ways
- Uses shallow linguistic analysis: POS tags and noun and verb phrase chunking
- Important to relate features to the right part of the plant: leaf size rather than plant size, colour of flowers vs colour of leaves etc.

Gate 2.1-beta1 build 1061

File Options Tools Help

Messages | R_a_FNA.txt_00743

Text | Annotations | Annotation Sets | Print

7. Ranunculus acris Linnaeus, Sp. Pl. 1: 554. 1753
□ Renoncule âcre, bouton d'or
Ranunculus acris var. latisectus Beck
Stems erect from short caudex or rhizome, never rooting nodally, hispid, strigose, or glabrous, base not bulbous. Roots never tuberous. Basal leaf blades pentagonal in outline, deeply 3- 5-parted, 1.8-5.2 X 2.7-9.8 cm, segments 1-2 X -lobed or -parted, ultimate segments narrowly elliptic or oblong to lanceolate, margins toothed or lobulate, apex acute to rounded. Flowers: receptacle glabrous; sepals spreading, 4-6(-9) x 2-5 mm, hispid; petals 5, yellow, 8-11(-17) X 7-13 mm. Heads of achenes globose, 5-7(-10) mm wide; achenes 2-3 X 1.8-2.4 mm, glabrous, margin forming narrow rib 0.1-0.2 mm wide; beak persistent, deltate, usually with tip short or long, straight or curved, subulate, 0.2-1 mm. 2n = 14.

Type	Set	Start	End	Features
PlantFeatures	Default	0	1	{type=number}
Header	Default	0	44	{}
PlantFeatures	Default	38	39	{type=number}
PlantFeatures	Default	103	113	{rule=HeadAdj}
Head	Default	119	124	{}
PlantFeatures	Default	125	130	{rule=HeadAdj}
PlantFeatures	Default	136	141	{rule=AdjHead}
Head	Default	142	148	{}
Head	Default	152	150	{}

Default annotations
- [x] Head
- [x] Header
- [ ] Lookup
- [x] PlantFeature
- [ ] Segment
- [ ] SegmentSplit
- [ ] Sentence
- [ ] SpaceToken
- [ ] Split
- [ ] Token

Original markups ar
- [ ] paragraph

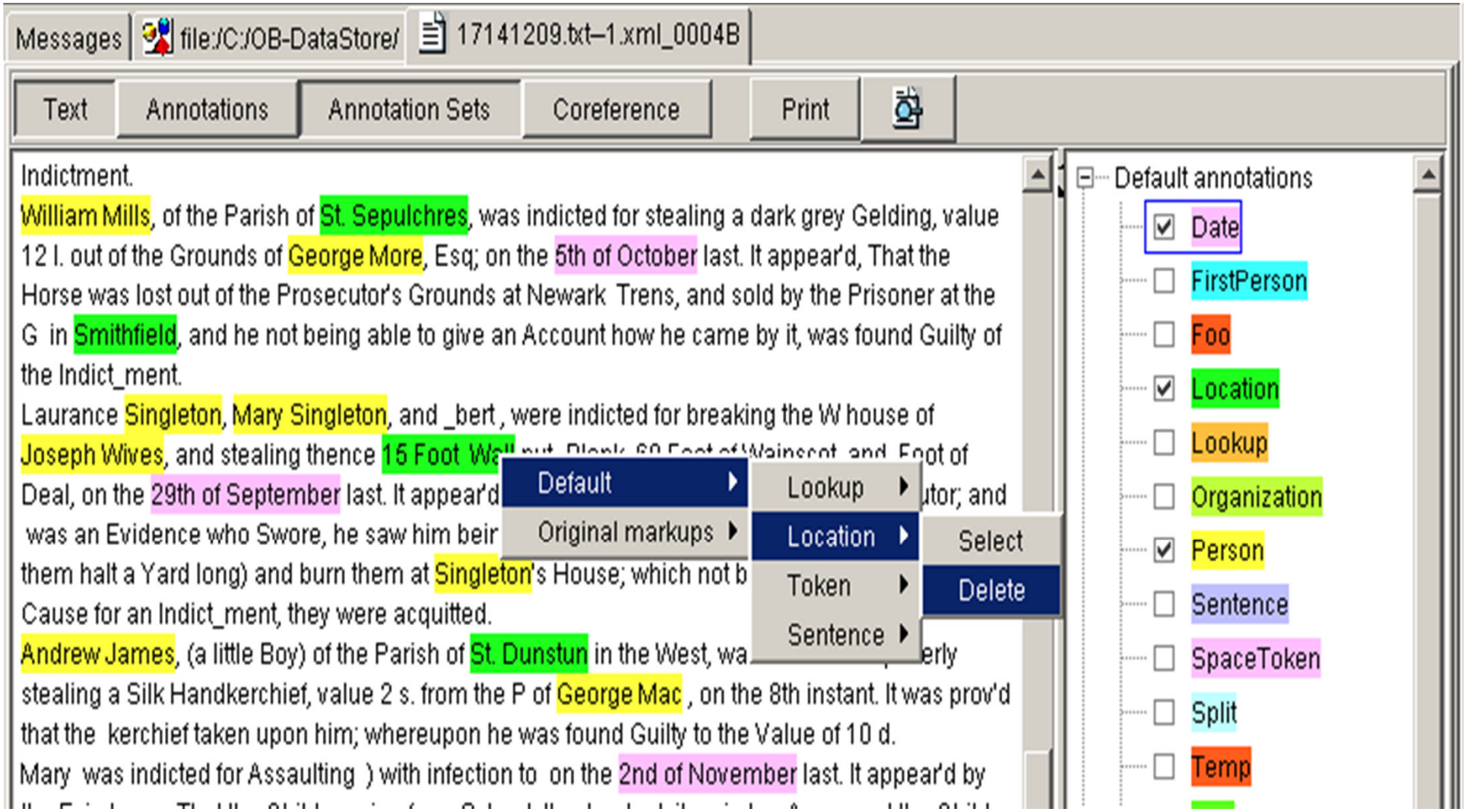
Annotations Editor | Features Editor

Hides this view

Old Bailey IE

- The Old Bailey Proceedings Online makes available a fully searchable, digitised collection of all surviving editions of the Old Bailey Proceedings from 1674 to 1913
- GATE was used to perform IE on the court reports, identifying names of people, places, dates etc.
- ANNIE was customised to only extract full Person names and to take account of old English language used
- More info at http://www.oldbaileyonline.org/static/Project.jsp

Old Bailey IE



IE in other languages

- ANNIE has been adapted to various other languages: some as test cases, some as real IE systems
- Many tools available as GATE PRs have datasets / models for different languages
- NER
- PoS tagging
- Gazetteer lookup
- Brief introduction to multilingual PRs in GATE later in this tutorial

Gate 2.1_02-beta build 1299

File Options Tools Help

te
Applications
arabic not trained
Language Resources
GATE document_00095
Processing Resources
orthomatcher
arabic not trained grammar
arabic gaz
arabic tokeniser
reset
Data stores
file:/share/nlp.18/diana/ga

GATE document_00095

file:/share/nlp.18/diana/gatecorpora/arabic/treebank/bbnfiles/test/processed/

Messages

Text | Annotations | Annotation Sets | Print

نيقوسيا 7-51 (أ ف ب) -0 ابدى نادي فيورنتينا الايطالي اهتمامه بضم مهاجم منتخب البرتغال ونادي بنفيكا نونو غوميش (42 عاما).

وكانت اندية ارسنال الانكليزي وفنربغشة التركي وديبورتيفو كورونا وريال سوسيداد الاسبانيان اعربت عن رغبتها في ضم غوميش الذي قدر بنفيكا قيمة انتقاله بنحو 61 مليون دولار .

وقع راديك بايل لاعب وسط منتخب تشيكيا ونادي اتلتيكو مدريد الاسباني الذي هبط الى 0 الدرجة الثانية عقدا انتقل بموجبه الى لنس الفرنسي لمدة 3 سنوات من دون ان تعرف قيمة الصفقة.

وكان بايل (72 عاما) احد افراد منتخب بلاد ه في كأس الامم الاوروبية الاخيرة لكنه خاض 031 دقيقة فقط في المباريات الثلاث التي خاضتها تشيكيا في البطولة لانها خرجت من الدور الاول .

وقع الكرواتي ميلان رابايتش مهاجم سروجي الايطالي عقدا لمدة سنتين مع فريق 0 فنربغشة التركي .

وبلغت قيمة انتقال رابايتش (72 عاما) نحو 5ر71 مليون دولار .

