
Contents

1	Introduction	1
1.1	Motivation	1
1.2	Organization	1
2	Background	2
2.1	Argumentation	2
2.1.1	General Definitions	2
2.1.2	Persuasion	2
2.1.3	Persuasion 2.0	2
2.2	NLP and the DKPro Framework	2
2.2.1	Natural Language Processing	2
2.2.2	UIMA	2
2.2.3	DKPro Core	3
2.2.4	DKPro Text Classification	4
3	The Research Work	7
3.1	Text Corpus	7
3.1.1	Manual Annotation	8
3.1.1.1	Categories in Persuasion	8
3.1.1.2	Examples	9
A	DKPro Core Overview	10
B	Maven	11

CHAPTER 1

Introduction

1.1 Motivation

I was motivated to write a Phd thesis because I did not want to work directly after finishing my study

1.2 Organization

This thesis is organized as follows, ...

The aim of this work is to create a model that predicts if an article comment or a forum post can be classified as "persuasive" or "non-persuasive". We'll first give general definitions about persuasion and argumentation, and how the corpus was annotated. Then, the tool that was used to perform the classification, DKPro Text Classification Framework, will be introduced and its functionalities will be explained. Finally we'll discuss about the algorithm used and the metrics to evaluate our model.

2.1 Argumentation

2.1.1 General Definitions

2.1.2 Persuasion

2.1.3 Persuasion 2.0

2.2 NLP and the DKPro Framework

2.2.1 Natural Language Processing

2.2.2 UIMA

DKPro stands for *Darmstadt Knowledge Processing* [3] and it's a software suite for NLP based on the Apache UIMA Framework. UIMA are software systems that analyse large volumes of unstructured information in order to discover knowledge that is relevant to an end user. An example UIMA application might ingest plain text and identify entities, such as persons, places, organizations; or relations, such as works-for or located-at:

The real power of UIMA is its Analysis Engines (AE) which basically analyse a document and record descriptive attributes. Those descriptive attributes will form the docu-

