COCKPIT

FP7-248222

Citizens Collaboration and Co-Creation in Public Service Delivery

Deliverable D2.1.1

Opinion Mining Tools 1st version

Editor(s):	Kostas Giannakakis
Responsible Partner:	ATC, INTRASOFT International S.A.
Status-Version:	Final – v0.4
Date:	30/06/2011
EC Distribution:	Restricted to other programme participants (including the Commission Services)

Project Number:	FP7-248222
Project Title:	COCKPIT

Title of Deliverable:	Opinion Mining Tools 1st version
Date of Delivery to the EC:	30/06/2011

Workpackage responsible for the Deliverable:	WP2 – Citizens' Opinion Mining and Deliberation
Editor(s):	Kostas Giannakakis (ATC)
Contributor(s):	ATC, INTRASOFT
Reviewer(s):	Ivan Ficano (ENG)
Approved by:	All Partners

Abstract:	This report forms a complementary documentation to the source code files that are delivered as part of this Deliverable which is of type 'Prototype'. It briefly describes the Opinion Mining software environment (1st version) for performing Web 2.0 content collection and sentiment analysis.
Keyword List:	opinion mining, Web 2.0, crawler, sentiment analysis

Document Description

Document Revision History

Version	*Date*	*Modifications Introduced*	
		Modification Reason	*Modified by*
v0.1	20/06/2011	Documentation of the report that is accompanied to the source code files supplied in the context of this deliverable that is of type 'Prototype'.	ATC
v0.2	28/06/2011	Delivery of final version of source code files.	INTRASOFT, ATC
v0.3	28/06/2011	Internal review of documentation and verification of supplied source code files. Provision of comments	ENG
v0.4	30/06/2011	Preparation of final version. Address of reviewing comments and final formatting.	INTRASOFT

Table of Contents

Table of Figures

Executive Summary

This is a supplemenatry document that describes all the material needed for the realisation of the 1st version of the Opinion Mining software environment for performing Web 2.0 content collection and sentiment analysis. A second and final release of this software component will be provided as part of Deliverable D2.1.2 before the initiation of the 2nd cycle of the piloting phase in M29.

1 Introduction

1.1 Purpose and Scope

With respect to the overall WP2 progress plan, the document at hand describes the Opinion Mining software environment for performing Web 2.0 content collection and sentiment analysis.

1.2 Structure of the Document

This documentation consists of the following sections:

- Section 2 describes the combined architecture of the Opinion Mining Tools and the Citizens' Deliberative Engagement Platform.
- Section 3 describes the opinion mining process. The components of the system are orchestrated in a complex workflow that assist users to collect data relevant to a service, build a trained model and then use it to classify new opinions.
- Section 4 describes in detail the software modules of the Opinion Mining Tools.
- Section 5 describes the database schemas used by the tools.

2 Software Architecture

The Opinion Mining Tools are part of a broader system that also includes the Citizens' Deliberative Engagement Platform. These two components are tightly interconnected and share common resources. Their combined architecture is depicted in the figure below.