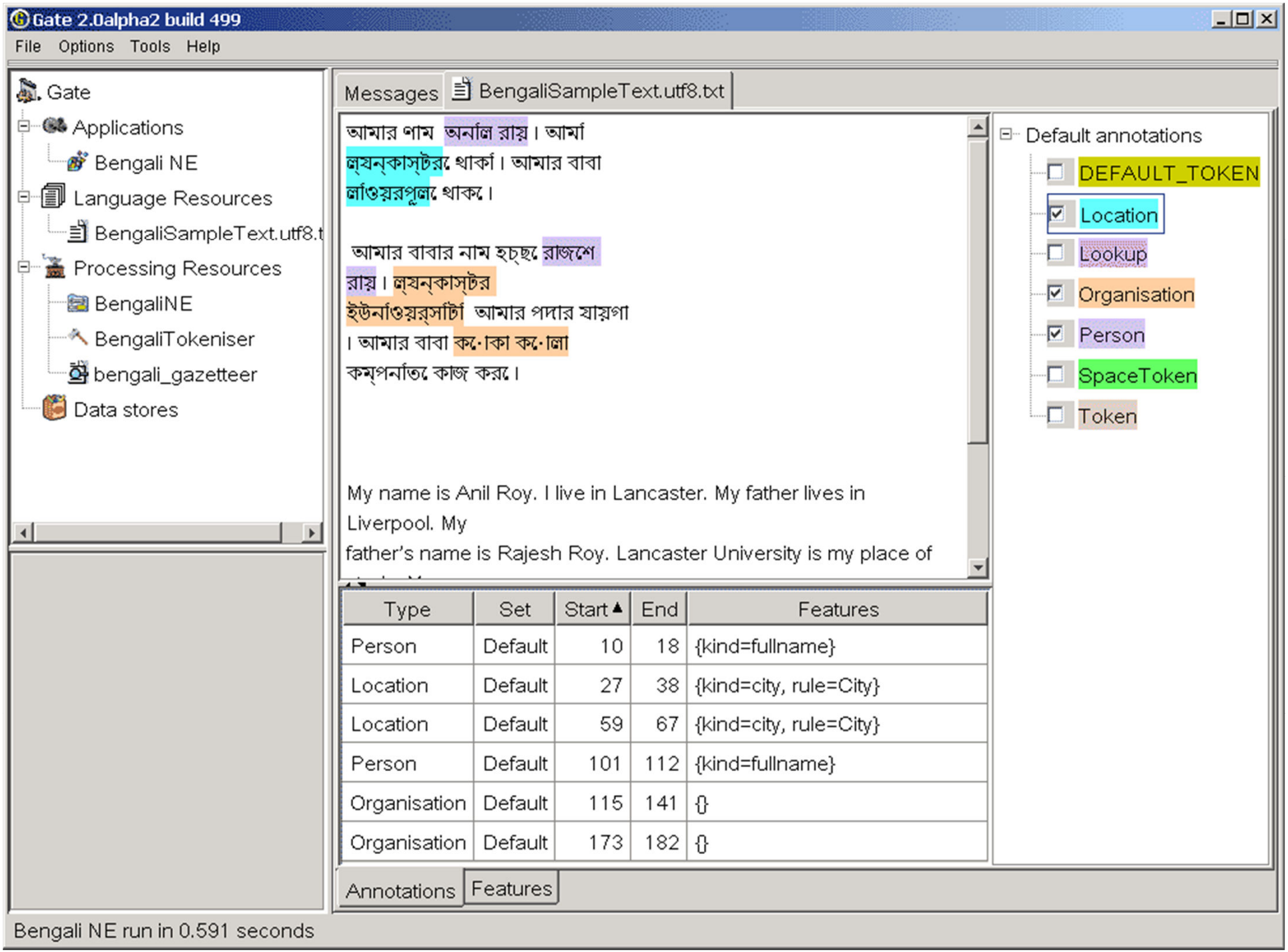
ولعب رابايتش 31 مباراة دولية مع منتخب بلاد ه وكان في صفوف نادي هايدوك سبليت الكرواتي قبل انتقاله الى ايطاليا .

Default annotations
Key annotations
- Cardinal
- Date
- Event
- Gpe
- Gpe_desc
- Money
- Nationality
- Ordinal
- Org_desc
- Organization
- Per_desc
- Person

Original markups annota

Annotations Editor | Features Editor | Initialisation Parameters

loaded in 2.677 seconds


Gate 2.0alpha2 build 499
File Options Tools Help
Gate
Applications
Bengali NE
Language Resources
BengaliSampleText.utf8.t
Processing Resources
BengaliNE
BengaliTokeniser
bengali_gazetteer
Data stores
Messages
BengaliSampleText.utf8.txt
আমার নাম অনিল রায়। আমি
ল্যন্কাস্টরে থাকি। আমার বাবা
লিভারপুলে থাকে।
আমার বাবার নাম হচ্ছে রাজেশ
রায়। ল্যন্কাস্টর
ইউনিভার্সিটি আমার পেশার যায়গা
। আমার বাবা কোকা কোলা
কম্পানিতে কাজ করে।
My name is Anil Roy. I live in Lancaster. My father lives in
Liverpool. My
father's name is Rajesh Roy. Lancaster University is my place of
Default annotations
DEFAULT_TOKEN
Location
Lookup
Organisation
Person
SpaceToken
Token
Type Set Start End Features
Person Default 10 18 {kind=fullname}
Location Default 27 38 {kind=city, rule=City}
Location Default 59 67 {kind=city, rule=City}
Person Default 101 112 {kind=fullname}
Organisation Default 115 141 {}
Organisation Default 173 182 {}
Annotations
Features
Bengali NE run in 0.591 seconds

ANNIE: A Nearly New Information Extraction system

About this tutorial

- As before, this tutorial will be a hands on session with some explanation as you go.
- We will use a corpus of news texts in the handout file. Unzip this file if it isn't already.
- Things for you to try yourself are in red.
  - There will be instructions for you to follow for each step
  - Each step will be demonstrated
  - Correct answers will be shown before moving on
- Restart GATE on your computer now (if you haven't already)

Extra exercises

- We need to pace the exercises for everyone.
- If it is too slow for you please feel free to skip through the exercises at your own pace.
- If you get a long way ahead, there are extra exercises at the end of these slides
  - You may not be able to do these extra exercises until you have finished the main tutorial exercises
  - You do not need to do this extra material to complete the tutorial. It is not a prerequisite for the rest of the course.

Extra exercises

- Note that instructions for the extra exercises are briefer than for the rest of the tutorial – they assume you now have the basics of GATE
- The extra exercises are:
  - Comparing different sentence splitters
  - Further evaluation exercises
  - Using the QA tools to compare three IE systems
- ANNIE
- LingPipe
- OpenNLP
  - Demonstration of an ontology based gazetteer

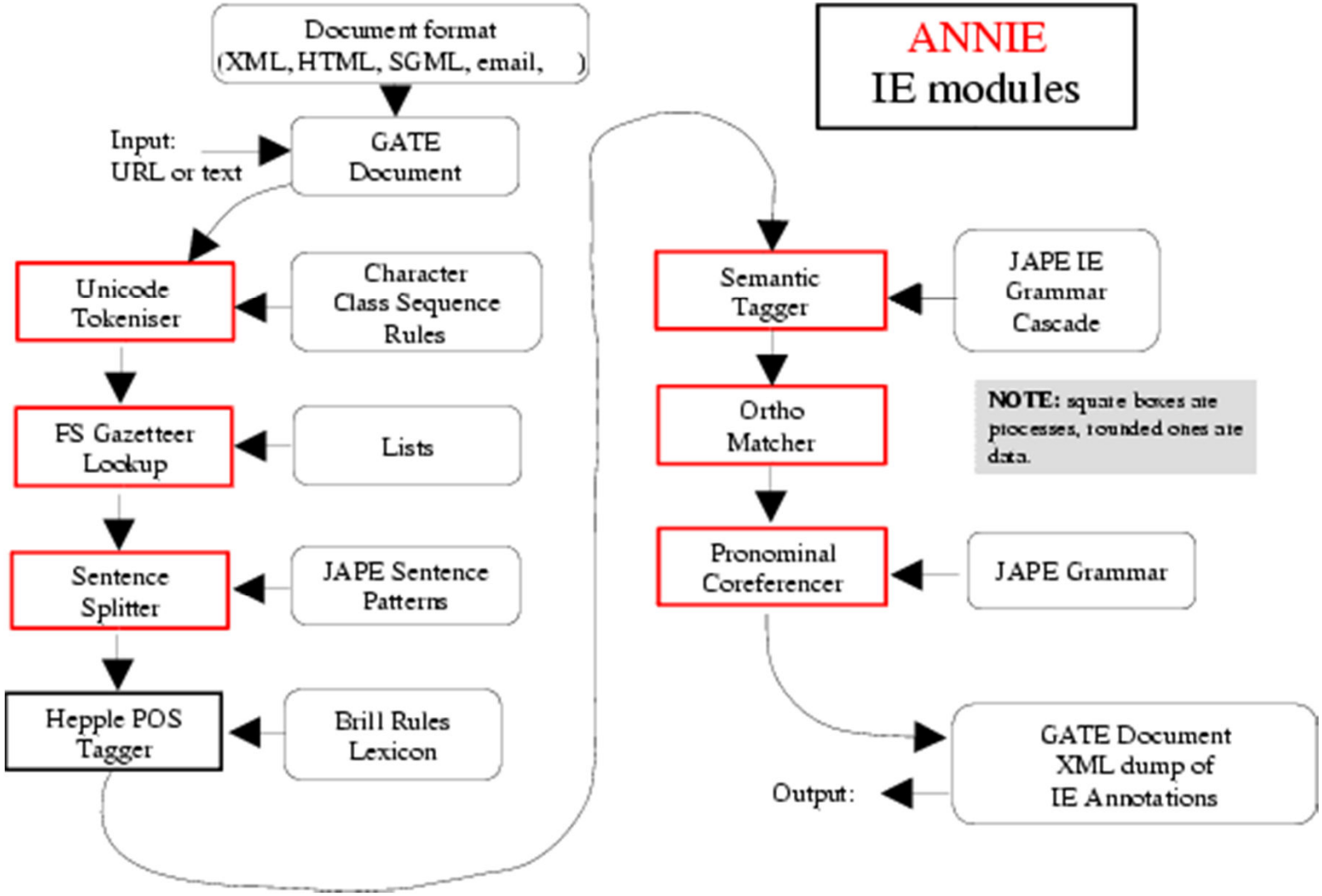
Nearly New Information Extraction

- ANNIE is a ready made collection of PRs that performs IE on unstructured text.
- For those who grew up in the UK, you can think of it as a Blue Peter-style “here's one we made earlier”.
- ANNIE is “nearly new” because
- It was based on an existing IE system, LaSIE
- We rebuilt LaSIE because we decided that people are better than dogs at IE
- Being 10 years old, it's not really new any more

What's in ANNIE?

