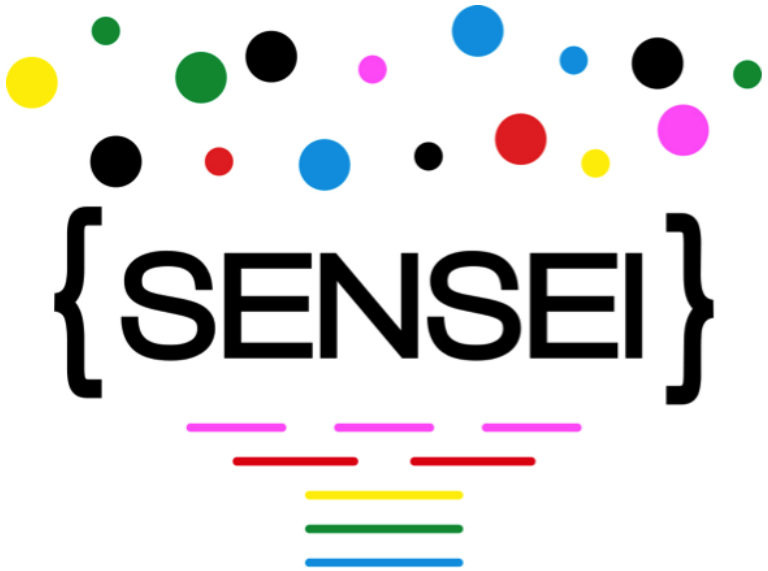

D4.2 – The SENSEI Discourse Analysis Tools, 2

Document Number	D4.2
Document Title	The SENSEI Discourse Analysis Tools, 2
Version	1.9
Status	Final
Workpackage	WP4
Deliverable Type	Report
Contractual Date of Delivery	31.10.2015
Actual Date of Delivery	30.10.2015
Responsible Unit	UESSEX
Keyword List	discourse parsing, event/temporal structure, argumentation structure, intra/inter document coreference
Dissemination level	PU

Editor

Mijail Kabadjov	(University of Essex, UESSEX)
Evgeny A. Stepanov	(University of Trento, UNITN)

Contributors

Evgeny A. Stepanov	(University of Trento, UNITN)
Fabio Celli	(University of Trento, UNITN)
Shammur A. Chowdhury	(University of Trento, UNITN)
Benoit Favre	(University of Aix-Marseille, AMU)
Adam Funk	(University of Sheffield, USFD)
Mijail Kabadjov	(University of Essex, UESSEX)
Udo Kruschwitz	(University of Essex, UESSEX)
Massimo Poesio	(University of Essex, UESSEX)

SENSEI Coordinator

Prof. Giuseppe Riccardi
Department of Information Engineering and Computer Science
University of Trento, Italy
giuseppe.riccardi@unitn.it

Document change history

Version	Date	Status	Author (Unit)	Description
0.1	23/07/2015	Draft	M. Kabadjov, M. Poesio, U. Kruschwitz (UESSEX)	Outline.
0.2	19/08/2015	Draft	F. Celli (UNITN)	Section 6.1 added.
0.3	31/08/2015	Draft	M. Kabadjov (UESSEX)	Sections 4.1, 6.2 added.
0.4	31/08/2015	Draft	E. Stepanov, S. A. Chowdhury (UNITN)	Section 2.3, 2.2, 2.1 added.
0.5	31/08/2015	Draft	A. Funk (USFD)	Section 3 added.
0.6	31/08/2015	Draft	B. Favre (AMU)	Section 4.1.1 added.
0.7	11/09/2015	Draft	M. Kabadjov (UESSEX)	Section 4.1 revised and extended, Sections 1 and 7 added.
0.8	14/09/2015	Draft	M. Kabadjov (UESSEX)	Sections 1 and 7 updated.
0.9	15/09/2015	Draft	F. Celli (UNITN)	Section 6.1 updated.
1.0	16/09/2015	Draft	M. Kabadjov (UESSEX)	Section 1 updated.
1.1	28/09/2015	Draft	A. Funk (USFD)	Section 1 updated on task 4.2.
1.2	04/10/2015	Draft	M. Kabadjov (UESSEX)	Sections 4.2, 5, Executive Summary added, 1 updated.
1.3	09/10/2015	Draft	E. Chiarani (UNITN)	Quality check completed.
1.4	16/10/2015	Draft	E. Stepanov (UNITN), A. Funk, B. Favre, M. Kabadjov	Various sections revised addressing internal review and quality check.
1.5	17/10/2015	Draft	E. Stepanov, S.A. Chowdhury (UNITN)	Section 2 revised and updated.
1.6	20/10/2015	Draft	M. Kabadjov (UESSEX)	Pre-final quality check revisions.
1.7	23/10/2015	Draft	M. Kabadjov (UESSEX)	Added acronyms.
1.8	27/10/2015	Draft	G. Riccardi (UNITN)	Final review.
1.9	29/10/2015	Final	M. Kabadjov (UESSEX)	Final revisions.

Executive summary

In this deliverable we present the progress on the discourse analysis methods developed within the project in Period 2. We continued the lines of work pursued during Period 1 of the project on discourse parsing of spoken conversations, on extracting event structure and temporal expressions and on inter- and intra-document coreference in social media, and in addition to these we pursued a new line of work on argumentation structure of conversations planned for Period 2.

The document is organised as follows: in Section §2, progress on discourse parsing for conversations is presented. Next, work on event extraction and temporal structure from conversation is discussed (§3). Then, progress on intra- and inter-document coreference resolution for conversations in social media is described (Sections §4 and §5) followed by a description of the work on argument structure (§6). Finally, conclusions and future plans are drawn.

Table of Contents

List of Acronyms and Abbreviations

Acronym	Meaning
AMT	Amazon Mechanical Turk
HGI	Harvard General Inquirer lexicon
JRC	Joint Research Centre of the European Commission
MLP	Multi-layer Perceptron
MPQA	Multi-Perspective Question Answering corpus/lexicon
NE	Named Entity
NLP	Natural Language Processing
NN	Neural Network
OnForumS	Online Forum Summarisation
PoS/POS	Part-of-Speech
SIGdial	Special Interest Group on Discourse and Dialogue
SVM	Support Vector Machines
UWB	University of West Bohemia
WP	Work Package

1 Introduction

The objective of WP4 is to develop tools supporting automated discourse analysis of conversations both as happening online (e.g., online forums) as well as in spoken dialogue (e.g., customer call centres). In particular, we aim to develop tools for discourse parsing, event/temporal structure, argumentation structure, and intra-/inter-document coreference in the two domains (social media conversations and call centre conversations) and three languages (English, French, and Italian) of the project. A key goal of the research is to investigate the performance of techniques developed for the most extensively studied forms of language use (e.g., news) in these new domains, and develop methods for adapting such techniques.

1.1 Follow-up to Period 1 Activities

On Discourse Parsing of Conversations (Task 4.1), in Period 1 of the SENSEI project the PDTB-style discourse parsing pipeline developed by [48] was tested for cross-domain and genre generalisation. In the pipeline, discourse parsing is broken down into several sub-tasks: discourse relation detection, argument position classification, argument span extraction, and relation sense classification. Each of the subtasks is different with respect to the type of the discourse relation: explicit (signalled by a discourse connective), or non-explicit – implicit, alternatively lexicalized, or entity relation. In Period 1 the pipeline and the analysis focused entirely on explicit relations; whereas in Period 2 the parser pipeline was extended to cover also non-explicit relations. Additionally, the third party tools that were used (e.g., addDiscourse [34] that was used for discourse connective detection), were replaced by in-house trained Conditional Random Fields and AdaBoost models. In Period 2 the scope of discourse parsing was extended to include Dialogue Act and Overlap Classification tasks (see Section §2).

Work on Event Extraction (Task 4.2) in Year 1 of the project was largely theoretical and investigative, as explained in Section 3 of D4.1. In year 2, we implemented a tool for event detection, as explained in Section §3 of this deliverable; this component is integrated with the project's conversational repository. We also carried out further investigation of temporal extraction in order to implement a component in that area in Year 3.

On intra-document coreference (Task 4.3), in Year 1 of the project the Blog subcorpus of the LiveMemories Anaphora corpus (Italian) was used to adapt the latest version of the BART toolkit on social media data and a new data set (English) from The Guardian was prepared for the OnForumS shared task and annotated for coreference. During Year 2 of the project further experiments on intra-document domain adaptation were carried out using the OnForumS corpus and various data sets from the ARRAU corpus [36], full details of this line of work are provided in Section §4.

On inter-document coreference (Task 4.4), in the second half of Year 1 of the project research

on available tools for entity disambiguation was carried out and a suitable candidate was identified in the JRC-Names resource developed at the Joint Research Centre of the European Commission [47]. During Year 2 of the project we got hold of the tool and ran preliminary experiments on the OnForumS corpus. This line of work is described in Section §5.

The work on Argumentation Structure in Conversations (Task 4.5) started in Year 2 of the SENSEI project according to plan and it progressed along two main lines. The first line of work consisted in designing a suitable annotation scheme and annotating argument structure in Italian, an effort pursued by UNITN. The second line of work involved the inclusion of an argumentation dimension alongside sentiment in the shared task on Online Forum Summarisation (OnForumS) as well as designing a crowdsourcing HIT for the human evaluation of system submissions that participated in the OnForumS shared task, an effort pursued by University of Essex (UESSEX) in collaboration with the University of West Bohemia and University of Trento (UNITN). Both lines are fully described in Section 6.

1.2 Follow-up to Recommendations from the First Review

Further to the recommendations received after the first year review of the project, there were three recommendations relevant to Work Package 4 (WP4). Next we cite each followed by a brief explanation of how the recommendation was addressed deferring the reader to the relevant sections in this deliverable.

R-1. Every language processing task, such as semantic role labelling, coreference resolution or summarization, should have a clear and formal definition, with a baseline given by the current state-of-the-art, and an upper bound of performance that can be expected.

In our experiments on coreference resolution we define and report meaningful performance baselines and upper bounds that can be expected in the context of domain adaptation for coreference resolution. For this purpose we use various data sets from several domains reporting performance shifts as training of models crosses domains (see Section §4.2).