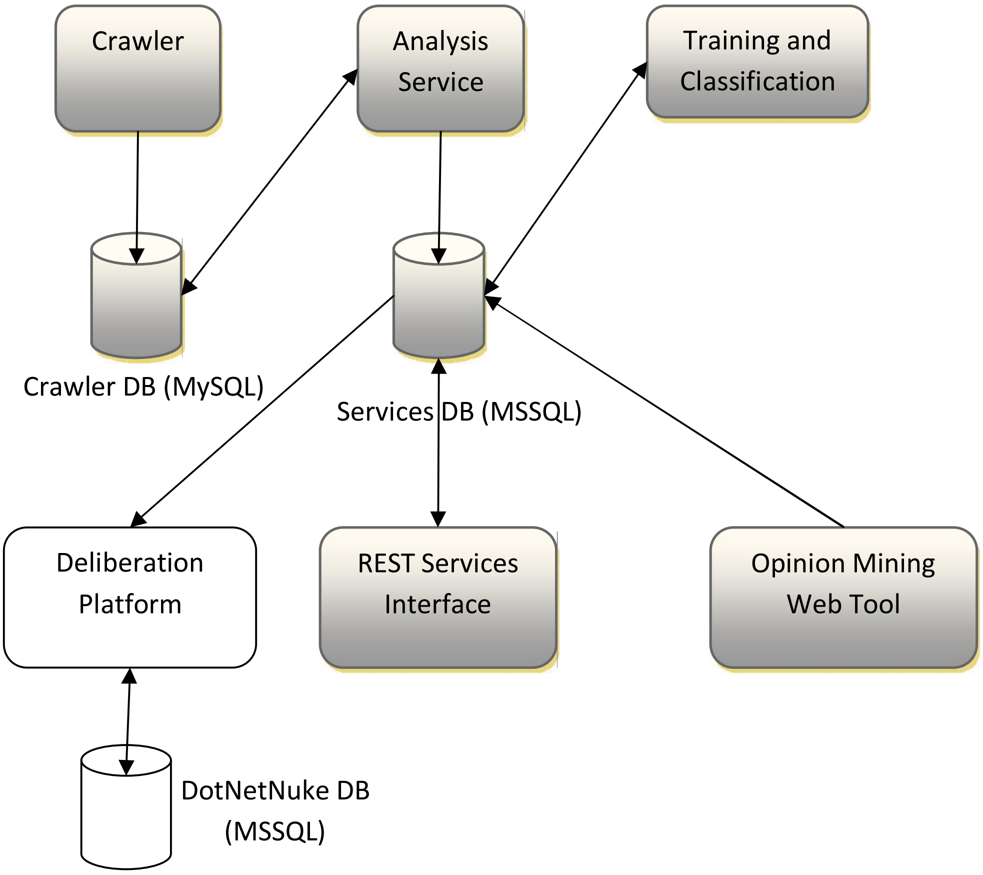


Figure 1: Opinion Mining Tools and Deliberative Engagement Platform Architecture

The Opinion Mining Tools consist of the following components:

- Crawler
- Analysis Service
- Opinion Mining Web Tool
- Training and Classification Component
- REST Services Interface

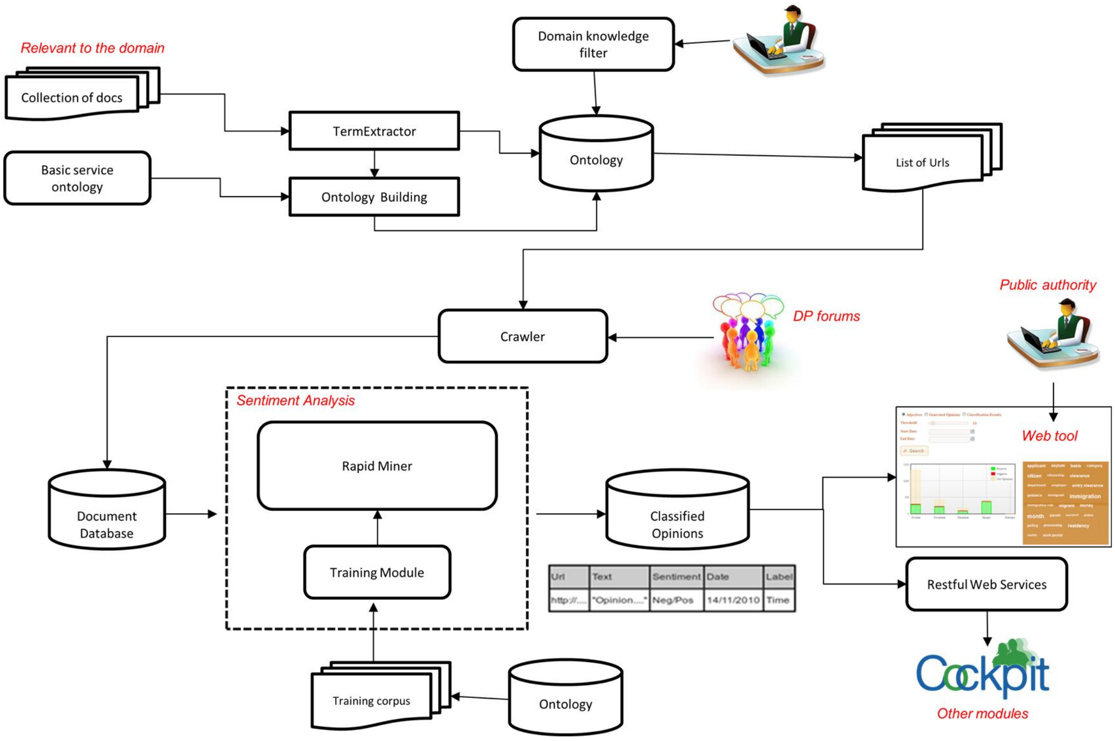
,utilising the following databases:

- Crawler Database (MySQL): This is the database that holds the text crawled from the web. For each service there is a different crawler database employed.
- Services Database (MSSQL): This is a central database that maintains

service specific information such as crawled documents, analysis data, training and classification information, service description, associated polls, votes and comments.