- The ANNIE application contains a set of core PRs:
- Tokeniser
- Sentence Splitter
- POS tagger
- Gazetteers
- Named entity tagger (JAPE transducer)
- Orthomatcher (orthographic coreference)
- There are also other PRs available in the ANNIE plugin, which are not used in the default application, but can be added if necessary
- NP and VP chunker

Core ANNIE components



Loading and running ANNIE

- Because ANNIE is a ready-made application, we can just load it directly from the menu
- Click the icon from the top GATE menu OR
  File →Ready Made Applications →ANNIE →ANNIE OR
  right-click Applications →Ready Made Applications →ANNIE →ANNIE
- Select "with defaults" if necessary
- Load the hands-on corpus from the "news-texts" directory in the zip file
- Run ANNIE and inspect the annotations
- You should see a mixture of Named Entity annotations (Person, Location etc) and some other linguistic annotations (Token, Sentence etc)

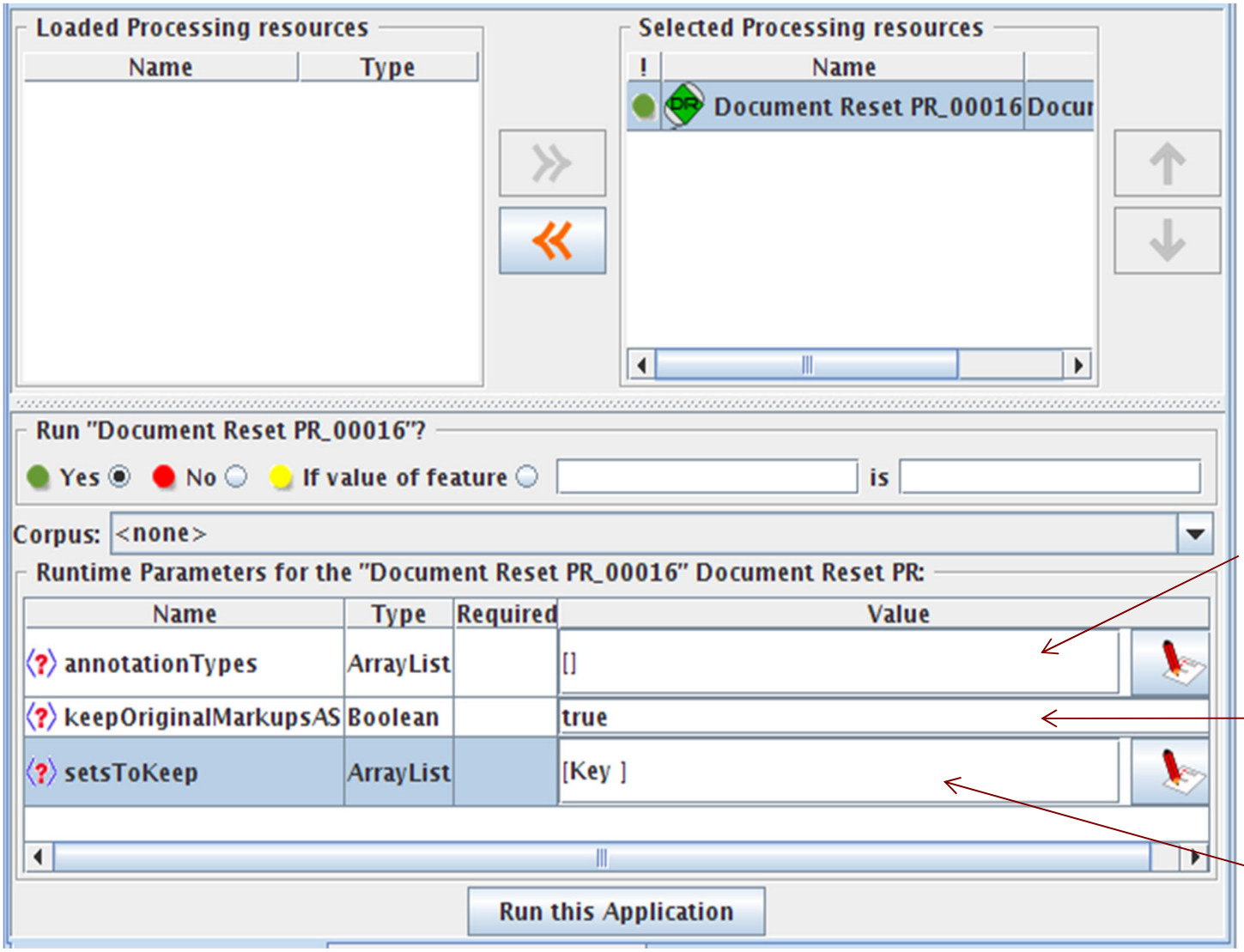
Let's look at the PRs

- Each PR in the ANNIE pipeline creates some new annotations, or modifies existing ones
- Document Reset → removes annotations
- Tokeniser → Token annotations
- Gazetteer → Lookup annotations
- Sentence Splitter → Sentence, Split annotations
- POS tagger → adds category features to Token annotations
- NE transducer → Date, Person, Location, Organisation, Money, Percent annotations
- Orthomatcher → adds match features to NE annotations

Document Reset

- This PR should go at the beginning of (almost) every application you create
- It removes annotations created previously, to prevent duplication if you run an application more than once
- It does not remove the Original Markups set, by default
- You can configure it to keep any other annotation sets you want, or to remove particular annotation types only

Document Reset Parameters



Specify any specific annotations to remove. By default, remove all.

Keep Original Markups set

Keep Key set

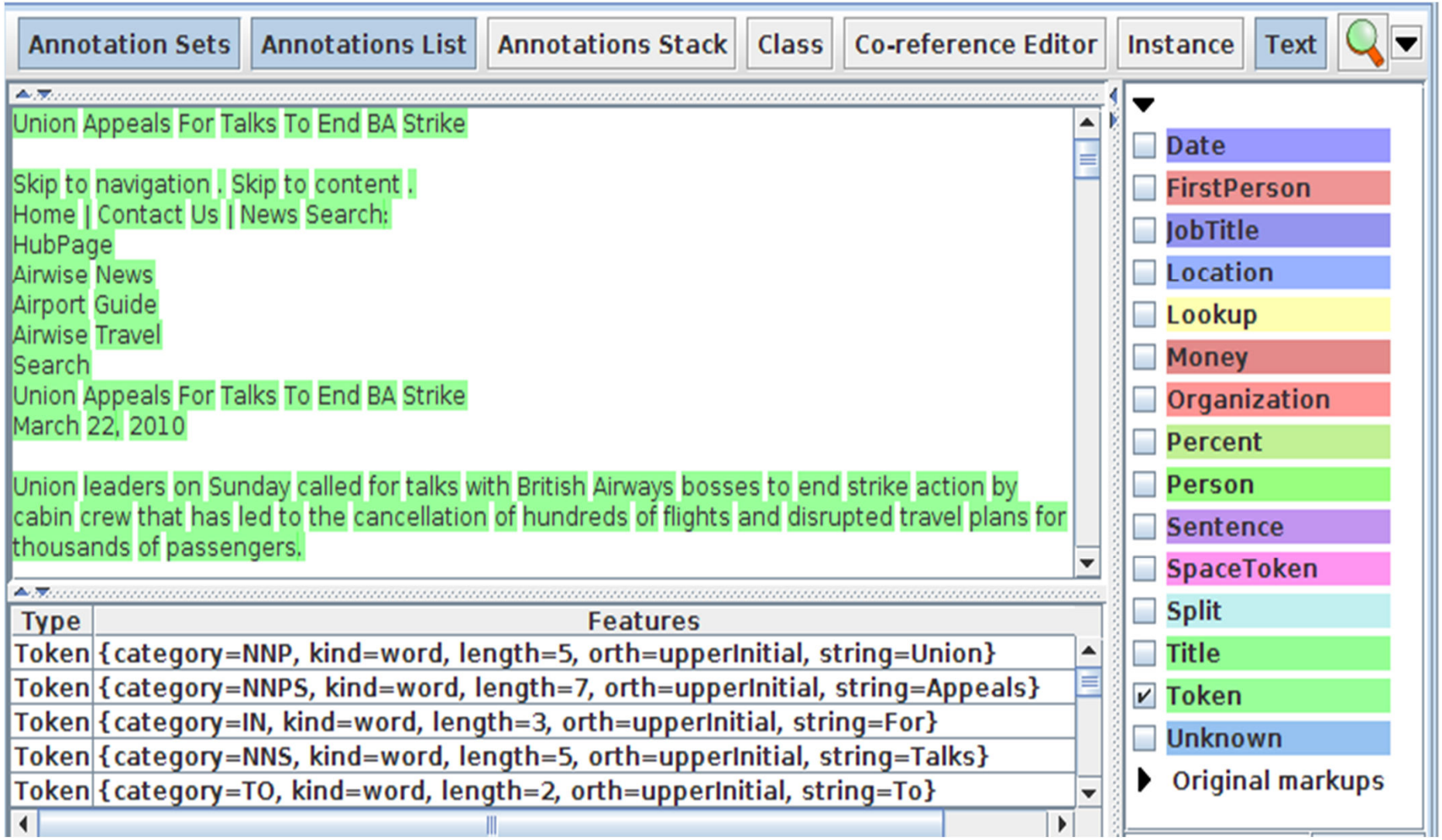
Tokenisation and sentence splitting

Tokeniser

- Tokenisation based on Unicode classes
- Declarative token specification language
- Produces Token and SpaceToken annotations with features orthography and kind
- Length and string features are also produced
- Rule for a lowercase word with initial uppercase letter

```
"UPPERCASE_LETTER" LOWERCASE_LETTER"* >
  Token; orthography=upperInitial; kind=word
```

Document with Tokens



ANNIE English Tokeniser

- The English Tokeniser is a slightly enhanced version of the Unicode tokeniser
- It comprises an additional JAPE transducer which adapts the generic tokeniser output for the POS tagger requirements
- It converts constructs involving apostrophes into more sensible combinations
- don’t → do + n't
- you've → you + 've