R-2. A systematic error analysis, including the coverage analysis of the language processing algorithms, such as semantic role labelling, coreference resolution or summarization, and the categorisation of errors, should be carried out in each task. Based on this analysis, the work should be prioritised.

In our coreference resolution experiments we carry out an analysis of errors in order to identify which types of coreferential expressions are more challenging and how these challenges are to be tackled. We plan to do further analysis to identify common sources of errors per coreferential expression type and across types (see subsection *Error Analysis* in Section §4.2 for full

details).

R-5. WP3 and WP4 should consider designing a joint processing architecture.

In order to address this recommendation, WP3 (University of Aix-Marseille, AMU) and WP4 (UESSEX) joined forces on integrating their processing architectures. The work was carried out in three main stages. Firstly, the ANCOR corpus of customer call centre conversations in French annotated with coreferential expressions was converted from a tab-separated, CoNLL-like format to BART's native format, MMAX. Secondly, a new language plug-in for French was developed in BART which together with the converter of ANCOR to MMAX made it possible for both processing pipelines, AMU's and UESSEX's, to be integrated at the data level. Thirdly, preliminary experiments were carried out with BART for French coreference. The joint work together with experimental results are described in Section §4.1.

2 Task 4.1: Discourse parsing for conversations

The activities of Task 4.1 within Period 2 are best presented given the perspective on a conversation (spoken synchronous dialogues or written asynchronous conversations in social media). The model we adopt for Task 4.1 is the latest instantiation of the 'information state update' model [53] given by the ISO 24617-2 international standard on dialogue act annotation [8] (see Figure 1). According to this dialogue act annotation meta model, a conversation consists of several *functional segments* – minimal spans of behavior (verbal or not) that have a *communicative function*. Communicative functions take place across multiple *semantic dimensions* (segments are dimension specific and can overlap). Thus, in a model, a *dialogue act* consists of a *communicative function* - *semantic dimension* pair, such that some *communicative functions* are dimension specific and others are general. A *dialogue act* has several *participants*: at least one *sender* and one or more *addressees*. A *communicative function* can be described by *function qualifiers* for aspects such as *sentiment*, *certainty*, and *conditionality*. Dialogue acts can be connected to each other by *functional* and *feedback dependency relations* and *rhetorical/discourse relations*. *Discourse relations* additionally connect *semantic content* to other *dialogue acts* or *semantic content* units of a conversation.

In Section 2.1 we presents experiments on automatic dialogue act classification on Italian LUNA corpus [14]. We identify *dimensions* and *communicative functions* of dialogue segments. Practical utility of dialogue act categorization for SENSEI objectives is in spoken conversation summarization. Since synopses – spoken conversation summaries described in the deliverable D5.2 together with the followed summarization approaches – summarize the **semantic content** of a conversation, the only segments relevant for such summaries belong to the Task dimension of a conversation. Thus, restricting a set of sentences with respect to semantic dimensions of dialogue acts potentially will improve the quality of generated summaries. This topic will be addressed in Period 3 of the project.

In Section 2.2 we present experiments on classification of speech overlaps as competitive and non-competitive using acoustic features. Overlaps can be mapped to the communicative functions in the dimensions of feedback and turn management. Their categorization into competitive vs. non-competitive, however, provides behavioral description of a conversation, e.g. a lot of competitive overlaps signal lack of collaboration. Overlap ratios are already used as behavioral descriptors of a conversation. In Period 3 of the project overlaps will be additionally applied as low-level features for behavioral summarization of spoken conversations.

In Section 2.3 we present the discourse parser trained on Penn Discourse Treebank [39] (English). From Period 1, the relation type coverage of the parser was extended and third party tools were replaced by in-house implementations. The parser was submitted for participation in CoNLL 2015 Shared Task on Shallow Discourse Parsing [57] and in the end-to-end evaluation

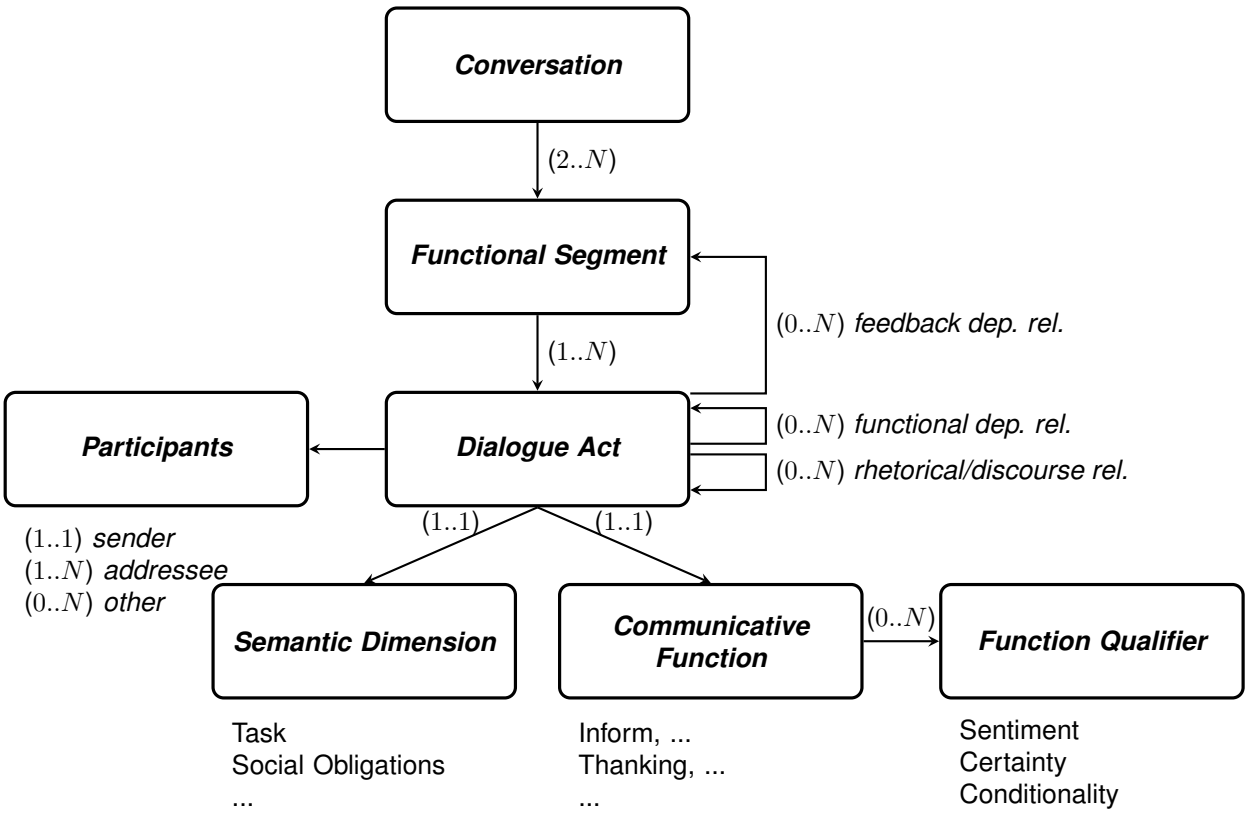

Figure 1: Conversation as a dialogue act annotation model of ISO 24617-2 [8]. A conversation consists of several *functional segments* (marked as $(2..N)$ for number) – minimal spans of behavior (verbal or not) that have a *communicative function* (56) – in multiple *semantic dimensions* (9) (segments are dimension specific and can overlap). Thus, a *dialogue act* consists of a *communicative function* - *semantic dimension* pair and is defined as having *participants* such as *sender* and one or more *addressees*. *Function Qualifiers* are describing how *communicative function* is performed: e.g. with positive sentiment or uncertainty. *Functional* and *feedback dependency relations* connect a *dialogue act* with previously identified conversation units. *Rhetorical/discourse relations* possibly relate *dialogue acts* and *semantic content* to other *dialogue acts* or *semantic content* units of a conversation.

(considering error propagated from all the discourse parsing sub-tasks) ranked the second (out of 16 submissions and 60 initial participant institutions). Thus, most of the components represent the state-of-the-art on PDTB-style discourse parsing. In Period 3 of SENSEI, the parser will be applied to SENSEI data to identify relations between units of a conversation and for generating features for downstream tasks and summarization.

2.1 Dialogue Act Classification

Dialogue Acts (DA) are fundamental for the analysis of conversations: they carry communicative functions such as question, answer, expression of agreement and disagreement, etc.. Consequently, the range of applications of DA analysis is quite wide and includes conversation summarization (both spoken and written), dialogue systems, etc.; and DAs have been extensively studied in both theoretical and computational linguistics. The supervised and unsupervised annotation and classification of DAs (e.g. [21]) and cross-domain and cross-media classification (e.g. forums, email, and spoken conversations [21; 50]) have been shown to yield good results.

A subset of 50 dialogues from Italian LUNA Human-Human corpus [14] was annotated with dialogue acts. The LUNA DA annotation scheme was inspired by DAMSL [13], TRAINS [54], and DIT++ [6]. The most common 15 dialog acts from these taxonomies are grouped into three categories [14]: *Core Dialog Acts* (8) are main actions in the dialog, such as request of information, response, or performing the task; *Conventional/Discourse Management Acts* (4) are utterances such as greetings, apologies, etc. whose function is to maintain general dialog cohesion; *Feedback/Grounding Acts* (3) are utterances whose function is to acknowledge, provide feedback, or just time fillers; and *Others* (1) to capture the rest. The unit of annotation for dialogue acts in LUNA Corpus is an utterance. However, due to the overlapping turns (both speakers speaking), an utterance can span several turns. Thus, the dialogue act annotation was preceded by additional utterance segmentation. In Period 2 of SENSEI, this dialogue act annotation was semi-automatically re-annotated with the recently accepted international ISO standard for DA annotation – Dialogue Act Markup Language (DiAML) [7; 8].

The DiAML annotation scheme [8] is illustrated in Figure 1. In this Section we focus on the DA tag set and dimensions. The DiAML annotation scheme consists of 56 core DA tags[1] (communicative functions), organized into 9 dimensions: 26 general (applicable to any dimension) and 30 dimension specific [16] (see Table 1 for a set of dimensions and communicative functions considered for LUNA Corpus re-annotation). In the following section we present our approach to dialogue act classification and the results obtained on LUNA Corpus.