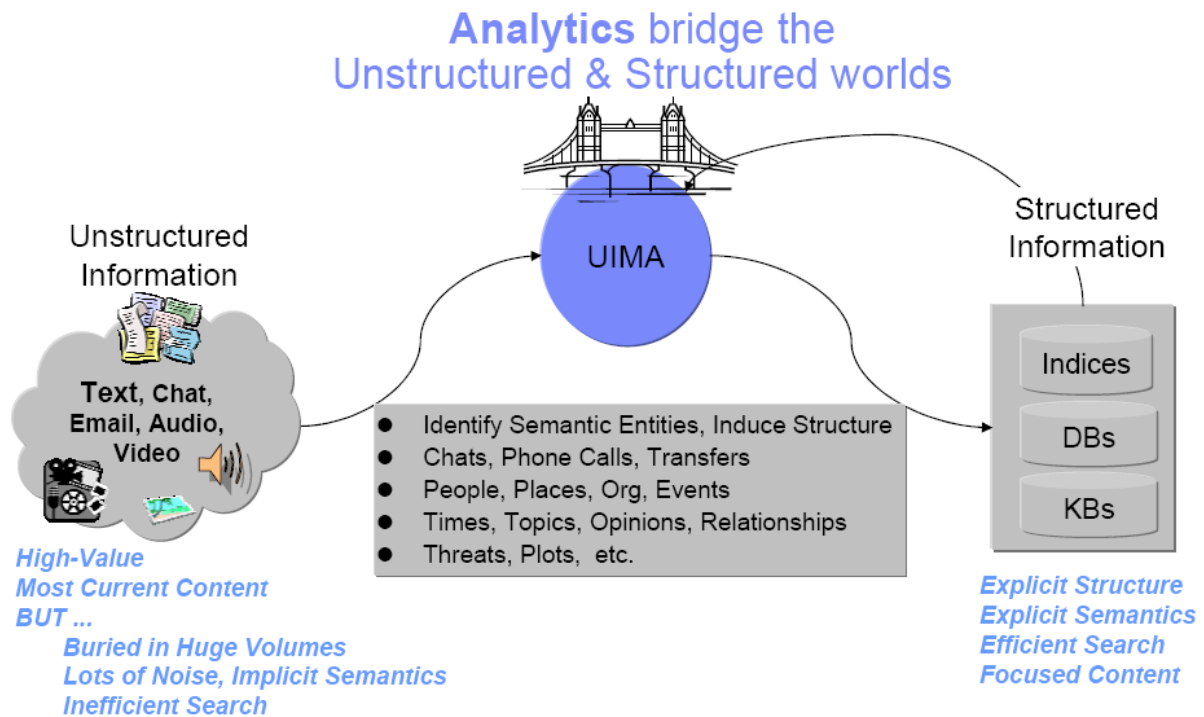


Figure 2.1: UIMA: Order unstructured data, from UIMA website [1]

ment's metadata that will be used for further analysis (such as Text Classification in our case).

GIVE EXAMPLES OF AE HERE AND DKPRO PROC PIPELINE AND CAS ?

Thus, the DKPro software use UIMA's AE in order to collect structured information about textual data.

2.2.3 DKPro Core

Many NLP tools are already freely available in the NLP research community. DKPro Core [2] provides UIMA components wrapping these tools (and some original tools) so they can be used interchangeably in UIMA processing pipelines. The provided components wrap a constantly growing set of stand-of-the-art NLP tools and also include several original components written Java covering a wide range of tasks including: tokenization/segmentation, compound splitting, stemming, part-of-speech tagging, lemmatization, constituency parsing, dependency parsing, named entity recognition, coreference resolution, language identification, spelling correction, grammar checking, and support for reading and writing various file and corpus formats.