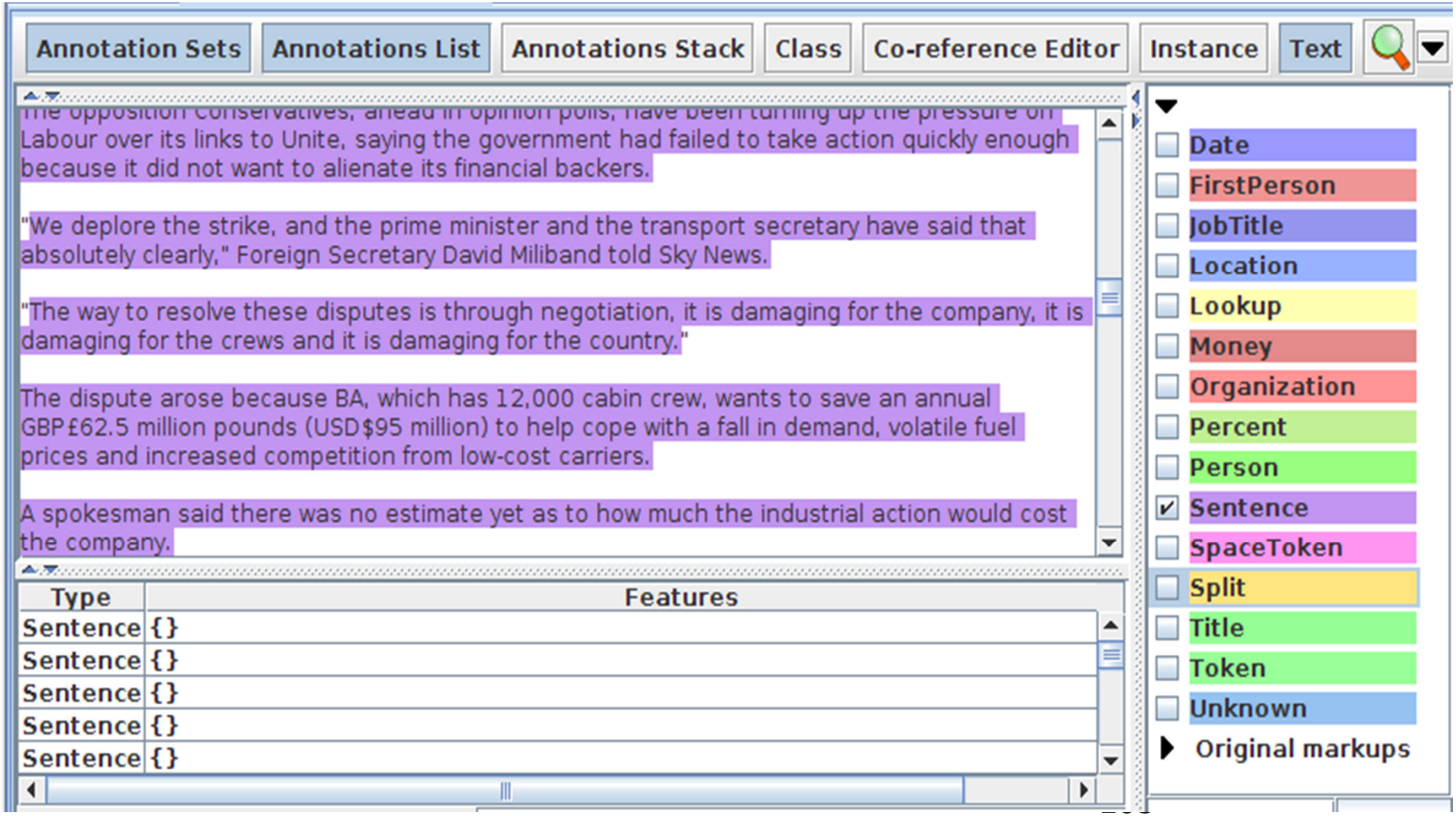
Looking at Tokens

- Tidy up GATE by removing all resources and applications (or just restart GATE)
- Load the news text hands-on corpus
- Create a new application (corpus pipeline)
- Load a Document Reset and an ANNIE English Tokeniser
- Add them (in that order) to the application and run on the corpus
- View the Token and SpaceToken annotations
- What different values of the “kind” feature do you see?

Sentence Splitter

- The default splitter finds sentences based on Tokens
- Creates Sentence annotations and Split annotations on the sentence delimiters
- Uses a gazetteer of abbreviations etc. and a set of JAPE grammars which find sentence delimiters and then annotate sentences and splits
- Load an ANNIE Sentence Splitter PR and add it to your application (at the end)
- Run the application and view the results

Document with Sentences



Sentence splitter variants

- An alternate set of rules can be loaded with the regular sentence splitter
- To do this, reload the sentence splitter using “main-single-nl.jape” instead of “main.jape” as the value of the grammar parameter
- The main difference is the way it handles new lines
- In some cases, you might want a new line to signal a new sentence, e.g. addresses
- In other cases, you might not, e.g. in emails that have been split by the email program
- A regular expression Java-based splitter is also available, called RegEx Sentence Splitter, which is sometimes faster
- This handles new lines in the same way as the default sentence splitter
- See “Further Exercises” to experiment with splitter variants

Shallow lexico-syntactic features

POS tagger

- ANNIE POS tagger is a Java implementation of Brill's transformation based tagger
- Previously known as **Hepple Tagger** (you may find references to this and to **heptag**)
- Trained on WSJ, uses Penn Treebank tagset
- Default ruleset and lexicon can be modified manually (with a little deciphering)
- Adds category feature to Token annotations
- Requires Tokeniser and Sentence Splitter to be run first

Morphological analyser

- Not an integral part of ANNIE, but can be found in the Tools plugin as an “added extra”
- Flex based rules: can be modified by the user (instructions in the User Guide)
- Generates “root” feature on Token annotations
- Requires Tokeniser to be run first
- Requires POS tagger to be run first if the considerPOSTag parameter is set to true

Shallow lexico-syntactic features

- Add an ANNIE POS Tagger to your app
- Add a GATE Morphological Analyser after the POS Tagger
- If this PR is not available, load the Tools plugin first
- Re-run your application
- Examine the features of the Token annotations
- New features of category and root have been added

Gazetteers

Gazetteers

- Gazetteers are plain text files containing lists of names (e.g rivers, cities, people, …)
- The lists are compiled into Finite State Machines
- Each gazetteer has an index file listing all the lists, plus features of each list (majorType, minorType and language)
- Lists can be modified either internally using the Gazetteer Editor, or externally in your favourite editor (note that the new Gazett`eer editor replaces the old GAZE editor you may have seen previously)
- Gazetteers generate Lookup annotations with relevant features corresponding to the list matched
- Lookup annotations are used primarily by the NE transducer
- Various different kinds of gazetteer are available: first we'll look at the default ANNIE gazetteer

Running the ANNIE Gazeteer

- Various different kinds of gazetteer are available: first we'll look at the default ANNIE gazetteer
- Load the ANNIE Gazetteer PR and double click on it in the resource pane to open and see the gazetteers
- Add it to the end of your pipeline
- Re-run the pipeline
- Look for “Lookup” annotations and examine their features

Ontologies in IE

- A typical way to use an ontology in IE is to create a gazetteer from names and labels in the ontology, and use this to annotate entities with IDs (URIs) from the ontology
- GATE includes several tools to help with this, including a basic ontology viewer and editor, several ontology backed gazetteers, and the ability to refer to ontology classes in grammars
- The extra exercises includes an example for you to try, a simple demo application that creates a gazetteer from a SPARQL endpoint, adds entity annotations, and then adds further information to the entities, from the ontology

NE transducers

NE transducer

- Gazetteers can be used to find terms that suggest entities
- However, the entries can often be ambiguous
  - “May Jones” vs “May 2010” vs “May I be excused?”
  - “Mr Parkinson” vs “Parkinson's Disease”
  - “General Motors” vs. “General Smith”
- Handcrafted grammars are used to define patterns over the Lookups and other annotations
- These patterns can help disambiguate, and they can combine different annotations, e.g. Dates can be comprised of day + number + month
- NE transducer consists of a number of grammars written in the JAPE language

ANNIE NE Transducer

- Load an ANNIE NE Transducer PR
- Add it to the end of the application
- Run the application
- Look at the annotations
- You should see some new annotations such as Person, Location, Date etc.
- These will have features showing more specific information (eg what kind of location it is) and the rules that were fired (for ease of debugging)

Modifying ANNIE

Modifying ANNIE

- Typically any new application you want to create will use some or all of the core components from ANNIE
- The tokeniser, sentence splitter and orthomatcher are basically language, domain and application-independent
- The POS tagger is language dependent but domain and application-independent
- You may also require additional PRs (either existing or new ones – e.g. morphological analyser
- The gazetteer lists and JAPE grammars may act as a starting point but will almost certainly need to be modified

ANNIE without defaults

- This option loads all the ANNIE PRs, but enables you to change the location of any of them
- It's useful If you want to use ANNIE but you want to change some of the PRs slightly or replace them with your own modified versions
- Restart GATE or remove all PRs and applications, to tidy up a little
- In your file browser or on the command line, look for plugins/ANNIE/resources/gazetteer in your GATE home directory
- Copy the whole gazetteer directory to a new location on your computer and make some changes to the lists and/or to the index in a text editor
- Load ANNIE from  but select “Without defaults”
- For each PR, select the default option, except for the gazetteer, where you should select your saved gazetteer index file (lists.def)

Multilingual IE

Building a language-specific application

- The following PRs are largely language-independent:
- Unicode tokeniser
- Sentence splitter
- Gazetteer PR (but do localise the lists!)
- Orthomatcher (depending on the nature of the language)
- Other PRs will need to be adapted (e.g. JAPE transducer) or replaced with a language-specific version (e.g. POS tagger)
- This topic is covered in more detail in Track 3 (Advanced IE module)

Useful Multilingual PRs

- Stemmer plugin
- Consists of a set of stemmer PRs for: Danish, Dutch, English, Finnish, French, German, Italian, Norwegian, Portuguese, Russian, Spanish, Swedish
- Requires Tokeniser first (Unicode one is best)
- Language is init-time param, which is one of the above in lower case
- Stanford tools
- Tokeniser, PoS tagger, NER
- TreeTagger
- a language-independent POS tagger which supports English, French, German and Spanish in GATE