3 Opinion Mining Process

The following figure depicts the conceptual architecture of the Opinion Mining System.



Url	Text	Sentiment	Date	Label
http://....	"Opinion...."	Neg/Pos	14/11/2010	Time

Figure 2: Opinion Mining conceptual architecture

The Opinion Mining process is performed in two phases:

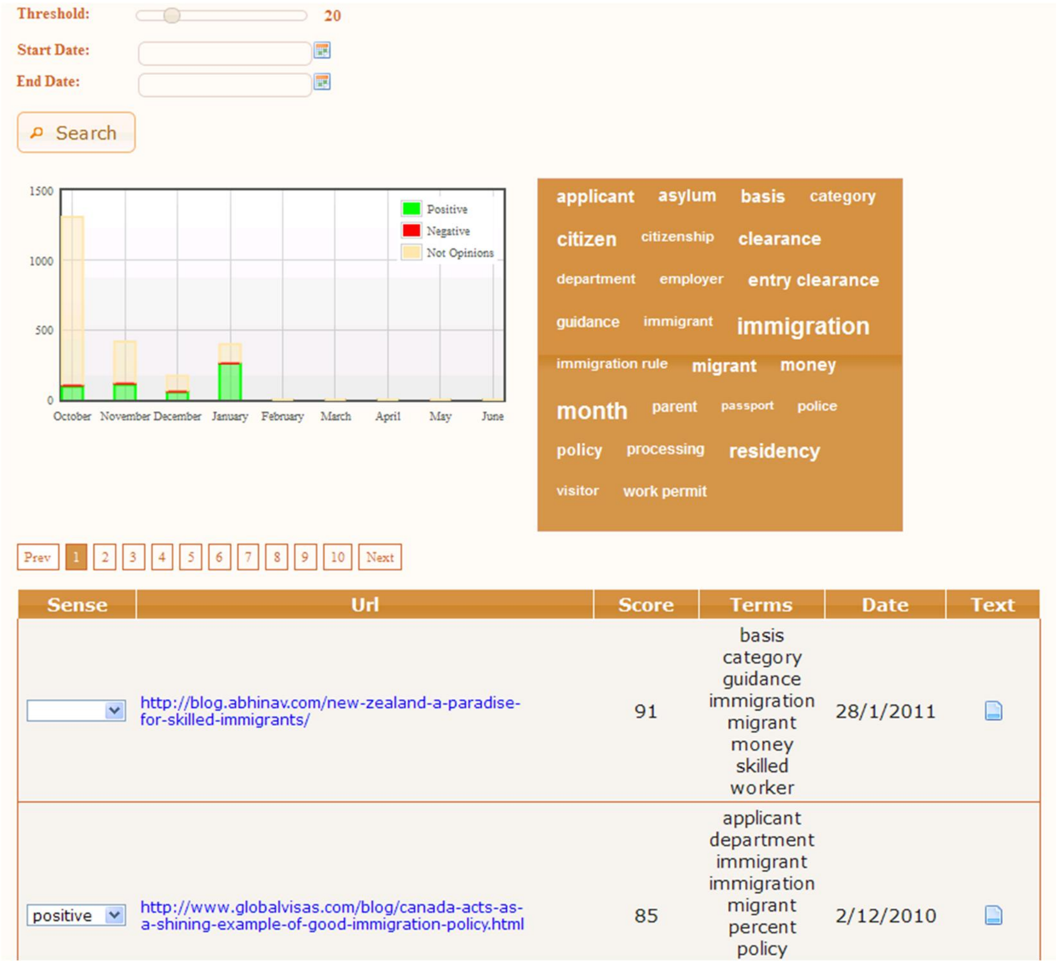
- Training phase: takes as input a list of relevant URLs and the service ontology, and delivers a trained model.
- Runtime phase: takes as input a list of relevant URLs and the trained model, and constantly monitors the Web 2.0 sources for new opinions, which are automatically classified to as positive or negative. The results are delivered through a web tool and/or a RESTful services interface.

Both of those processes are described briefly in the following sub-sections.