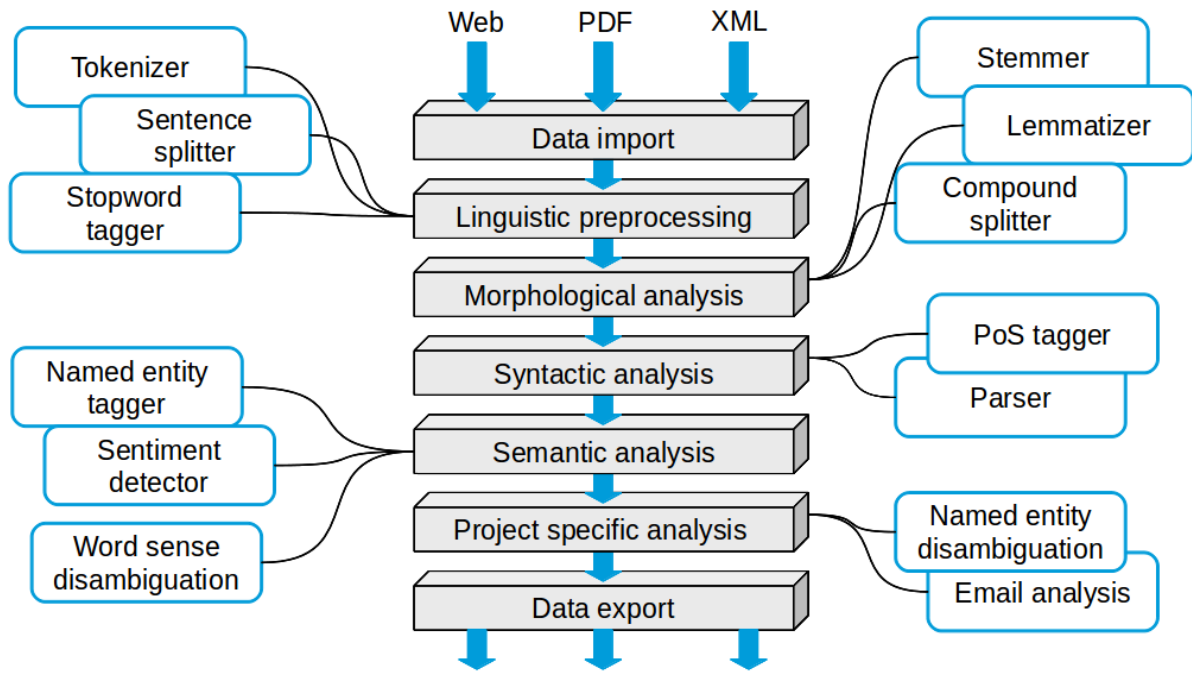


Figure 2.2: DKPro Core Pipeline

Core has several annotators either developed in-house or wrapped¹ from the state-of-the-art NLP libraries. Here is a non exhaustive list:

- **Stanford NLP** - segmentation, la lemmatisation, part of speech...
- **OpenNLP** - machine learning based toolkit for the processing of natural language text: tokenization, sentence segmentation, part-of-speech tagging, named entity extraction, chunking, parsing, and coreference resolution.
- **CleanNLP** - robust NLP components implemented in Java for part-of-speech tagging, dependency parsing, semantic role labelling...

2.2.4 DKPro Text Classification

The aim of DKPro TC (commonly called TC) is to allow the user to apply machine learning algorithms easily on the extracted annotations. This framework was built in order to execute the following tasks:

- Supervised learning Classification. The user should provide annotated textual data.
- Can work atomically on text (word, sentence, paragraph) or on pairs of documents.
- Can perform single-label classification, multi-label classification and regression. DEFS

¹A wrapper function is a subroutine in a software library or a computer program whose main purpose is to call a second subroutine or a system call with little or no additional computation. Source: Wikipedia

Concerning the algorithms used, TC relies on Weka² (*Waikato Environment for Knowledge Analysis*). Developed by the Waikato University in New Zealand, Weka is an open-source Data Mining software written in Java, which makes available to its users not only Machine Learning algorithms but also processing features (attribute selections and transformations) and a user interface with visualization tools. Regularly updated, Weka is one of the main state-of-the-art data mining software used in research.

Weka is integrated to TC thank to one major component: the **feature**. In the code, the feature is usually a class that computes a certain value (ex: length of a post, number of adjectives in a text, ect...) using the annotation provided by the DKPro pipeline. Features can be implemented by polymorphism from the mother-class *FeatureExtractorResource_ImplBase*. The results are then saved in an *ARFF*³ file, which corresponds to Weka's format files.







			
	Single-label	Multi-label	Regression
 Document Mode	<ul style="list-style-type: none"> · Spam Detection · Sentiment Detection 	<ul style="list-style-type: none"> · Text Categorization · Keyphrase Assignment 	<ul style="list-style-type: none"> · Text Readability
 Unit/Sequence Mode	<ul style="list-style-type: none"> · Named Entity Recognition · Part-of-Speech Tagging 	<ul style="list-style-type: none"> · Dialogue Act Tagging 	<ul style="list-style-type: none"> · Word Difficulty
 Pair Mode	<ul style="list-style-type: none"> · Paraphrase Identification · Textual Entailment 	<ul style="list-style-type: none"> · Relation Extraction 	<ul style="list-style-type: none"> · Text Similarity

Figure 2.3: The different usages of DKPro TC

In a nutshell, TC adds 4 more steps to DKPro Core:

- A step where the data are labelled, since we're doing supervised learning. In practice, the label information is extracted by a function of the reader ??WHERE do we define reader ?
- Extraction of the features from the annotations.
- Data Processing and Cross Validation ??DEF
- Report of the results (Accuracy, Macro F-Measure, ect...)

The TC developers give regularly some helpful tutorials on their website⁴.

License and usage

While most DKPro TC modules are available under the Apache Software License (ASL) version 2, there are a few modules that depend on external libraries and are thus licensed under the General Public Licence (GPL).