Annotation and Evaluation

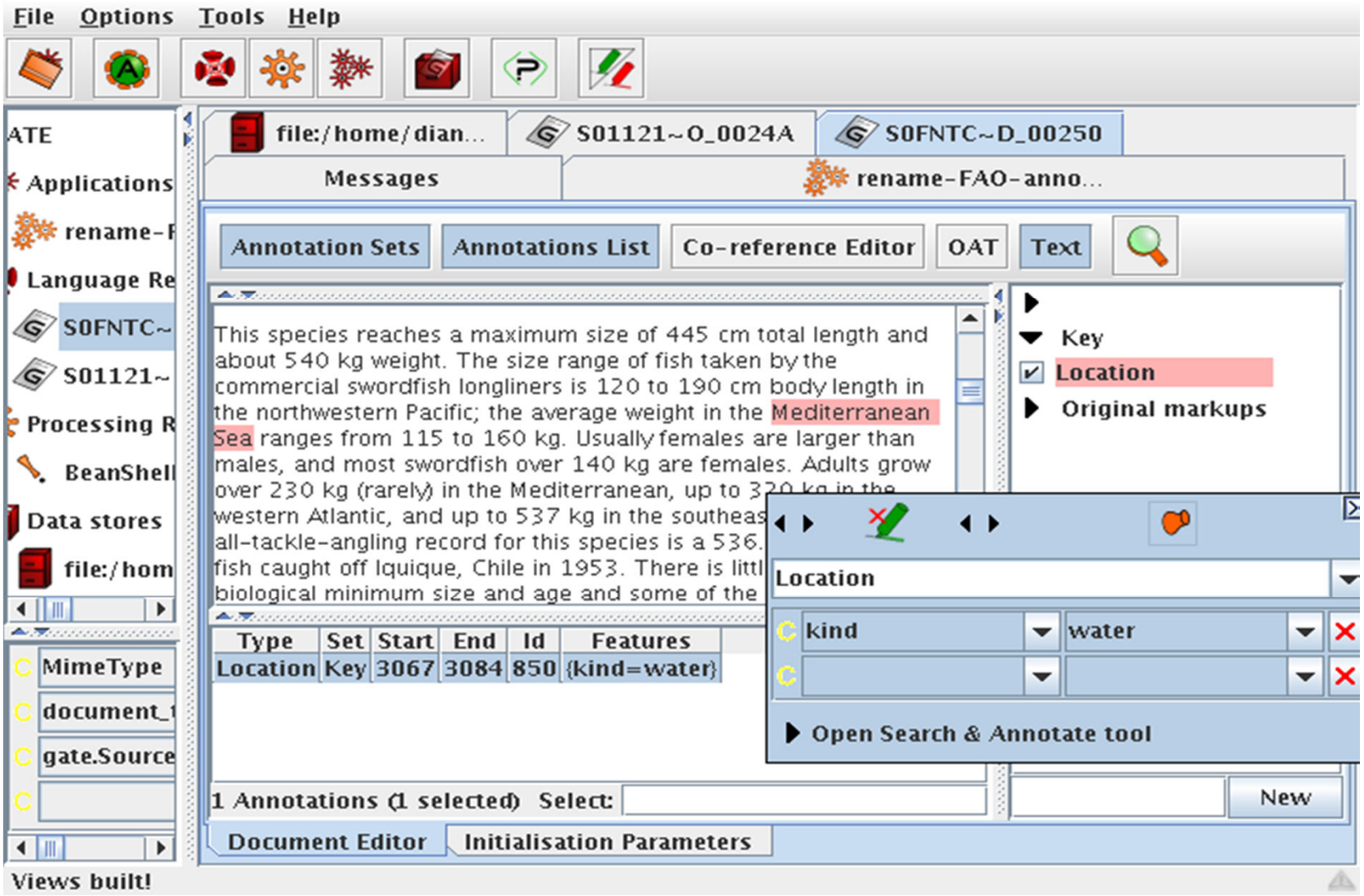
Topics covered

- Defining annotation guidelines
- Recap on manual annotation using the GATE GUI
- Using the GATE evaluation tools

Before you start annotating...

- You need to think about annotation guidelines
- You need to consider what you want to annotate and then to define it appropriately
- With multiple annotators it's essential to have a clear set of guidelines for them to follow
- Consistency of annotation is really important for a proper evaluation

Annotation Editor



Annotation Recap

- Adding annotation sets
- Adding annotations
- Resizing them (changing boundaries)
- Deleting
- Changing highlighting colour
- Setting features and their values
- Using the co-reference editor

Evaluation exercises: preparation

- Restart GATE, or close all documents and PRs to tidy up
- Load the hands on corpus
- Take a look at the annotations.
- There is a set called “Key”. This is a set of annotations against wish we want to evaluate ANNIE. In practice, they could be manual annotations, or annotations from another application.
- Load the ANNIE system with defaults
- **Important:** Change the runtime parameters for the Document Reset PR, adding “Key” to the setsToKeep parameter. This stops the application deleting our Key annotations when we run it.
- Run ANNIE: You should have annotations in the Default set from ANNIE, and in the Key set, against which we can compare them.

AnnotationDiff

- Graphical comparison of 2 sets of annotations
- Visual diff representation, like tkdiff
- Compares one document at a time, one annotation type at a time

Annotations are like squirrels…

Annotation Diff helps with “spot the difference”

Annotation Diff Exercise

- Open the document “ft-airlines-27-jul-2001.xml”
- Open the AnnotationDiff (Tools → Annotation Diff or click the icon
- For the Key set (containing the manual annotations) select **Key** annotation set
- For the Response set (containing annotations from ANNIE) select **Default** annotation set
- Select the **Organization** annotation
- Click on “Compare”
- Scroll down the list, to see correct, partially correct, missing and spurious annotations

Annotation Diff

Annotation Diff Tool

Key doc: ft-airlines-27-jul-200... Key set: Key Type: Organization Weight

Resp. doc: ft-airlines-27-jul-200... Resp. set: [Default set] Features: all some none 1.0 Compare

Start	End	Key	Features	=?	Start	End	
1932	1936	Nats	{}	=	1932	1936	Nats
2456	2460	Nats	{}	=	2456	2460	Nats
2070	2075	LATCC	{}	=	2070	2075	LATCC
1354	1362	Barclays	{}	=	1354	1362	Barclays
1784	1788	Nats	{}	=	1784	1788	Nats
1751	1768	The·Airline·Group	{}	~	1755	1768	Airline·Grou
938	955	The·Airline·Group	{}	~	942	955	Airline·Grou
1669	1686	the·Airline·Group	{}	~	1673	1686	Airline·Grou
2412	2429	The·Airline·Group	{}	~	2416	2429	Airline·Grou
1266	1283	The·Airline·Group	{}	~	1270	1283	Airline·Grou
1052	1068	Monarch·Airlines	{}	~	1030	1068	Britannia·A
2029	2068	London·Area·and·Terminal·Control·Centre	{}	~	2045	2068	Terminal·C
634	640	Labour	{}	-?			
1030	1047	Britannia·Airways	{}	-?			
				?-	2029	2040	London·Are
				?-	2386	2395	Hampshire

			Recall	Precision	F-measure
Correct:	19				
Partially correct:	7	Strict:	0.68	0.68	0.68
Missing:	2	Lenient:	0.93	0.93	0.93
False positives:	2	Average:	0.80	0.80	0.80

10 documents loaded

Statistics Adjudication

Finding Precision, Recall and F-measure

Annotation Diff Tool

Key doc: ft-airlines-27-jul-200... Key set: Key Type: Organization Weight

Resp. doc: ft-airlines-27-jul-200... Resp. set: [Default set] Features: all some none 1.0 Compare

Start	End	Key	Features	=?	Start	End	
1932	1936	Nats	{}	=	1932	1936	Nats
2456	2460	Nats	{}	=	2456	2460	Nats
2070	2075	LATCC	{}	=	2070	2075	LATCC
1354	1362	Barclays	{}	=	1354	1362	Barclays
1784	1788	Nats	{}	=	1784	1788	Nats
1751	1768	The·Airline·Group	{}	~	1755	1768	Airline·Grou
938	955	The·Airline·Group	{}	~	942	955	Airline·Grou
1669	1686	the·Airline·Group	{}	~	1673	1686	Airline·Grou
2412	2429	The·Airline·Group	{}	~	2416	2429	Airline·Grou
1266	1283	The·Airline·Group	{}	~	1270	1283	Airline·Grou
1052	1068	Monarch·Airlines	{}	~	1030	1068	Britannia·A
2029	2068	London·Area·and·Terminal·Control·Centre	{}	~	2045	2068	Terminal·C
634	640	Labour	{}	-?			
1030	1047	Britannia·Airways	{}	-?			
				?-	2029	2040	London·Are
				?-	2386	2395	Hampshire

			Recall	Precision	F-measure
Correct:	19				
Partially correct:	7	Strict:	0.68	0.68	0.68
Missing:	2	Lenient:	0.93	0.93	0.93
False positives:	2	Average:	0.80	0.80	0.80

Statistics Adjudication

10 documents loaded

scores displayed

Annotation Diff defaults to F1

Annotation Diff Tool

Key doc: ft-airlines-27-jul-200... Key set: Key Type: Organization Weight Compare

Resp. doc: ft-airlines-27-jul-200... Resp. set: [Default set] Features: all some none 1.0

Start	End	Key	Features	=?	Start	End	
1932	1936	Nats	{}	=	1932	1936	Nats
2456	2460	Nats	{}	=	2456	2460	Nats
2070	2075	LATCC	{}	=	2070	2075	LATCC
1354	1362	Barclays	{}	=	1354	1362	Barclays
1784	1788	Nats	{}	=	1784	1788	Nats
1751	1768	The·Airline·Group	{}	~	1755	1768	Airline·Grou
938	955	The·Airline·Group	{}	~	942	955	Airline·Grou
1669	1686	the·Airline·Group	{}	~	1673	1686	Airline·Grou
2412	2429	The·Airline·Group	{}	~	2416	2429	Airline·Grou
1266	1283	The·Airline·Group	{}	~	1270	1283	Airline·Grou
1052	1068	Monarch·Airlines	{}	~	1030	1068	Britannia·A
2029	2068	London·Area·and·Terminal·Control·Centre	{}	~	2045	2068	Terminal·C
634	640	Labour	{}	-?			
1030	1047	Britannia·Airways	{}	-?			
				?-	2029	2040	London·Are
				?-	2386	2395	Hampshire

			Recall	Precision	F-measure
Correct:	19				
Partially correct:	7	Strict:	0.68	0.68	0.68
Missing:	2	Lenient:	0.93	0.93	0.93
False positives:	2	Average:	0.80	0.80	0.80

Statistics Adjudication

10 documents loaded

F-measure weight set to 1

Statistics can mean what you want them to....