[1] In the literature the number of dimensions and dimension specific communicative functions varies.

Table 1: Core dimensions and communicative functions from ISO 24617-2 standard considered for LUNA Corpus re-annotation.

Dimension	Comm.Function	Group
General (Task)	*Information Transfer Functions*	
	Question Set Question Choice Question Propositional Question Check Question	Information Seeking
	Inform Answer Confirm Disconfirm Agreement Disagreement Correction	Information Providing
	Action Discussion Functions	
	Offer Promise Address Request Accept Request Decline Request Address Suggest Accept Suggest Decline Suggest	Commissives
	Suggest Request Instruct Address Offer Accept Offer Decline Offer	Directives
Time Management	Stalling, Pausing	
Auto-Feedback	Positive, Negative	
Allo-Feedback	Positive, Negative, Feedback Elicitation	
Social Obligations Management	Initial-Greeting, Return-Greeting Initial-Self-Intro, Return-Self-Intro Apology, Accept-Apology Thanking, Accept-Thanking Initial-Goodbye, Return-Goodbye	

Table 2: Distribution of dialogue acts in LUNA corpus. The counts are given per annotated dimension and in total.

Dimension	**Train** (40)		**Test** (10)		**Total** (50)	
General (Task)	1,456	(74.7%)	494	(25.3%)	1,950	(59.7%)
Social	197	(78.8%)	53	(21.2%)	250	(7.6%)
Auto-Feedback	530	(78.8%)	143	(21.2%)	673	(20.6%)
Allo-Feedback	36	(81.8%)	8	(18.2%)	44	(1.3%)
Time Management	74	(64.9%)	40	(35.1%)	114	(3.5%)
Other	154	(65.0%)	83	(35.0%)	237	(7.3%)
Total	2,447	(74.9%)	821	(25.1%)	3,268	(100.0%)

Table 3: Precision (P), recall (R) and F_1 of dialogue act classification into dimensions.

Dimension	**P**	**R**	**F1**
General (Task)	0.79	0.82	0.81
Social	0.92	0.81	0.86
Time + Feedback	0.69	0.80	0.74
Other	0.29	0.14	0.19
Micro	0.75	0.75	0.75

2.1.1 Classification Methodology

For the dialogue act classification, we used Sequential Minimal Optimisation (SMO), a support vector machine implementation with its linear kernel and default parameters [20]. We perform classification into dimensions and into communicative functions. The distribution of labels in each layer (dimensions and communicative functions) is unbalanced (see Table 2); however, we do not address balancing issues. Since we are mostly interested at detecting the *Task* dimension, we merged *Feedback* and *Time Management* dimensions. The *Social Obligations Management* dimension was kept separate to be compatible with original LUNA dialogue act sets [14]. Performance is evaluated using standard precision, recall and F_1.

2.1.2 Experiments and Results

Table 3 report results on dimension classification. As it can be observed from the table, the model can categorize dimension with 75% accuracy. For the dimension of our interest (i.e. *Task*) the F_1 is satisfactory (0.81).

As a baseline classification into communicative functions we directly classify into 44 communicative functions (*Task*: 26, *Social*: 10, *Auto-Feedback*: 2, *Allo-Feedback*:3, *Time Management*:

2, and *Other*: 1) without considering context. Since some of the communicative functions (e.g. *Confirm*, *Accept Request*, etc.) are hardly distinguishable without considering the partner's dialogue act, we do not expect good performance. The micro-averaged F_1 of such baseline is 0.42.

2.1.3 Conclusions and Period 3 Plans

We described the DiAML [8] annotation scheme that was applied to LUNA corpus. We presented dialogue act classification into dimensions, that have satisfactory levels of performance. Additionally, we have presented the baseline model for classification into communicative functions.

In Period 3 of the project the dialogue act classification models will be improved. The dimension classification will be used for filtering out conversation segments not relevant for the summary generation.

2.2 Overlap Detection and Classification

Overlapping speech is one of the most frequently occurring events in the course of human-human conversations. Understanding the dynamics of overlapping speech is crucial for conversational analysis and for modelling agent-client behaviour. Overlapping speech may signal the speaker's intention to grab the floor with a competitive vs non-competitive act, it also indicate the level of co-operation between the speakers (see Figure 2 for examples of competitive and non-competitive overlaps). In SENSEI, it is used as one of the behavioural descriptors of a spoken conversation. Discourse-wise, overlaps address the *Turn Management* and *Feedback* dimensions of the dialogue act model.

The overlap classification model relies on the identification of the overlapping segments of speech. In case conversation participants are recorded on separate channels, the detection of these segments is trivial. Unfortunately, call center data is usually recorded on a single channel; thus, an overlap detection step from single channel is required. The task is known to be a hard one. For both task – overlap detection and classification – we train model by remixing channels of the annotated data. The data and the process is described in Section 2.2.1. Then, we describe overlap detection and classification experiments in Sections 2.2.2 and 2.2.3, respectively. In Section 2.2.4 we provide conclusions and Period 3 plans.

Non-Competitive
S1: e quando [**cambiamo** ↘] (.)
S2: [**sì sì** ↘ **ho già detto**] di cambiare ↘
S1: and when [***we change*** ↘] (.)
S2: [***yes yes*** ↘ ***I have already told***] *to change* ↘
Competitive
S1: io non lo so [**io devo risparmiare**] ↘(.)
S2: [**ma no la tariffa**] è buona ↗
S1: I do not know [***I had to save***] ↘(.)
S2: [***but no the***] *rate is good* ↗

Figure 2: Competitive and non-competitive overlap examples. Speech overlaps are in **bold** between [and], Hesitations: (.), Rising intonation: ↗, Falling intonation: ↘.

Table 4: Description of the overlap classification data set and the distribution of competitive (C) and non-competitive (N) overlaps in training, development and test sets.

	Dialogues		**Overlaps**		**C**		**N**	
Train	341	(60.35%)	9,537	(2h 55m)	2,379	(24.95%)	7,158	(75.06%)
Dev	109	(19.29%)	3,019	(1h 15m)	724	(23.98%)	2,295	(76.02%)
Test	115	(20.35%)	3,343	(0h 58m)	763	(22.82%)	2,580	(77.18%)
Total	565	(100.0%)	15,899	(5h 08m)	3,866	(24.32%)	12,033	(75.68%)

2.2.1 Training Data and Pre-Processing

The data used for training and testing the overlap detection and classification models is a collection of Italian human-human spoken conversations sampled from a large scale call centre conversations providing customer care support. The conversations are recorded over two separate channels at a sample rate of 8 kHz, 16 bits and have an average duration of 395 seconds. The corpus consists of 565 conversations with approximately 62 hours of data and 5 hours of overlaps. The data split and the distribution of competitive and non-competitive overlaps in the data set is given in Table 4.

Since call centre data is usually recorded on a single channel; to evaluate the performance of the overlap classification on such data, we apply channel remixing on both training and testing data using SoX (Sound eXchange[2]). The whole process is depicted in Figure 3 including training and testing stages, which are described next.

[2] `http://sox.sourceforge.net/`

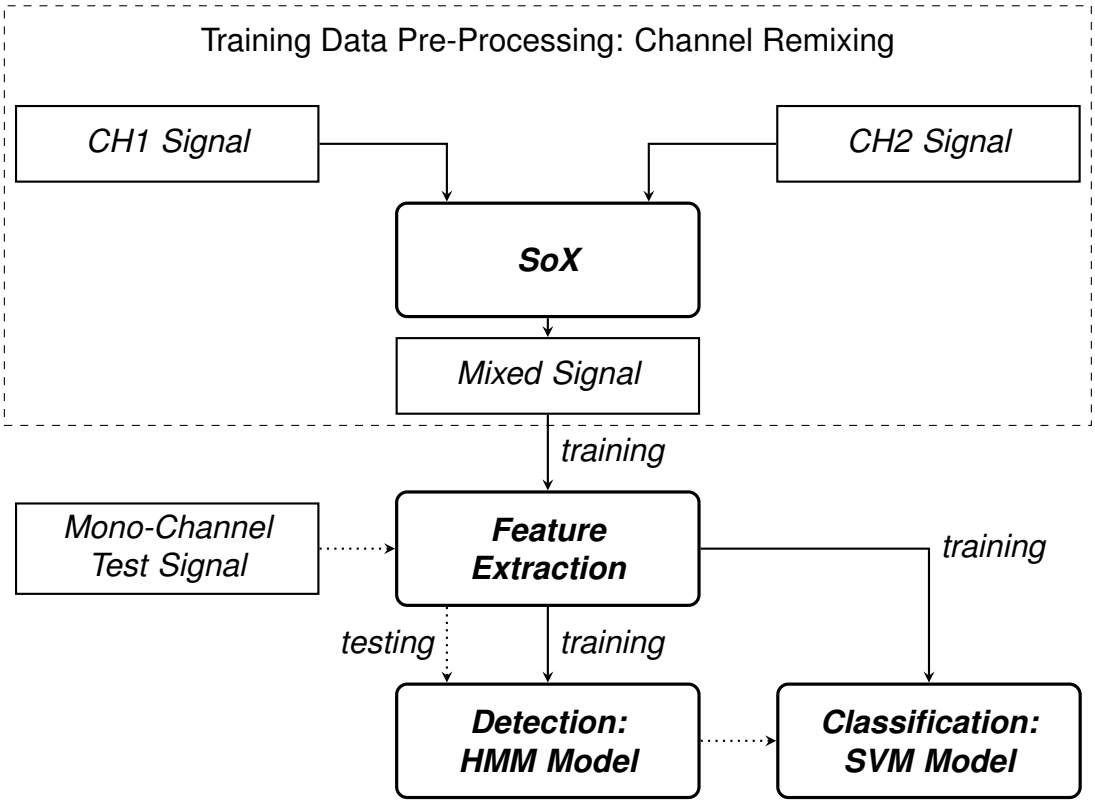

Figure 3: Overlap detection and classification system. Channel remixing (boxed), training (solid arrows) and testing (dotted arrows) pipelines.

Table 5: Mono-channel overlap detection performance as duration weighted (WR) and un-weighted recall (R).