²<http://www.cs.waikato.ac.nz/ml/weka/>

³An ARFF (Attribute-Relation File Format) file is an ASCII text file that describes a list of instances sharing a set of attributes.

⁴<https://code.google.com/p/dkpro-tc/wiki/DemoExperiments>

The SVN⁵ commits of the different modules are available through Google code and the integration of new functionalities in a project is made possible by Maven (appendix B). It's also possible to use DKPro TC with Groovy which is an object-oriented programming language for the Java platform and destined to be run on a server.

⁵Apache Subversion (often abbreviated SVN, after the command name svn) is a software versioning and revision control system distributed as free software under the Apache License.

The Research Work

In this part, we'll see how the theoretical knowledge in Argumentation Theory and NLP presented were applied to a concrete case study. We'll first see the types of textual data we have, how they were annotated for the supervised learning task and then how features were engineered to perform an automatic classification.

3.1 Text Corpus

The data set used for this study was originally composed of 990 text files. They contain forum posts or articles about 6 different domains related to education, that provoke debate in the American society:

- **homeschooling**: It's the education of children outside the formal settings of public or private schools and is usually undertaken directly by parents or tutors.
- **redshirting**: The practice of postponing entrance into kindergarten of age-eligible children in order to allow extra time for socioemotional, intellectual, or physical growth.
- **prayer in schools**: Debate about whether or not a public school should allow and allocate time and buildings for religious practices.
- **public vs private schools**: Which kind of school offers the best education.
- **mainstreaming**: In the context of education, it's the practice of educating students with special needs in regular classes during specific time periods based on their skills.
- **single sex education**: The practice of conducting education where male and female students attend separate classes or in separate buildings or schools.

The meta-information for each text (id, type of post, domain) is given by the name of the file in itself such as follow:

$\frac{\text{Text}}{1} \frac{\text{Text}}{2} \frac{\text{Text}}{3} ??$ FIND A WAY TO DO IT THIS WAY

Later in the internship, I started to use *XMI* files¹ that contain more information about the post or comment in itself such as the author, the date, ect... ??Annexe with how xmi looks like ?

3.1.1 Manual Annotation

As mentioned before, the classification we want to perform is a supervised learning problem since it wants to imitate the human decision on judging if a post is persuasive or not. An annotation guideline was written by Ivan Habernal [4] in order for the annotators to understand the task. In this section, we'll discuss about the general ideas of this guideline.

3.1.1.1 Categories in Persuasion

The task: Distinguish, whether the comment is persuasive regarding the discussed topic. The key question to answer is: *Does the author intend to convince us clearly about his/her attitude or opinion towards the topic?* If the answer is yes, we classify the comment as persuasive. There are two main categories in this task, namely **P1:Persuasive** and **P2:Non-persuasive**. The second category is further divided into more categories, that basically cover the various phenomena that may be encountered in the data.

However, It is not necessary to categorize the data exactly into one of the categories under **P2:Non-persuasive**. For example, a particular text may be both off-topic and out-of-context; in that case, choose either of these categories.

Remember: we are mainly interested in finding the **P1:Persuasive** documents that represent on-topic texts with intentions to persuade and convince the readers.

Quick overview of possible distinct categories:

P1: Persuasive

P2: Non-persuasive

P2.1: Out-of-context or reaction to other comment

P2.3: Off-topic

P2.4: Personal worries

P2.5: Story-sharing without intentions to persuade

P2.7: Impossible to decide about persuasiveness without deep background knowledge

While the annotation phase, the annotators were charged of determining if a text was in the P1 class and if not they had to specify to which of the seven P2 subclasses it belongs to.

¹The XML Metadata Interchange (XMI) is an Object Management Group (OMG) standard for exchanging metadata information via Extensible Markup Language (XML).