- How we want to measure partially correct annotations may differ, depending on our goal
- In GATE, there are 3 different ways to measure them
- The most usual way is to consider them to be “half right”
- Average: Strict and lenient scores are averaged (this is the same as counting a half weight for every partially correct annotation)
- Strict: Only perfectly matching annotations are counted as correct
- Lenient: Partially matching annotations are counted as correct. This makes your scores look better :-)

Strict, Lenient and Average

Annotation Diff Tool

Key doc: ft-airlines-27-jul-200... Key set: Key Type: Organization Weight

Resp. doc: ft-airlines-27-jul-200... Resp. set: [Default set] Features: all some none 1.0 Compare

Start	End	Key	Features	=?	Start	End	
1932	1936	Nats	{}	=	1932	1936	Nats
2456	2460	Nats	{}	=	2456	2460	Nats
2070	2075	LATCC	{}	=	2070	2075	LATCC
1354	1362	Barclays	{}	=	1354	1362	Barclays
1784	1788	Nats	{}	=	1784	1788	Nats
1751	1768	The·Airline·Group	{}	~	1755	1768	Airline·Grou
938	955	The·Airline·Group	{}	~	942	955	Airline·Grou
1669	1686	the·Airline·Group	{}	~	1673	1686	Airline·Grou
2412	2429	The·Airline·Group	{}	~	2416	2429	Airline·Grou
1266	1283	The·Airline·Group	{}	~	1270	1283	Airline·Grou
1052	1068	Monarch·Airlines	{}	~	1030	1068	Britannia·A
2029	2068	London·Area·and·Terminal·Control·Centre	{}	~	2045	2068	Terminal·C
634	640	Labour	{}	-?			
1030	1047	Britannia·Airways	{}	-?			
				?-	2029	2040	London·Are
				?-	2386	2395	Hampshire

			Recall	Precision	F-measure
Correct:	19				
Partially correct:	7	Strict:	0.68	0.68	0.68
Missing:	2	Lenient:	0.93	0.93	0.93
False positives:	2	Average:	0.80	0.80	0.80

Statistics Adjudication

10 documents loaded

Comparing the individual annotations

- In the AnnotationDiff, colour codes indicate whether the annotation pair shown are correct, partially correct, missing (false negative) or spurious (false positive)
- You can sort the columns however you like

Comparing the annotations

Annotation Diff Tool

Key doc: ft-airlines-27-jul-200... Key set: Key Type: Organization Weight

Resp. doc: ft-airlines-27-jul-200... Resp. set: [Default set] Features: all some none 1.0 Compare

Start	End	Key	Features	=?	Start	End	
1932	1936	Nats	{}	=	1932	1936	Nats
2456	2460	Nats	{}	=	2456	2460	Nats
2070	2075	LATCC	{}	=	2070	2075	LATCC
1354	1362	Barclays	{}	=	1354	1362	Barclays
1784	1788	Nats	{}	=	1784	1788	Nats
1751	1768	The·Airline·Group	{}	~	1755	1768	Airline·Grou
938	955	The·Airline·Group	{}	~	942	955	Airline·Grou
1669	1686	the·Airline·Group	{}	~	1673	1686	Airline·Grou
2412	2429	The·Airline·Group	{}	~	2416	2429	Airline·Grou
1266	1283	The·Airline·Group	{}	~	1270	1283	Airline·Grou
1052	1068	Monarch·Airlines	{}	~	1030	1068	Britannia·A
2029	2068	London·Area·and·Terminal·Control·Centre	{}	~	2045	2068	Terminal·C
634	640	Labour	{}	-?			
1030	1047	Britannia·Airways	{}	-?			
				?-	2029	2040	London·Are
				?-	2386	2395	Hampshire

			Recall	Precision	F-measure
Correct:	19				
Partially correct:	7	Strict:	0.68	0.68	0.68
Missing:	2	Lenient:	0.93	0.93	0.93
False positives:	2	Average:	0.80	0.80	0.80