Model	R	WR
HMM	48.05	43.35

2.2.2 Overlap Detection

Overlapping speech is detected using an Hidden Markov Model (HMM)-based overlap segmenter. In HMM, speech or overlap segment is represented with six-states and non-speech with a five-states model. The state emission probabilities are modeled with a multivariate Gaussian Mixture Model (GMM) with 32 components. The segmenter consists of three classes — non-speech, speech, and overlapped speech. Speech, non-speech, and overlap regions are identified in the training data using Automatic Speech Recognition (ASR) forced-alignment segment time, generated from manual transcriptions. The segmentation and labeling of the conversation is performed using a single Viterbi decoding pass on the full audio signal. The non-speech segments (mainly silence) are merged with their surrounding speech/overlap segment. The system is evaluated using NIST speaker diarization evaluation approach [10; 32].

The performance of the system on mono-channel signal is reported in Table 5 as recall and recall weighted by the duration of the overlap segment. The model is able to detect approximately 48% of overlaps. From the results we can observe that it is harder to detect longer overlaps, since duration weighted recall is lower (43.35). Overall, results are promising, and the task will be addressed further in Period 3.

2.2.3 Overlap Classification

The overlap classification model is trained using Sequential Minimal Optimisation (SMO), a support vector machine implementation of weka [20] using linear kernel with default parameter settings. The models is an adaptation of [11; 12] system to mono-channel data.

Models are trained using low-level acoustic features extracted using openSMILE [15] with the FrameSize = 25 ms and FrameStep = 10 ms, which yields approximately 100 frames per second. The groups of these low-level features such as prosodic, energy, etc. with counts are given in Table 6. The extracted low-level features and their derivatives are projected onto statistical functionals such as range, absolute position of max and min, linear and quadratic regression coefficients and their corresponding approximation errors, moments-centroid, variance, standard deviation, skewness, kurtosis, zero crossing rate, peaks, mean peak distance, mean peak, geometric mean of non-zero values and number of non-zeros.

The overlap classification results are given in Table 7. The reported numbers are without error propagation from the overlap detection step. Due to the high ratio of non-competitive overlaps

Table 6: Low-level acoustic features extracted using openSMILE for overlap classification, with the feature counts per channel.

Feature Group	**#**
Prosodic	288
pitch (fundamental frequency F0, F0-envelop) loudness, voice probability	
Voice Quality	288
jitter, shimmer logarithmic harmonics-to-noise ratio (logHNR)	
MFCC	936
Mel-Frequency Cepstral Coefficients (MFCC 0-12)	
Energy	72
Logarithmic signal energy from PCM frames	
Spectral	864
Energy in spectral bands (0-250Hz, 0-650Hz, 250-650Hz, 1-4kHz) roll-off points (25%, 50%, 70%, 90%) centroid, flux, max-position, min-position	
Total	2448

Table 7: Macro- and micro- average F_1 for overlap classification using mono-channel model and a majority baseline.

Model	**Macro-F_1**	**Micro-F_1**
Baseline	43.6	77.2
Dual-Channel	64.4	76.0
Mono-Channel	61.8	76.0

in the test set (77.2%) the micro-averaged F_1 of the majority baseline is high. However, we are interested in both classes; thus, we also report macro-averaged F_1. The described overlap classification system significantly outperforms the baseline considering the macro-averages in both settings – dual channel and single channel.

2.2.4 Conclusions and Period 3 Plans

Classification of overlaps into competitive and non-competitive is used as a behavioural descriptor of conversations already. Currently, the models rely on manual transcriptions or dual-channel data for overlap detection. Additional to the overlap classification system on dual- and mono-channel, we also presented preliminary experiments on overlap detection from mono-channel

data.

In Period 3 of the project we plan to evaluate utility of overlap classification for other SENSEI tasks. Due to the fact that call centre data is usually mono-channel, in Period 3 of the project we also plan to address overlap detection problem.

2.3 PDTB-Style Discourse Parsing

For the identification of discourse relations between dialogue acts and semantic content units of conversations we adopt the Penn Discourse Treebank (PDTB) [39] approach to discourse parsing that can be roughly partitioned into detection of discourse relations, extractions of their argument spans and sense classification. The system described here is the extension of the parser of [48]. The system ranked second in CoNLL 2015 Shared Task on Shallow Discourse Parsing [49; 57] on the end-to-end parsing on a blind test set using strict evaluation that required exact match of all the spans and labels. Thus, most of the systems components represent state-of-the-art performances.

PDTB adopts non-hierarchical binary view on discourse relations: a discourse connective and its two arguments – *Argument 1* and *Argument 2*, which is syntactically attached to the connective. And, a relation is assigned particular sense from the sense hierarchy. In Section 2.3.1 we describe the simplified PDTB discourse relation sense hierarchy that was used in CoNLL 2015 Shared Task on Shallow Discourse Parsing [57]. The parser architecture is described in Section 2.3.2. The features and individual model details are described in Sections 2.3.3 and 2.3.4, respectively. In Section 2.3.5 we provide end-to-end evaluation results and in Section 2.3.6 conclusions and Period 3 plans.

2.3.1 Discourse Relations and Their Senses

In PDTB discourse relations are annotated using 3-level hierarchy of senses. The top level (level 1) senses are the most general: **Expansion**: one clause elaborates on the information given in another (e.g. 'and', 'in addition'); **Comparison**: there is a comparison or contrast between two clauses (e.g. 'but'); **Contingency**: there is a causal relationship between clauses (e.g. 'because'); and **Temporal**: two clauses are connected time-wise (e.g. 'before').

A relation signaled by a discourse connective is an *explicit* discourse relation. *Implicit* discourse relations between text segments (usually sentences), on the other hand, are inferred. The two classes are almost equally represented (53% vs. 47%). While detection of senses of *implicit* discourse relations is a hard problem [25; 57]; presence of a discourse connective in a sentence is sufficient for detection and classification of *explicit* discourse relations.

There are two levels of ambiguity present for a connective [34]: (1) it might be used to con-

Table 8: Simplified PDTB discourse relation sense hierarchy from CoNLL 2015 Shared Task.

Class	Type	Sub-Type
Comparison	*Contrast*	–
	Concession	–
Contingency	*Cause*	Reason Result
	Condition	–
Expansion	*Conjunction*	–
	Instantiation	–
	Restatement	–
	Alternative	– Chosen Alternative
	Exception	–
Temporal	*Synchronous*	–
	Asynchronous	Precedence Succession

nect discourse units, or coordinate smaller constituents (e.g. 'and'); (2) some connectives might have different senses depending on usage (e.g. 'since' might signal causation or temporal relation). AddDiscourse tool was developed by [34] to resolve these ambiguities. While using just connectives the 4-way sense classification accuracy of the tool is 0.9367, incorporating syntactic features raises performance to 0.9415; which is as good as the inter-annotator agreement on the same data (PDTB corpus - 94% [39]). For the CoNLL 2015 Shared Task on Shallow Discourse Parsing some senses were merged, and partial senses were disallowed [57]; as a result, there are only 14 senses listed in Table 8. We classify discourse connectives into this simplified hierarchy of senses.

2.3.2 System Architecture

The overall architecture of the parser is depicted in Figure 4. The approach structures discourse parsing into a pipeline of several subtasks, mimicking the Penn Discourse Treebank (PDTB) [39] annotation procedure as in [26].

The first step is *Discourse Connective Detection* (DCD) that identifies explicit discourse connectives and their spans. Then *Connective Sense Classification* (CSC) is used to classify these connectives into the PDTB hierarchy of senses; and *Argument Position Classification* (APC) to classify the connectives as requiring their *Argument 1* in the previous (PS) or the same sentence as *Argument 2* (i.e. classify relations as inter- and intra-sentential). With respect to the decision of the step an *Argument Span Extraction* (ASE) model is applied to label

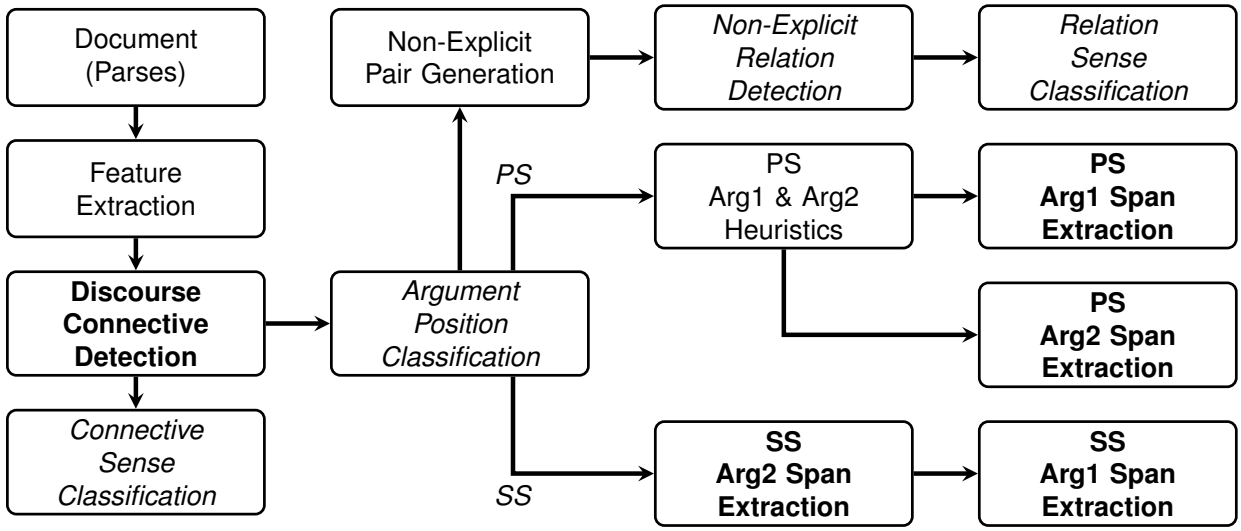

Figure 4: Discourse parser architecture. CRF modules are in **bold**; classification modules are in *italic*.

the spans of both arguments.