3.1 Training phase

The training phase starts with the COCKPIT Crawler being configured to collect documents from URLs relevant to a service. For each service, a different instance of the COCKPIT Crawler is set up. The collected documents are analysed and a score is attached to them. This score is calculated based on the number of adjectives and the ontology terms found in the text, and gives an indication whether they present an opinion regarding the service in question.

After a considerable amount of documents has been collected, the users are asked to go through them and select at least 100 negative and positive opinions out of them. The selection is aided by the Opinion Mining Web Tool, which presents all collected documents is a tabular form and can sort them based on their score.



Sense	Url	Score	Terms	Date	Text
	http://blog.abhinav.com/new-zealand-a-paradise-for-skilled-immigrants/	91	basis category guidance immigration migrant money skilled worker	28/1/2011	🗎
positive	http://www.globalvisas.com/blog/canada-acts-as-a-shining-example-of-good-immigration-policy.html	85	applicant department immigrant immigration migrant percent policy	2/12/2010	🗎

Figure 3: Opinion Mining Web Tool

When an adequate number of opinions are collected, the web tool allows the user to create the trained model. The trained model is created with Rapid Miner and is stored, so that it can later be used in the running phase.

3.2 Runtime phase

The running phase utilises the trained model created during the training phase in order to automatically classify new opinions. The crawler service is still active and collecting new documents. A classification service uses Rapid Miner to classify the new opinions and store the results in a database.

A user can monitor the results through the Opinion Mining Web Tool. The following information is offered:

- Total number of documents referring to a service collected
- Percentage of positive opinions
- Percentage of negative opinions

A time range of interest can be defined (start and stop date). Also, the text of each individual document can be reviewed.

The results are also made available through the RESTful Web Services Interface.

4 Software Modules

4.1 Crawler

Technology	Java
Type	Service
Description	The crawler is based on the open-source sync3crawler. The original crawler has been extended in order to better support multi-lingual texts. The crawler accepts as input a list of RSS feeds and constantly monitors them for new data. When new data are found the crawler fetches the content, cleans the HTLM tags (using boilerpipe library) and stores them into a MySQL database. For each service a different instance of the crawler and the database are deployed.
Source Code	CockpitCrawler.zip

4.2 Analysis Service

Technology	.NET
Type	Service
Description	This is a service that runs periodically and collects new data from the crawler databases. The new data are analyzed and inserted into the Services Database. The data analysis involves searching for relevant terms of a service and adjectives inside the cleaned text. This information is utilized by the Opinion Mining Web Tool in order to provide suggestions to the user during the training phase.
Source Code	AnalysisService.zip

4.3 Opinion Mining Web Tool

Technology	ASP .NET
Type	Web Application
Description	The Opinion Mining Web Tool is a Web Application that is integrated into the SE tool and assists users to: • Configure the Opinion Mining Tool by providing the service related terms and URLs. • Select negative and positive opinions from the crawled documents. The Opinion Mining Web Tool assists the users to perform this task by providing suggestions. • Initiate the Training and Classification phases

	• Monitor the results of the Classification phase The Web Tool for a specific service is available at http://paris.atc.gr/OpinionMining/?service=1
Source Code	OpinionMining.zip

4.4 Training and Classification Component

Technology	RapidMiner
Type	Application
Description	The Training and Classification Component is based on RapidMiner Technology. In the **Training Phase** a set of negative and positive opinions that were selected by the users are read from the Services Database and a classification model is created. The classification model is used during the **Classification Phase** in order to provide polarity predictions for the new documents inserted into the Services Database.
Source Code	TrainingClassification.zip

4.5 REST Services Interface

Technology	ASP .NET
Type	Web Application
Description	This application implements the REST Services Interface for the Deliberation Platform and Opinion Mining Tool. This interface interconnects the Opinion Mining Tool and the Deliberation Platform with other components of the COCKPIT System. The home page of the services is available at http://paris.atc.gr/cockpit-services
Source Code	CockpitServices.zip

5 Databases

5.1 Crawler Database

Technology	MySQL
Description	This is the database that holds the text crawled from the web. For each service there is a different crawler database instance deployed.
Schema	CockPitCrawler.sql

5.2 Services Database

Technology	Microsoft SQL Server
Description	This is a central database that maintains all information including the services. For each service the corresponding crawled documents, analysis, training and classification information, service description, associated poll, votes and comments are stored.
Schema	Cockpit.sql