3.1.1.2 Examples

Clearly belongs to P1

#2013 (forumpost, single-sex-education) School should be co-ed. Some children are awkward with others of their own gender. For example, certain girls who are tom-boys might not be comfortable in a room full of girls. The mixed genders are preparing kids for the real world, where things are not segregated.

In #2013, the author states that schools should not be single-sex, also provides reasons why he/she thinks so.

APPENDIX A

DKPro Core Overview

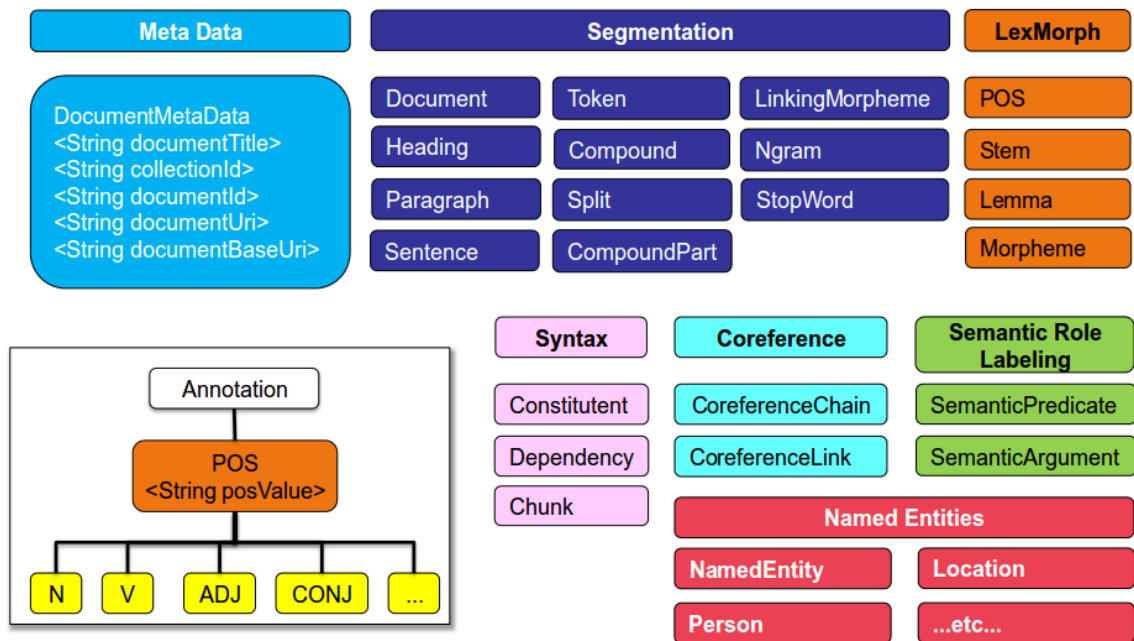


Figure A.1: DKPro Type System

APPENDIX B

Maven

TODO: Explain the functionalities of Maven.

Glossary

mathematics Mathematics is what mathematicians do. 3

Bibliography

- [1] Overview and setup, uima.apache.org, 2006.
- [2] Richard Eckart de Castilho and Iryna Gurevych. A broad-coverage collection of portable nlp components for building shareable analysis pipelines. In *Proceedings of the Workshop on Open Infrastructures and Analysis Frameworks for HLT (OIAF4HLT) at COLING 2014*, pages 1–11, August 2014.
- [3] Iryna Gurevych, Max Mühlhäuser, Christof Müller, Jürgen Steimle, Markus Weimer, and Torsten Zesch. Darmstadt knowledge processing repository based on uima. In *Proceedings of the First Workshop on Unstructured Information Management Architecture at Biannual Conference of the Society for Computational Linguistics and Language Technology*, Tübingen, Germany, April 2007.
- [4] Ivan Habernal, Judith Eckle-Kohler, and Iryna Gurevych. Argumentation mining on the web from information seeking perspective. In *Frontiers and Connections between Argumentation Theory and Natural Language Processing*, page (to appear), July 2014.