Separate *Argument Span Extraction* models are trained for each of the arguments of intra- and inter-sentential explicit discourse relations. Identification of *Argument 2* is much easier, since it is the argument syntactically attached to the discourse connective. Thus, for the intra-sentential (SS) relations, models are applied in a cascade such that the output of *Argument 2* span extraction in the input for *Argument 1* span extraction. For the inter-sentential (PS) relations, a sentence containing the connective is selected as *Argument 2*, and the sentence immediately preceding it as a candidate for *Argument 1*. Even though in 9% of all inter-sentential relations *Argument 1* is located in non-adjacent previous sentence [39], this heuristic is widely used [26; 48], and is known as Previous Sentence Heuristic.

In PDTB, the Non-Explicit discourse relations – Implicit, AltLex, and EntRel – are annotated for pairs of adjacent sentences except the pairs that were already annotated as explicit discourse relations [38]. Thus, in the *Non-Explicit Pair Generation* (NPG) step a list of adjacent sentence pairs is generated omitting the inter-sentential explicit relations identified in the APC step. In the *Non-Explicit Relation Detection* (NRD) step the candidate pairs are classified as holding a relation or not. The pairs identified as a relation are then classified into relation senses in the *Relation Sense Classification* (RSC) step.

Since the goal of *Discourse Connective Detection* and *Argument Span Extraction* tasks is to label the *spans* of a connective and its arguments, they are cast as token-level sequence labelling with CRFs using *CRF++* [24]. The *Non-Explicit Relation Detection* and *Sense* and *Argument Position* classification tasks are cast as supervised classification using AdaBoost algorithm [18] implemented in *icsiboost* [17]. In Section 2.3.3 we describe the features used for token-level sequence labelling and classification tasks; and in Section 2.3.4 models for each of the subtasks in more detail.

Table 9: Token-level features for Discourse Connective Detection (DCD) and Argument Span Extraction (ASE) for intra-sentential (SS) and inter-sentential (PS) explicit discourse relations.

Feature	**DCD**	**ASE: SS**		**ASE: PS**	
		A1	*A2*	*A1*	*A2*
Token	Y	Y	Y	Y	Y
POS-tag	Y		Y	Y	Y
Chunk-tag	Y				
IOB-chain	Y	Y	Y	Y	Y
Dependency chain	Y		Y		
Connective Head	Y				
Connective Label		Y	Y		Y
Argument 2 Label		Y			

2.3.3 Features

Besides tokens as Part-of-Speech tags, the parser relies on features extracted from syntactic constituency and dependency parse trees. These parse trees are used to extract and generate both token-level and argument/relation-level features. Additionally, for argument/relation-level features Brown Clusters [55] are used.

Token-level Features

Discourse Connective Detection and *Argument Span Extraction* tasks of discourse parsing are cast as token-level sequence labelling with CRFs. The list of features used for the models is given in Table 9. Besides tokens and POS-tags, the rest of the features is described below. Features extracted from syntactic constituency parse trees – Chunk-tag and IOB-chain – are illustrated in Figure 5, and features extracted from syntactic dependency parse trees – dependency chain – are illustrated in Figure 6.

Chunk-tag is the syntactic chunk prefixed with the information whether a token is at the beginning (B-), inside (I-) or outside (O) of the constituent (i.e. IOB format) (e.g. 'B-VP' indicates that a token is at the beginning of Verb Phrase chunk). The information is extracted from constituency parse trees using chunklink script [5].

IOB-chain is the path string of the syntactic tree nodes from the root node to the token, similar to Chunk-tag, it is prefixed with the IOB information. For example, the IOB-chain 'I-S/B-VP' indicates that a token is the first word of the verb phrase (B-VP) of the main clause (I-S).The feature is also extracted using the chunklink script [5].

Dependency chain is a feature inspired by *IOB-chain* and is the path string of the functions of the parents of a token, starting from root of a dependency parse. For example, the dependency

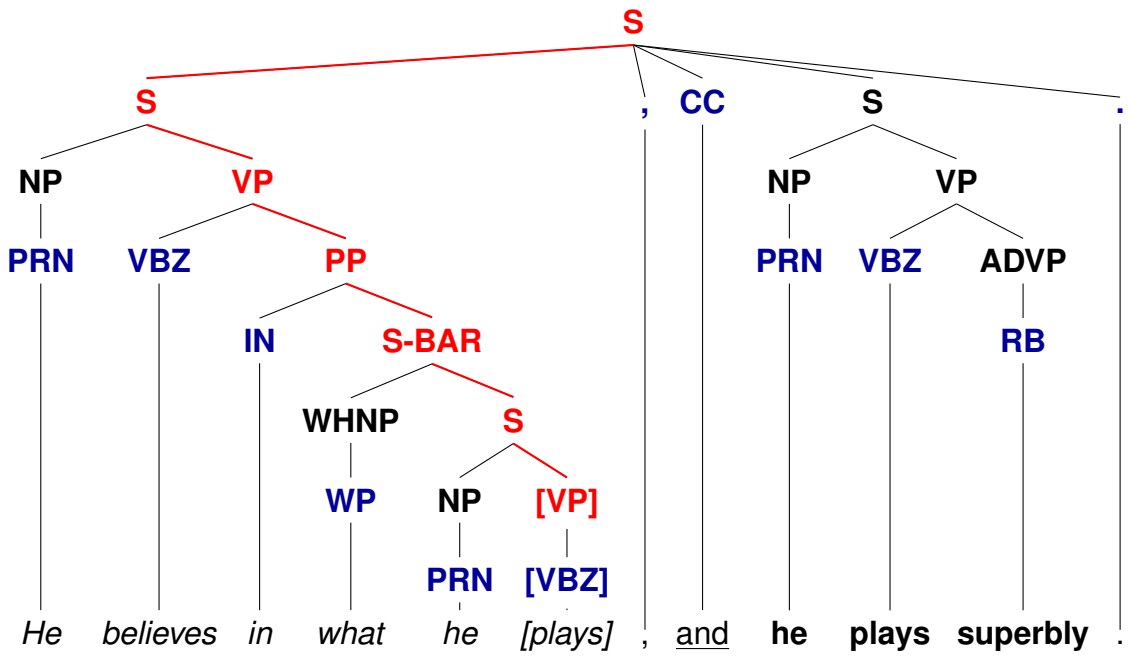

Figure 5: Syntactic constituency tree with the derived features for token *plays* (bracketed). The Part-of-Speech Tag of the token is **VBZ** (bracketed). The token appears at the beginning of the **VP** chunk (bracketed); thus, it has a *Chunk-tag* feature **B-VP**. The IOB-chain (in red) feature for the token is **I-S/I-S/I-VP/I-PP/I-SBAR/I-S/B-VP**.

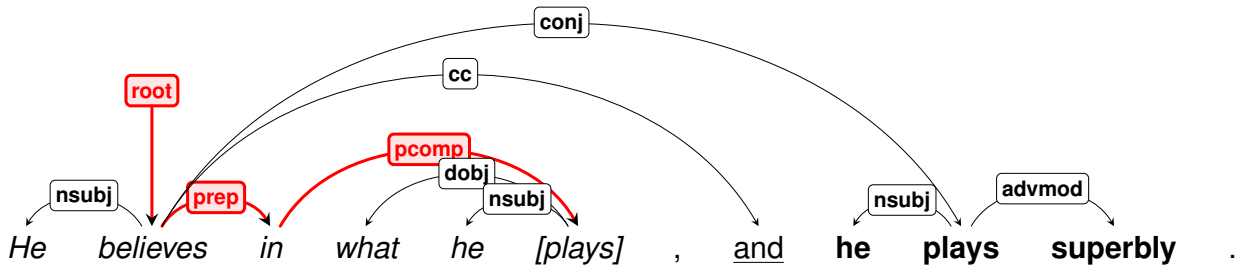

Figure 6: Syntactic dependency tree with the *Dependency-chain* (in red) for token *plays* (bracketed). The feature values is: **root/prep/pcomp**.

chain ‘root/nsubj/det’ indicates that a token is a determiner of the subject of a sentence.

Connective Head is a binary feature that indicates whether a token is in the list of 100 PDTB discourse connectives. For example, all ‘and’ tokens will have this feature value ‘1’.

Connective Label and *Argument 2 Label* are the output labels of the *Discourse Connective Detection* and *Argument 2 Span Extraction* models respectively. The outputs are the IOB-tagged strings ‘CONN’ and ‘ARG2’. Using these labels as features for Argument Span Extraction is useful for constraining the search space, since the *Connective*, *Argument 1* and *Argument 2* spans are not supposed to overlap.

Besides the features mentioned above, we have experimented with other token-level features: (1) morphological: lemma and inflection; (2) dependency: main verb of a sentence (i.e. root of the dependency parse) as a string and binary feature; (3) Connective Head as string. Even though previous work on discourse parsing (e.g. [19; 48] found these features useful in token-level sequence labelling approach to *Argument Span Extraction* using gold parse trees, in greedy hill climbing using features from automatic parse trees their contributions were negative.

Using templates of CRF++ the token-level features are enriched with ngrams (2 & 3-grams) in the window of ± 2 tokens. That is, for each token there are 12 features per feature type: 5 unigrams, 4 bigrams and 3 trigrams. All features are conditioned on the output label independently of each other. Additionally, CRFs consider the previous token’s output label as a feature.

Argument & Relation-level Features

In this section we describe features used for detecting non-explicit discourse relations and their sense classification. Since in these tasks the unit of classification is a relation rather than token, these features are extracted per argument of a relation and a relation as a whole.

Previous work on the topic makes use of wide range of features ranging from first and last tokens of arguments to a Cartesian product of all tokens in both arguments, which leads to a very sparse feature set. To reduce the sparseness in [41] the authors map the tokens to Brown Clusters [55] and improve the classification into top-level senses.

Inspired by the previous research, we have experimented with the following features that are extracted from both arguments:

1. Bag-of-Words;
2. Bag-of-Words prefixed with the argument ID (Arg1 or Arg2);
3. Cartesian product of all the tokens from both arguments;

4. Set of unique pairs from Cartesian product of Brown Clusters of all the tokens from both arguments (inspired by [41]);
5. First, last, and first 3 words of each argument (from [35; 41]);
6. Predicate, subject (both passive and active), direct and indirect objects, extracted from dependency parses (8 features);
7. Ternary features for pairs from 6 to indicate matches (1, 0) or NULL, if one of the arguments misses the feature (extension of 'similar subjects or main predicates' feature of [41]) (16 features);
8. Cartesian product of Brown Clusters of 6 (16 features);

These features are used for *Non-Explicit Discourse Relation Detection* and *Sense Classification* tasks, which are described in the next section.