Statistics Adjudication

10 documents loaded

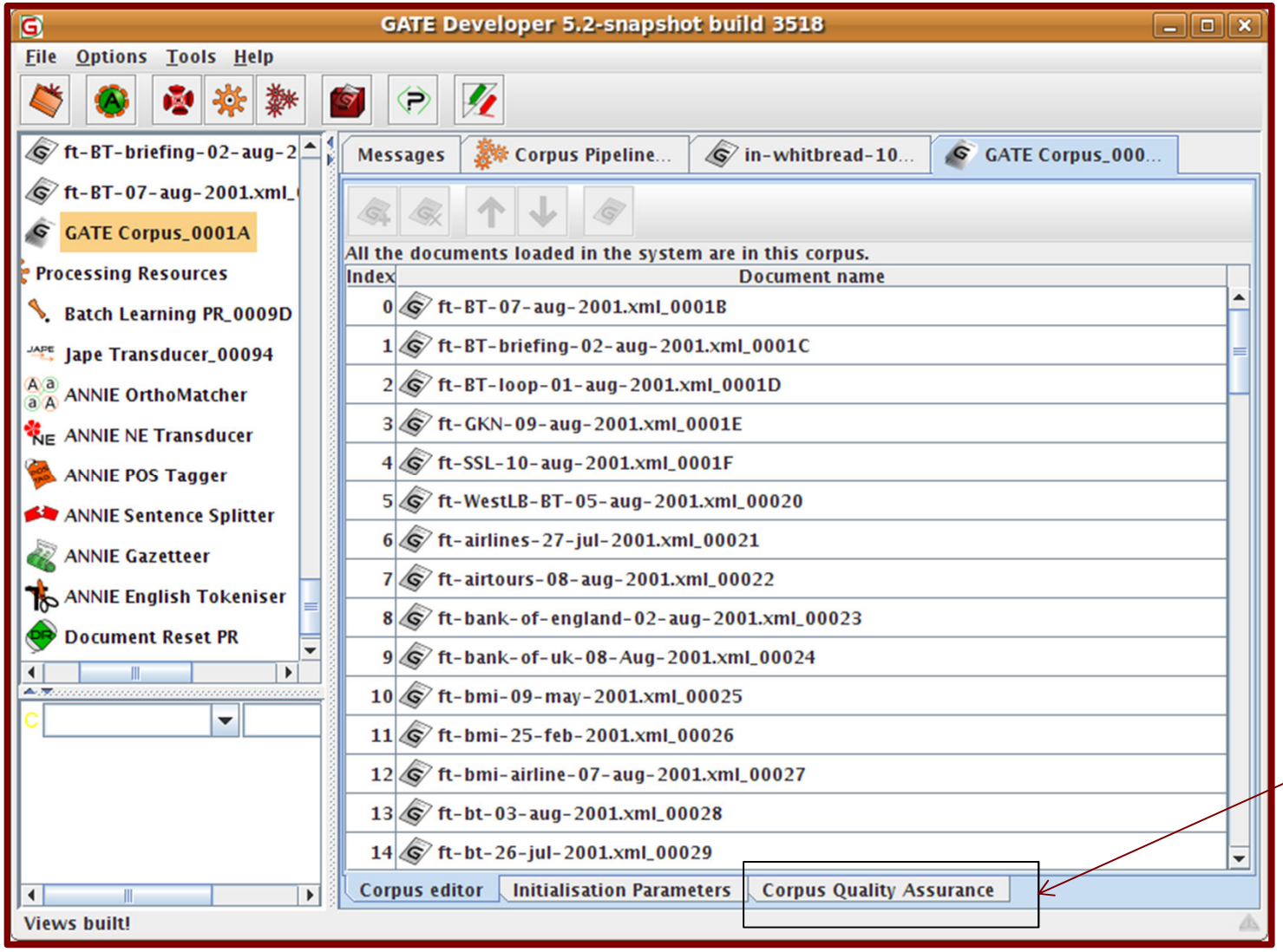
Key annotations

Response annotations

Corpus Quality Assurance

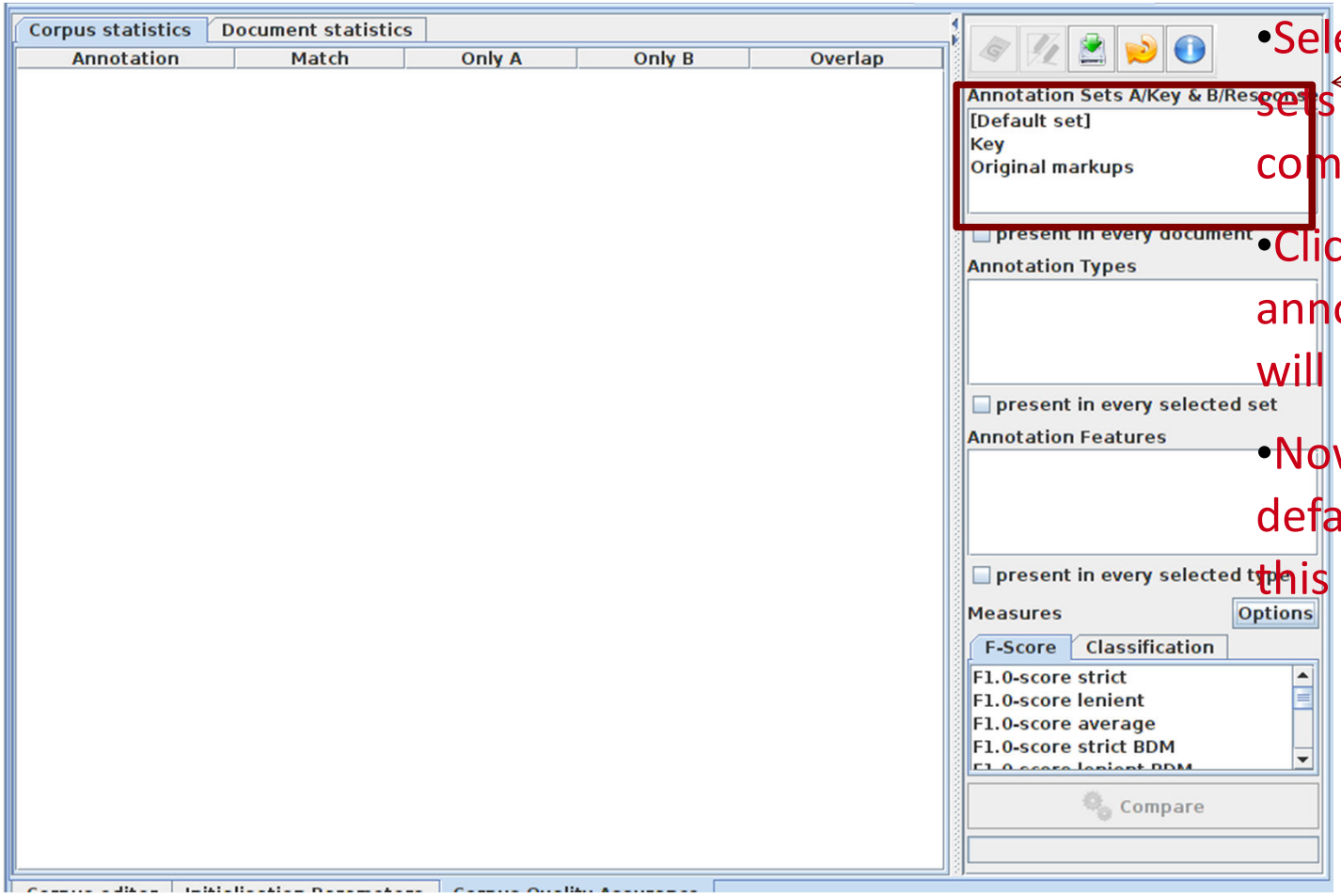
- Corpus Quality Assurance tool extends the Annotation Diff functionality to the entire corpus, rather than on a single document at a time
- It produces statistics both for the corpus as a whole (Corpus statistics tab) and for each document separately (Document statistics tab)
- It compares two annotation sets, but makes no assumptions about which (if either) set is the gold standard. It just labels them A and B.
- This is because it can be used to measure Inter Annotator Agreement (IAA) where there is no concept of “correct” set

Try out Corpus Quality Assurance

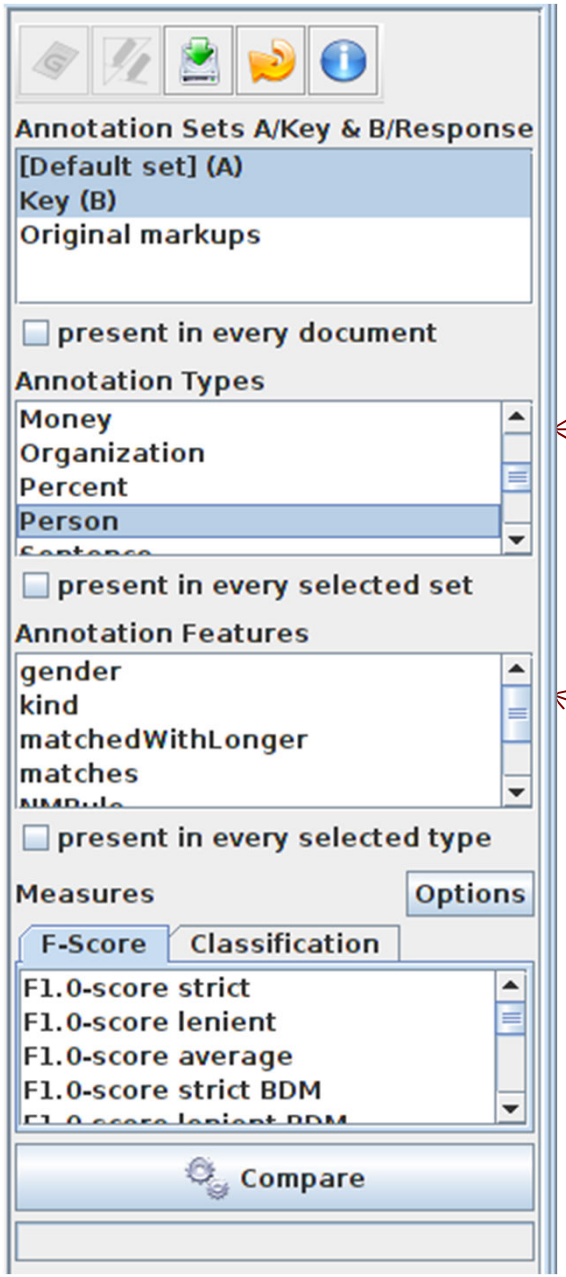


- Open your hands-on corpus and click the Corpus Quality Assurance tab at the bottom of the Display pane.

Select Annotation Sets

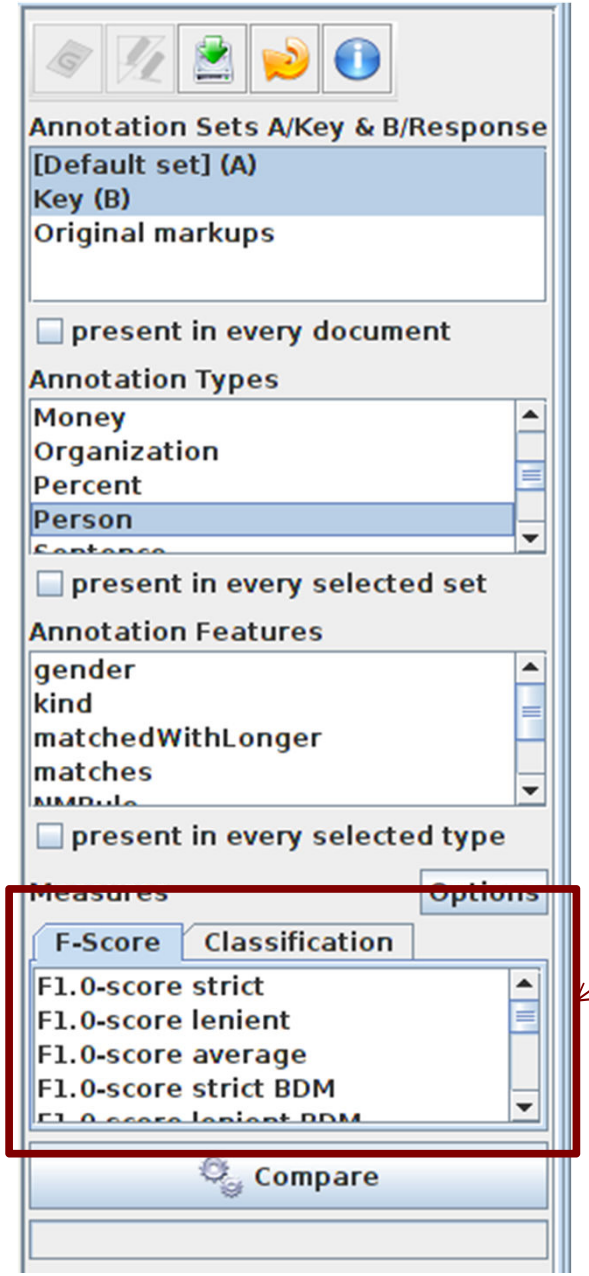


- Select the annotation sets you wish to compare.
- Click on the Key annotation set – this will label it set A.
- Now click on the default annotation set - this will label it set B.



Select Type

- Select the annotation type to compare (suggestion: select Organisation, Person and Location for now)
- Select the features to include (if any – leave unselected for now)
- You can select as many types and features as you want.



Select measure

- In the “Measures” box, select the kind of F score you want “Strict, Lenient, Average” or any combination of them. Suggestion: try just “lenient” at first

- Select Compare

Corpus Statistics Tab

Corpus statistics | Document statistics

Annotation	Match	Only A	Only B	Overlap	Rec.B/A	Prec.B/A	F1-strict
Person	328	26	11	7	0.91	0.95	0.93
Macro summary					0.91	0.95	0.93
Micro summary	328	26	11	7	0.91	0.95	0.93

- Each annotation type is listed separately
- Precision, recall and F measure are given for each
- Two summary rows provide micro and macro averages

Document Statistics Tab

Corpus statistics | Document statistics

Document	Match	Only A	Only B	Overlap	Rec.B/A	Prec.B/A	F1-strict
in-reed-10-aug-2001.xml_00072	10	1	0	0	0.91	1.00	0.95
in-rover-10-aug-2001.xml_00073	3	0	0	0	1.00	1.00	1.00
in-scoot-10-aug-2001.xml_00074	1	0	0	0	1.00	1.00	1.00
in-shell-cirywire-03-aug-2001.xml_00075	7	1	0	0	0.88	1.00	0.93
in-tesco-citywire-07-aug-2001.xml_00076	1	0	0	0	1.00	1.00	1.00
in-whitbread-10-aug-2001.xml_00077	1	0	0	0	1.00	1.00	1.00
Macro summary					0.95	0.95	0.94
Micro summary	328	26	11	7	0.91	0.95	0.93

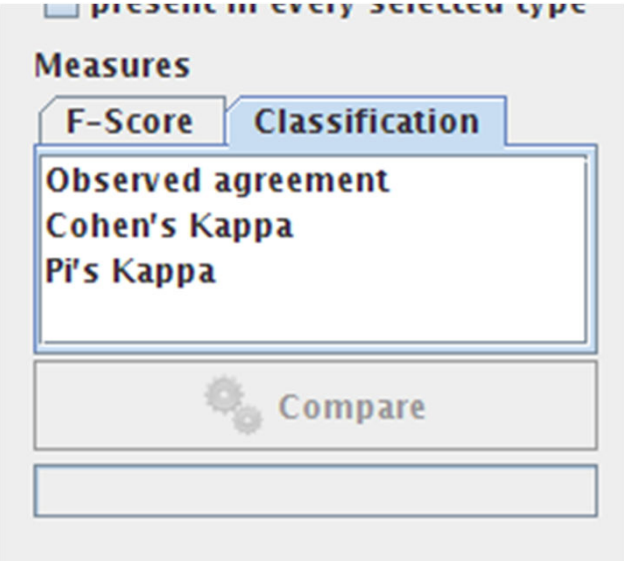
Corpus editor | Initialisation Parameters | Corpus Quality Assurance