2.3.4 Discourse Parsing Components

In this section we describe individual discourse parsing subtasks discussing features and models.

Discourse Connective Detection

Discourse Connective Detection is the first step in discourse parsing. The CRF model makes use of all the features in Table 9 (except Connective Label – its own output – and Argument 2 Label – the output of downstream component). Using just cased token features (i.e. 1, 2, 3-grams in the window of $\pm$ 2 tokens already has F-measure above 0.85. Adding other features gradually increases the performance on the PDTB development set to 0.9379. Other than the token itself, the feature that contributes the most to the performance is IOB-chain.

Connective Sense Classification

Connective Sense Classification takes the output of *Discourse Connective Detection* and classifies identified connectives into the hierarchy of PDTB senses. We have experimented with two approaches: (1) flat – directly classifying into full spectrum of senses including class, type and subtype [39]; and (2) hierarchical – first classifying into 4 top level senses (Comparison, Contingency, Expansion and Temporal) and then into the rest of the levels. For the purposes of the Shared Task partial senses (e.g. just class) were disallowed; thus, for the flat classification, instances having partial senses were removed from both training and development sets.

The flat classification into 14 senses using just cased token strings as bag-of-words yields the best performance and has accuracy of 0.8968 on the filtered development set using gold connective spans. The 4-way classification into top-level senses on a full development set using just connective tokens has accuracy of 0.9426. Adding POS-tags increases accuracy to 0.9456. Due to the error propagation, going to the second level of the hierarchy drops the performance slightly below the flat classification. None of the other features listed in Table 9 has a positive effect on classification. Adding argument spans lowered the performance as well.

Argument Position Classification

Argument Position Classification is an easy task, since explicit discourse connectives have a strong preference on the positions of its arguments, depending on whether they appear at the beginning or in the middle of a sentence. In the literature the task was reported as having a very high baseline (e.g. [48], 95% for whole PDTB). The features used for classification are cased connective token string (case here carries the information about connective's position in the sentence), POS-tags and IOB-chains. The accuracy on the PDTB development set given gold connective spans is 0.9868.

Argument Span Extraction

Argument Span Extraction models that make use of the Connective and Argument 2 Labels are trained on reference annotation. Even though, the performance of the upstream models (*Discourse Connective Detection* and *Argument Position Classification*) is relatively high compared to the *Argument Span Extraction* models, there is still error propagation.

For the *Argument Span Extraction* of explicit relations the search space is limited to a single sentence; thus, all multi sentence arguments are missed. This constraint has a little effect on *Argument 2* spans. However, since as a candidate for inter-sentential *Argument 1* we use only immediately preceding sentence, together with this constraint we miss 12% of relations. Thus, detection of *Argument 1* spans of inter-sentential relations is a hard task, additionally due to the fact that there is no other span (connective or Argument 2) to delimit it. For the task we have trained CRF models; however, previous sentence heuristic performs with insignificant difference. The heuristic is additionally augmented with the removal of sentence initial and final punctuation. For *Argument 2* of inter-sentential relations performance of CRF models is acceptably high (≈ 0.80) on the PDTB development set.

The span of *Argument 2* of intra-sentential relations is the easiest to detect, since it is syntactically attached to the connective; and performances are high (≈ 0.89 on the PDTB development set using the features in Table 9). Thus, its output is used as a feature for *Argument 1* ex-

traction. Interesting fact is that POS-tags have a negative effect on the *Argument 1 Span Extraction*.

Non-Explicit Relation Detection

Based on the output of *Argument Position Classification* a set of adjacent sentence pairs is generated as candidates for non-explicit discourse relations: Implicit, AltLex, and EntRel. For training the classification models we have generated No-Relation pairs using reference annotation, excluding all the sentences involved in inter-sentential relations (some relations have multiple sentence arguments). Additionally, since arguments of non-explicit relations are stripped of leading and trailing punctuation, the No-Relation pairs were pre-processed. The task of detecting relations proved to be hard.

Similar to *Connective Sense Classification* we attempted (1) flat classification into all PDTB senses + No-Relation (i.e. merging the task with *Relation Sense Classification* described in Section 2.3.4) and (2) hierarchical – first detect the presence of a relation then classify it into the hierarchy of senses. For the hierarchical detection of Non-Explicit relations we tried (1) Relation vs. No-Relation classification and (2) classification into relation types (Implicit, AltLex, EntRel) + No-Relation. The model that has the highest F-measure for actual relations turned out to be binary Relation vs. No-Relation classification (0.6988). However, since in the end-to-end parsing automatic argument spans are used the performance drops significantly. The most robust feature combination for the task is Cartesian product of Brown Clusters of all the tokens from both arguments and Cartesian product of Brown Clusters of predicate, subject and direct and indirect objects (4 and 8 from Section 2.3.3).

Relation Sense Classification

After a sentence pair is classified as a relation, it is further classified into the hierarchy of senses. The models are trained on all the features from Section 2.3.3, excluding prefixed Bag-of-Words and Cartesian product of all tokens. Relations are classified directly into 14 PDTB senses + EntRel.

The task is extremely hard, the classification accuracy is 0.3899 and the model misses infrequent senses. Table 10 lists the captured senses with their percentages in training data and F_1 on the development set. The distribution of senses has a direct effect on its F_1.

The performances reported so far are on a specific task without error propagation from the upstream tasks. In the next section we report per task and end-to-end results on PDTB development and test sets with error propagation from all the steps.

Table 10: F_1 of non-explicit relation sense classification per sense and as micro-average. Senses are ordered by frequency in the training set.

Sense	%	F_1
Expansion.Conjunction	19.0	0.4247
Expansion.Restatement	14.4	0.3212
Contingency.Cause.Reason	12.2	0.2945
Comparison.Contrast	9.5	0.0980
Contingency.Cause.Result	8.6	0.0563
Expansion.Instantiation	6.5	0.1918
Temporal.Asynchronous.Precedence	2.7	0.1290
Less Frequent and Partial Senses	4.1	0.0000
EntRel	23.1	0.5730
All (micro-average)	–	0.3899

Table 11: Task-level and parser-level F_1 of the parser on PDTB development and test sets for explicit and non-explicit relations individually and jointly. The Sense values are macro-averages.

	Explicit		Non-Explicit		All Relations	
Task	**Dev**	**Test**	**Dev**	**Test**	**Dev**	**Test**
Connective	0.9219	0.9271	–	–	0.9219	0.9271
Arg1	0.5646	0.5008	0.4586	0.4437	0.5225	0.4775
Arg2	0.7748	0.7616	0.4912	0.4744	0.6230	0.6068
Arg1&2	0.5075	0.4460	0.4000	0.3730	0.4499	0.4065
Sense	0.4573	0.3260	0.0601	0.0678	0.3121	0.2526
Parser	0.4760	0.3956	0.1577	0.1330	0.3055	0.2536

2.3.5 End-to-End Parser Evaluation

In the end-to-end evaluation a discourse relation is considered to be predicted correctly if the parser correctly identifies (1) discourse connective span (head), (2) spans and labels of both arguments, and (3) sense of a relation. The predicted connective and arguments spans have to match the reference spans exactly. Consequently, to get a true positive for a relation the parser has to get true positive on all the subtasks. The evaluation is *very* strict. For practical purposes identification of partial spans might be sufficient.

The performance of the parser on each of the sub-tasks on PDTB development and test sets is reported individually and jointly for explicit and non-explicit discourse relations in Table 11. From the results, it is clear that non-explicit Relation Sense Classification is the hardest task. The next hardest task is inter-sentential *Argument 1 Span Extraction*.

2.3.6 Conclusions and Period 3 Plans

We have presented state-of-the-art PDTB-style discourse parser extended for full range of discourse relations within SENSEI project. The end-to-end performance of the parser using strict evaluation is relatively low. The main factor that lowers performance are non-explicit discourse relations. However, as it was observed in [52], in conversations the ratio of explicit relation is higher than in written monologues (65% in LUNA vs 53% in PDTB). Thus, the utility of the discourse parser might be higher.

So far the developed discourse parser has been trained and evaluated on written text only. The objective of Period 3 of the project is to apply discourse parsing to spoken and social conversations. For spoken conversations the parser will utilize dialogue act classification information as well as output of Work Package 3. The parser output will be applied for the generation of summaries as well as low-level features for other SENSEI tasks.

3 Task 4.2: Extracting event and temporal structure from conversations

In this task we develop tools for identifying events and temporal relations in conversations.

3.1 Baseline application

As a baseline implementation for English, we have adapted for SENSEI a combined GATE pipeline originally developed in the ARCOMEM project [27; 28]. This pipeline carries out standard NLP tasks along with NER (named entity recognition), event detection, and sentiment detection.

It consists of the following GATE processing resources.

- Language detection using the TextCat algorithm (if the language detected is not English, the rest of the pipeline is skipped and no output is produced for that document).
- Basic NLP tasks for English (tokenization, sentence-splitting, POS-tagging, lemmatization).
- Named entity recognition for English using ANNIE (gazetteers and rules), and orthographic coreferencing of named entities.
- Noun phrase and verb phrase chunking.
- Date normalization (this will be especially useful for anchoring temporal expressions in the near future).
- Event detection using gazetteers (currently oriented towards financial and major political events and industrial action) and rules.
- Event detection using a large gazetteer of verb nominalizations and rules.
- Sentiment detection using gazetteers and rules.
- Processing the annotations to select the important ones for transfer back to the conversational repository.

3.2 Integration

For SENSEI, we developed a Java wrapper component specifically to interact with the SENSEI document repository developed in WP6. The wrapper polls the repository for batches of documents that have not yet been processed by it, runs the GATE pipeline over them, and adds selected annotation sets and document features back to the same repository documents; it also sets a "flag" feature on them so they do not get processed again by this tool. The wrapper is configurable using an external JSON file which specifies the GATE pipeline to run as well as the annotation sets and document feature to feed back to the repository. The software "wrapper" will therefore be re-usable for other work in SENSEI using GATE applications.

The wrapper and GATE pipeline were successfully used in the "shared task" of linking readers' comments to sentences in newspaper articles, as reported at SIGDIAL [2] and MultiLing [1].

3.3 Conclusion

At this point the achievements in this task consist principally of the integration of GATE applications with the conversational repository and the implementation of a baseline component that includes event detection. The latter has not yet been formally evaluated.

Our further work in this task will include improving this tool and tuning it better for the texts relevant to SENSEI, as well as adding temporal extraction using TimeML or another appropriate representation.

4 Task 4.3: Intra-document coreference for conversations and social media

In this task we tune statistical intra-document coreference algorithms to work with conversational and social media data using the BART platform.

4.1 Coreference in French spoken conversations (UESSEX; AMU)

This section describes our effort towards building coreference resolution system for conversational French. The objective is to apply this coreference resolution system on the Decoda corpus to complement and enrich the existing pronominal anaphora annotations, and eventually enable summarization methods developed in WP5, and semantic annotation approaches from WP3 to make use of detected coreferences.

The initial point for this work is Ancor-Centre, an annotated corpus of spoken conversations for coreference in French, a set of tools provided by WP3 for automatically extracting linguistic annotations from French conversations, and BART, a robust and mature system for coreference open to extensions to new languages. The former two are provided and developed by the University of Aix Marseille (AMU), whereas the latter by the University of Essex (UESSEX). Next, we give a brief description of the Ancor corpus.

4.1.1 The ANCOR corpus

Coreference resolution training corpora is available in a variety of languages (for instance, the SemEval 2010 data cover Catalan, Dutch, English, German, Italian and Spanish), mainly on the News domain. Coreference resolution has been studied on speech through multimodal cues, but resources are much scarcer except on English (Ontonotes data, for instance). In French the only available annotated data is the recently released Ancor-Centre corpus.

The Ancor-Centre corpus was created for the French regional project Ancor by Laboratoire d'Informatique de l'Université François-Rabelais de Tours and Laboratoire Ligérien de Linguistique, Université d'Orléans et de Tours, two NLP labs in the centre of France. The corpus is made of 488,000 words from speech transcripts (30h of speech) and is made of 4 subcorpora with different origin, as detailed in Table 12.

Even though the sources are different from that of the Decoda corpus, they also contain call-centre recordings. The corpus is described in details in [30; 31; 42].

Table 12: Details of the subcorpora of the Ancor-Centre corpus.

Name	Words	Type
CO2	35,000	sociolinguistic interviews
ESLO	417,000	sociolinguistic interviews
OTG	26,000	tourist information call centre
Acccueil_UBS	10,000	university helpdesk call centre

The corpus is labelled with entity mention boundaries, mention features and coreference links. It contains 105,575 mentions with an average length of 1.64 words. Coreference links are established between the current mention and the first mention of the corereference chain. A total of 45,965 coreference events are annotated, resolving to 4,813 different entities. Coreferences are categorised according to their type: pronoun (he, she, it...), bridging anaphora (part of, etc.), pronominal bridging anaphora (such as metonymy with pronoun), direct reference (same nominal head) and indirect reference (different head, such as synonym or hypernym).

Table 13: Types of corereference in the Ancor-Centre corpus.

Frequency	Type
19,557	Pronoun
18,726	Direct reference
4,072	Bridging anaphora
3,203	Indirect reference
407	Pronominal bridging anaphora

Linguistic features are additionally annotated in the corpus, they indicate the gender (feminine, masculine, unknown), number (plural, singular, unknown), named entity type (amount, event, function, location, organisation, person, product, time, n/a), genericity (specific, generic, n/a), definiteness of nominal group (definite, indefinite, demonstrative, expletive), prepositionalness of the group (yes/no), novelty of the discourse element (yes/no). Detailed statistics are available in the documentation of the corpus. The creators of the corpus also provide inter-annotator estimation. It shows a Kappa of 0.45 for segmentation inter-annotator agreement, 0.91 for segmentation intra-annotator agreement, 0.80 for labelling inter-annotator agreement.

Since the corpus is raw text, we have added annotation layers provided by WP3 in order to be able to train a co-reference resolution system. We have added tokenisation, part-of-speech tagging, lemmatisation, morphology and dependency parsing, based on models trained on the Decoda corpus. In future work, we plan on extending this annotation with the semantic frame layer for which tools developed for Decoda require adaptation.

The coreference annotations are extracted from their native xml format and put together with the linguistic annotation layers in the same tab-separated format as the one used for the Decoda corpus, a format similar to the CoNLL tab-separated format. The fields are as follows:

1. File name
2. Global word number
3. Word number in sentence
4. Word text
5. N/A (only used in Decoda)
6. Part-of-speech tag
7. N/A (only used in Decoda)
8. N/A (only used in Decoda)
9. Dependency label
10. Governor
11. Identifier for mapping annotation back to xml
12. Word lemma
13. Morphology
14. Speaker id
15. N/A (only used in Decoda)
16. N/A (only used in Decoda)
17. N/A (only used in Decoda)
18. N/A (only used in Decoda)
19. Mention span begin-inside-outside (BIO) label
20. Coreference features if inside a mention
21. Coreference link and label if any (global word id of head word of antecedent)

The Ancor-Centre corpus, with these annotations, will be distributed to foster work on coreference resolution in the community. We have already communicated with the team of researchers who created the original corpus and they welcomed the contribution from the SENSEI project.

4.1.2 Extending BART to French

The work involved in extending the coreference system BART to French comprised three main stages:

1. Converting the ANCOR corpus to BART's native format, MMAX
2. Developing a language plug-in for French
3. Training coreference models for French

The first step enables integration at the data level of the French pipeline developed in Work Package 3 of the project and BART developed in Work Package 4 (i.e., AMU and UESSEX systems). The second and third steps enable BART to resolve coreference in French.

The coreference system BART already features several format converters from and to its native format MMAX, hence, developing a converter of the ANCOR corpus from CoNLL-like tab-separated format to MMAX involved adapting a converter used for one of the CoNLL shared tasks on coreference where BART has been successfully employed yielding state-of-the-art results.

Developing a French language plugin for BART involved again building on the system's already existing language plugins. The English plugin was chosen for this purpose, and thus the work consisted of translating closed class words such as pronouns, mapping some key part-of-speech tags and adapting some lower-level heuristics for finding the head noun in noun phrases, gender, person and number identification, as well as reading features already available in the input (i.e., which is the output from AMU's pipeline).

Once the converter to MMAX and the French language plugin were completed, it was possible to train coreference models for French. For this, the ANCOR corpus was randomly split in two sets, one consisting of approximately 90% for training and one of 10% for testing the models. Then we ran two experiments to assess where we stand and to produce two coreference models to work with.

The core set of input features (or extractors in BART's terminology) used in both experiments is listed below:

```
<extractor name="FE_MentionType_Coarse"/>
<extractor name="FE_MentionType_Fine_Ana"/>
<extractor name="FE_Gender"/>
<extractor name="FE_Number"/>
<!--  extractor name="FE_Alias"/  -->
<extractor name="FE_AnimacyAgree"/>
<extractor name="FE_DistDiscrete"/>
```

```
<extractor name="FE_SemClassValue"/>
<!--  extractor name="FE_BetterNames"/  -->
<extractor name="FE_First_Mention"/>
<extractor name="FE_CorefChain"/>
<extractor name="FE_DistanceMarkable"/>
<extractor name="FE_DistanceSentence"/>
<extractor name="FE_FirstSecondPerson"/>
<extractor name="FE_NonPron_StrMatch"/>
<extractor name="FE_PronounWordForm"/>
<extractor name="FE_PrprName_StrMatch"/>
<!--  extractor name="FE_WebPatterns"/  -->
<extractor name="FE_CCommand"/>
<extractor name="FE_SameMaxNP"/>
<extractor name="FE_Copula"/>
<extractor name="FE_DE_ShallowRelationIncompatibility"/>
```

In both experiments the learning scheme used was *weka.classifiers.trees.J48*, which is the implementation of the C4.5 decision tree algorithm [40] in the WEKA toolkit.[3]

The first experiment was run using one of BART's coreference encoders called *Soon*, based on work by Soon et al. [45], but using a different set of features. The second experiment uses the coreference encoder called *Split*, which creates and trains separate classifiers for pronouns and for other types of anaphoric expressions. The results for the first experiment are shown in Table 14.

Table 14: Coreference resolution performance using *Soon* encoder.

	Recall	**Precision**	**F1**
Coreference links	38.35%	1.9%	3.63%
Non-coreference links	96.33%	16.15%	27.67%
BLANC	67.34%	9.03%	15.65%
MUC	69.80%	25.20%	37.00%

The results for the second experiment are shown in Table 15.

In both Tables 14 and 15 the top three rows are produced by the official CoNLL scorer[4], whereas the bottom row, MUC score, is generated by BART. From the results it can be seen that BART is able to achieve reasonable recall scores (e.g., MUC score between $65 - 70\%$), but very low precision. It is worth noting here that the Anchor corpus annotation includes only first-mention coreferences and not the whole coreference chain as conventionally done, which explains in part the substantially higher recall on the expense of precision. In future

[3] http://www.cs.waikato.ac.nz/ml/weka/

[4] http://conll.cemantix.org/2012/software.html

Table 15: Coreference resolution performance using *Split* encoder.

	Recall	Precision	F1
Coreference links	37.73%	6.11%	10.53%
Non-coreference links	98.57%	16.02%	27.56%
BLANC	68.15%	11.07%	15.04%
MUC	65.60%	25.50%	36.70%

experiments we will attempt at computing scores taking this factor into account. Additionally, precision can be increased by improving and extending the set of extractors (i.e., input features) and by using higher-precision machine learning schemes such as Support Vector Machines (SVMs) or Neural Networks (NN), though, decision trees are an excellent tool for feature analysis and the interpretability of the models learnt.

4.2 Adapting intra-document coreference to social media (UESSEX)

Coreference has been an intensively researched problem over the past couple of decades. The main focus of this work is to shed some light on what happens when you cross domains (e.g., news, social media) with the aim of identifying effective strategies for adapting/training models to/for new domains making the most of the training data available.