- Each document is listed separately
- Precision, recall and F measure are given for each
- Two summary rows provide micro and macro averages

Micro and Macro Averaging

- Micro averaging treats the entire corpus as one big document, for the purposes of calculating precision, recall and F
- Macro averaging takes the average of the rows

Classification Measures



- By default, Corpus Quality Assurance presents the F-measures
- However, classification measures are also available
- These are not suitable for entity extraction tasks

Corpus Quality Assurance PR

- Corpus QA can also be carried out as part of a GATE pipeline, using the Corpus QA PR
- The Corpus QA PR can be found in the tools plugin
- The PR writes out HTML pages, giving the same measures as the Corpus QA viewer
- The Corpus QA PR is executed when a pipeline reaches the last document in the corpus.
- You can set parameters for:
  - Annotation sets to use as key and response
  - Annotation types and features to compare
  - Evaluation metric to use

Corpus Quality Assurance PR

- You must also set the URL of an output directory
- The PR writes HTML pages to this directory, giving the same measures as the Corpus QA viewer:
  - Per-document metrics
  - Corpus and annotation type metrics
- The output HTML is also linked to HTML generated by the Annotation Diff tool for each document
- You can thus use the PR to generate a full evaluation and click through to error reports for each document
- The extra exercises contains an example of running a pipeline with the Corpus QA PR

Summary

- You should now have a basic understanding of:
- what IE is
- how to load and run ANNIE
- what each of the ANNIE components do
- how to modify ANNIE components
- multilingual capabilities of GATE
- Evaluation

End of exercises

Optional advanced material follows

This work was partially supported by the European Union and the European Social Fund through project FuturICT.hu (grant no.: TAMOP-4.2.2.C-11/1/KONV-2012-0013).

Further exercises: sentence splitter variants

Sentence Splitter variants

- Organisations do not span sentence boundaries, according to the rules used to create them.
- Load the default ANNIE and run it on the document in the directory module2-hands-on/universities
- Look at the Organisation annotations
- Now remove the sentence splitter and replace it with the alternate sentence splitter (see slide on Sentence Splitting variants for details)
- Run ANNIE again and look at the Organisation annotations.
- Can you see the difference?
- Can you understand why? If not, have a look at the relevant Sentence annotations.

Further exercises:
an ontology gazetteer

Ontology Gazetteer

- This exercise opens a pre-configured application that contains an ontology based gazetteer, so that you can run it and look at the kind of results produced
- The exercise is not intended to explain ontologies or any of the ontology technologies used, and does not attempt to configure anything. These are covered in the advanced GATE course
- It is intended only to give you a flavour of what is possible
- GATE contains various ontology tools and gazetteers. We will use the Large Knowledge Base Gazeteer
- This is found in the Gazetteer_LKB plugin

Ontology Gazetteer

- You need a working internet connection for this exercise
- Restart GATE, or close all documents and PRs to tidy up
- Using the “File > Restore Application from file” menu, navigate to this directory in your GATE installation:
  - plugins\Gazeteer_LKB\samples\sample_pipelines
- Select and open this application file
  - sample_linked_data_mashup.gapp

Ontology Gazetteer

- The example application file contains a corpus pipeline with three PRs, and a corpus containing a single document from which to load it. In some versions of GATE, the document does not work. If this is the case, try loading and running over http://en.wikipedia.org/wiki/Ricky_Hui
- Open the pipeline and take a look at the order of the PRs
  - The first PR is a Document Reset PR
  - The second is an LKB Gazetteer
- Double click on the LKB Gazetteer in the Processing Resources tree, to see its initialization parameters

Ontology Gazetteer

- The LKB parameter dictionaryPath points to a directory that contains configuration files.
- These tell it where to find an ontology and how to use it. In our case, one of these points to an ontology at http://factforge.net/sparql and another contains a query to retrieve the names of actors from this ontology.
- When initialized, the PR builds a gazetteer from the results of the query. It can be configured to cache this gazetteer locally.
- When run, it will create Lookup annotations from this gazetteer, with features for classes and instances in the ontology.

Ontology Gazetteer

- The third PR is a Semantic Enrichment PR
- Look at its initialization parameters
- The parameter repositoryUrl points to an ontology, in this case the same one as before - FactForge
- Look at its runtime parameters in the pipeline view
  - The parameter annotationTypes contains the single type Lookup
  - The parameter called query contains a query against the ontology
- The query will take ontology identifiers from Lookup annotations, look for their birthplace in FactForge, and add it to the annotation

Ontology Gazetteer

- Run the pipeline over the corpus, and examine the annotations in the single document
- You should see Lookup annotations marking actors. Features are:
  - class, the URI of the class of Actor
  - inst, the URI of this particular actor
  - connections, URI of the actor's birthplace

Further exercises: Quality Assurance PR

Quality Assurance PR

- Corpus QA can also be carried out as part of a GATE pipeline, using the Quality Assurance PR.
- The PR writes out HTML pages, giving the same measures as the Corpus QA viewer.
- This exercise repeats the corpus evaluations from earlier in the tutorial, this time using the Quality Assurance PR
- The Quality Assurance PR can be found in the tools plugin
- Restart GATE, or close all documents and PRs to tidy up
- Load the tools plugin, via the Plugin Management Console

Quality Assurance PR: preparation

- Create a new corpus and load it with the tutorial documents
- Take a look at the annotations.
- There is a set called “Key”. This is a set of annotations against whish we want to evaluate ANNIE. In practice, they could be manual annotations, or annotations from another application.
- Load the ANNIE system with defaults, and open in the viewer
- **Important:** Change the runtime parameters for the Document Reset PR, adding “Key” to the setsToKeep parameter. This stops the application deleting our Key annotations when we run it.
- Create a new Quality Assurance PR
- Create an empty directory somewhere on your computer, into which results will be saved.

Quality Assurance PR

- Add the Quality Assurance PR to the end of the pipeline
- Set parameters for:
  - keyASName set to Key
  - responseASName left blank to use the default set
  - Add the following to the annotationTypes list:
  - Organization
  - Person
  - Location
  - Evaluation metric to use, the “measure” parameter. Choose your preferred measure, e.g. F1_STRICT

Quality Assurance PR

- Set the QA PR's outputFolderUrl to the output directory that you created earlier
- Run the pipeline
- Examine the results in the output directory
- corpus-stats.html shows the corpus statistics
- document-stats.html shows the document statistics, and links to an annotation diff for each document and annotation type

Further exercises: comparing ANNIE, LingPipe and OpenNLP

Comparing ANNIE, LingPipe and OpenNLP

- The idea of this exercise is to run and compare three different IE systems using the Corpus QA tools.
- As well as ANNIE, GATE includes wrappers for the independently developed NLP pipelines, LingPipe and OpenNLP
- All three systems are provided as pre-built applications through the GATE File menu
- Note that this is not a proper evaluation!
  - we are not using a gold standard
  - the three applications may have been built with different sets of guidelines

Comparing ANNIE, LingPipe and OpenNLP

- Close any applications, documents and PRs that you have open in GATE
- Create a new corpus and populate it from the corpus in your tutorial material
- From the File → Ready Made Applications menu, load three applications:
  - ANNIE with defaults
  - LingPipe
  - OpenNLP

Comparing ANNIE, LingPipe and OpenNLP

- We will compare the way in which the three applications create Person, Organization and Location annotations
- For comparison, we will need to put annotations from each application into a different annotation set. We will also need to normalize their names, so that each application creates annotations with exactly the same names
- We will do all of the above by using an Annotation Set Transfer PR at the end of each application. This is in the Tools plugin
- Load the Tools plugin via the Plugin Management Console

ANNIE pipeline

- Create a new Annotation Set Transfer PR, calling it “annie transfer”
- Open the ANNIE application in the viewer
- Add “annie transfer” to the end and set parameters:
  - Set outputASName to “annie”
  - Add the following to the annotationTypes list, to copy these annotations:
- Person
- Organization
- Location
- Select the first PR, the Document Reset PR, and add the following to the setsToKeep parameter list:
  - opennlp
  - lingpipe

LingPipe pipeline

- Create a new Annotation Set Transfer PR, calling it “lingpipe transfer”
- Open the LingPipe application in the viewer
- Add “lingpipe transfer” to the end and set parameters:
  - Set outputASName to “lingpipe”
  - Add the following to the annotationTypes list , to copy and rename these annotations:
- PERSON=Person
- ORGANIZATION=Organization
- LOCATION=Location
- Select the first PR, the Document Reset PR, and add the following to the setsToKeep parameter list:
  - opennlp
  - annie

OpenNLP pipeline

- Create a new Annotation Set Transfer PR, calling it “opennlp transfer”
- Open the OpenNLP application in the viewer
- Add “opennlp transfer” to the end and set parameters:
  - Set outputASName to “opennlp”
  - Add the following to the annotationTypes list , to copy these annotations:
- Person
- Organization
- Location
- Select the first PR, the Document Reset PR, and add the following to the setsToKeep parameter list:
  - annie
  - lingpipe

Comparing ANNIE, LingPipe and OpenNLP

- Run each of the three applications over your corpus
- Open the Corpus QA view, and do pair-wise comparisons of the three annotation sets, for the three annotation types
- Look at the Document statistics tab, and open individual documents that differ
- How do the three applications differ?

More exercises with ANNIE, LingPipe and OpenNLP

- It is possible to mix the different PRs from the three applications, e.g. to replace the tokeniser of one with the tokeniser from another
- This doesn't always work – sometimes there are dependencies not met by equivalent PRs in the other applications
- The GATE documentation for the OpenNLP and LingPipe plugins has some notes on this
- For further exercises, you could try comparing the annotations output by individual PRs from each application
- You could also see what effect mixing PRs from different applications has on the final entity annotations

Thank you for your attention!

This work was partially supported by the European Union and the European Social Fund through project FuturICT.hu (grant no.: TAMOP-4.2.2.C-11/1/KONV-2012-0013).