In the coreference literature the most widely studied domain is the news domain, which is regarded as the standard domain for evaluating coreference resolution systems [37]. The domain of interest to this work is that of social media, in particular, the online forum discussions occurring on news provider websites, such as The Guardian[5].

During the second year of the project we worked mainly on experiments with English. We ran five strands of experiments. Firstly, we needed to set a baseline on which to improve, naturally, this is to be the performance of a baseline coreference system within the target domain (i.e., online forums). For that, we ran a 10 fold cross validation (henceforth, 10XVal for short) using the OnForumS corpus [22].

Secondly, we needed to set a meaningful upper bound, and the most obvious choice was to estimate coreference performance within the standard, news domain in the same conditions as the baseline (i.e., using the same system, model and training parameters). For that, we used the news part of the ARRAU corpus, the RST Discourse Treebank [36], which had already conveniently been split into training and test sets.[6]

[5] `www.theguardian.co.uk`

[6] The RST Discourse Treebank is a subset (about a third) of the Penn Treebank whose discourse structure was annotated according to Rhetorical Structure Theory (RST) by Daniel Marcu and collaborators.

Then, the next two natural choices of experiments were to train on different, but similar domain and test on target domain, and likewise, to train on standard news domain and test on target domain. For the former, we used the dialogues part of the ARRAU corpus, the Trains_93[7] [36], whereas for the latter we used the train set of the RST Discourse Treebank, and in both cases testing of trained models was carried out over the OnForumS corpus.

Finally, we ran three experiments crossing domains, by mixing training data from different domains and testing on target domain. For that, we ran the same 10 fold cross validation as for the baseline, only this time we were adding to the train sets (i.e., the remaining 9 folds) data from different domains – we added first dialogues only (Trains_93), then news only (the train set of the RST Discourse Treebank) and last we added both dialogues and news.

All experiments are summarised in Table 16.

Table 16: Coreference resolution performance across domains (with std. dev. across folds or files within brackets).

	Recall	**Precision**	**F1**
Standard domain (news) – upper bound	$54.5\%(\sigma = 9.6)$	$68.5\%(\sigma = 13.6)$	$60.7\%(\sigma = 9.6)$
In-domain (online forums) 10XVal – baseline	$49.2\%(\sigma = 5.7)$	$56.6\%(\sigma = 4.8)$	$52.5\%(\sigma = 5.0)$
Training on similar domain (dialogues)	$42.6\%(\sigma = 4.9)$	$53.3\%(\sigma = 4.6)$	$47.3\%(\sigma = 4.6)$
Training on standard domain (news)	$51.7\%(\sigma = 6.4)$	$53.3\%(\sigma = 4.8)$	$52.4\%(\sigma = 5.3)$
Crossing domains, 10XVal online forums + :			
+ dialogues	$48.2\%(\sigma = 5.9)$	$55.2\%(\sigma = 5.2)$	$51.4\%(\sigma = 5.3)$
+ news	$49.7\%(\sigma = 5.5)$	$56.2\%(\sigma = 4.3)$	$52.7\%(\sigma = 4.6)$
+ news & dialogues	$49.7\%(\sigma = 5.7)$	$56.2\%(\sigma = 4.5)$	$52.7\%(\sigma = 4.8)$

From Table 16 we can see that training on standard domain (news) and testing on target (online forums), $F1 = 52.4\%(\sigma = 5.3)$, is actually better than training on similar domain (dialogues) and testing on target, $F1 = 47.3\%(\sigma = 4.6)$. There is at least two reasons for this. The main reason perhaps is that our OnForumS corpus is a collection of news articles with the corresponding online forum discussions that these evoked, hence naturally, has a substantial overlap with the news domain. And the second reason is corpus size, Trains_93 is substantially smaller in size than RST Discourse Treebank.

Other factors that might account for the differences between Trains_93 and OnForumS are American vs. British English, respectively, and language evolution which is particularly notable in spontaneous human interaction such as happening in dialogues or online forums – the TRAINS 93 corpus is now 22 years old (potentially including texts dating even further back), whereas the online forums discussions in OnForumS are from nowadays (at the beginning of the 90's the concept of online forums either did not exist yet, or was barely in its initial phases of coming to being).

[7] Texts from the TRAINS corpus collected in 1993 to support the development of a conversational agent as part of the TRAINS project at the University of Rochester.

Based on the above discussion we can see from Table 16 that whenever the Trains 93 corpus (dialogues) is included there is a deterioration in performance.

Combining both OnForumS and news data to train models produces best performance, though, not significantly better than the baseline.

Error Analysis

In order to gain some insight into what type of mistakes the system makes we took a closer look at two of the ten folds, fold 3 and fold 9. We first extracted a breakdown of the coreference performance per type of corefential expression. The breakdown is shown in Table 17.

Table 17: Breakdown of coreference resolution performance by coreferential expression type (two folds).

	Fold 3			Fold 9		
Type	**Recall**	**Precision**	**F1**	**Recall**	**Precision**	**F1**
Pronouns	54.7%	35.5%	43%	63.7%	47.7%	54.5%
- 'it'	50%	21.1%	29.6%	45.5%	19.2%	27%
Appositions	100%	25%	40%	0.0%	0.0%	0.0%
Nominals	61.9%	47%	53.5%	58.2%	42.1%	48.9%
Names	82.2%	78.7%	80.4%	89.7%	83.6%	86.5%

From Table 17 it can be seen that the resolution of names, as expected, is quite good and yields the best performance of all types (F1 80.4% and 86.5%), whereas the lowest performance is seen in the resolution of pronouns, in particular, the pronoun 'it' – a well known case amongst the most difficult and ambiguous anaphors to resolve. More effort will have to be devoted towards improving the resolution of pronouns, experimenting with BART's *Split* encoder (mentioned in the previous section) which trains a separate classifier for each type of coreferential expression and allowing each to have its own feature space.

Next, we plan to have a look at specific instances of mistakes in order to come up with common sources of errors for each type of coreferential expression, so that further development can be tailored towards addressing the key sources of errors.

5 Task 4.4: Inter-document coreference for conversations and social media

5.1 Preliminary experiments on inter-document coreference on the social media domain

For our inter-document coreference experiments we used the JRC-Names resource developed at the Joint Research Centre of the European Commission [47]. It is a highly multilingual named entity resource (persons and organisations), which consists of large lists of names and their multiple (in the order of hundreds) spelling variants and transliterations across scripts (Latin, Greek, Arabic, Cyrillic, Japanese, Chinese, etc.). Plugged into a standard pattern matcher, it can be used for multilingual named entity disambiguation across documents, keeping in mind that it was created by analysing millions of news articles in many languages and over many years[8], hence, it can be expected that its optimal performance would be in the news domain.

We ran JRC-Names over the utf8-encoded text versions of the OnForumS files both English and Italian. The number of different entities found per file for English and Italian is summarised in Table 18.

Table 18: Number of entities found per OnForumS file using the JRC entity disambiguator.

English		**Italian**	
File	Entities	File	Entities
1957284403	12	1141349550	14
1965754064	17	1301428792	6
233465322	21	1573695198	14
283147769	12	418022346	21
362778020	18	697213815	10
37793736	16	825497969	18
389321649	30		
540607195	13		
60134403	14		
887344770	25		

Examples of entities identified by JRC-Names can be seen in the following snippet of sample output of processing file 1965754064:

[8] See `http://emm.newsexplorer.eu/`

```
found entity id = 1698324 type p as Edward Snowden (1231)
found entity id = 1698324 type p as Ed Snowden (11958,19023)
found entity id = 988552 type o as Facebook (9945,10536,15759)
found entity id = 17876 type o as Cisco (11545,11638)
found entity id = 17876 type o as CISCO (11664)
found entity id = 5084 type o as Al Quaeda (14050)
```

And likewise for Italian, the following is a snippet of sample output of processing file 1141349550:

```
found entity id = 27275 type p as Oriana Fallaci (17723)
found entity id = 1390261 type p as Jorge Mario (2279)
found entity id = 143519 type o as Lega Nord (16056)
found entity id = 68352 type p as Mark  Twain (5264)
found entity id = 140854 type p as Mario Bergoglio (2285)
found entity id = 140854 type p as Jorge Mario Bergoglio (2279)
found entity id = 963668 type p as Di Maio (6704,6704)
```

In order to have a sense of coverage of the tool we counted the number of coreference chains in the gold standard annotation of the English files[9]. We include these statistics in Table 19. The reason why we show them in a separate table is because they are not directly comparable with the number of entities shown in Table 18, they are only indicative of coverage.[10]

Table 19: Number of coreference chains in the gold standard annotation of the OnForumS corpus.

English	
File	Coreference Chains
1957284403	132
1965754064	127
233465322	177
283147769	150
362778020	122
37793736	119
389321649	147
540607195	137
60134403	155
887344770	113

From Tables 19 and 18 we can see that JRC-Names is able to identify roughly between 8% and 22% of the entities represented by the annotated coreference chains. It is worth noting

[9]The work on annotating the Italian files is still ongoing.

[10]The set of identified entities is not necessarily subsumed by the set of coreference chains, that is, there may be potentially only partial overlap between the two sets.

that many of the coreference chains do not represent persons or organisations which are the scope of the JRC-Names resource.

Next, we plan to integrate the JRC-Names output with that of BART as it provides the bridge of coreference chains across documents, because the JRC-Names disambiguates entities to a unique global id regardless of input.

6 Task 4.5: The argumentation structure of conversations

In this section we describe work during the second year of the project on Task 4.5 of Work Package 4 (WP4). It is divided in two main subsections: one covers work on annotation of argument structure in Italian, the other summarises the effort on the OnForumS shared task and how argument structure was defined and annotated via crowdsourcing for the purposes of the task. We present each in turn below.

6.1 Annotating argument structure in Italian data

In the social media domain, we defined the Argument Structure at two different levels of granularity: coarse grained and fine-grained.

Coarse-grained level. At this level of granularity the Argument Structure is defined as the structure of relations between messages in news blog conversations driven by direct replies, as depicted in Figure 7. This information is available as metadata in all the online newspapers we selected as data sources.

Fine-grained level. At this level the Argument Structure is defined as a set of relations between sentences in news blog conversations constrained by the structure of direct replies and with the same topic, which must be explicit. The fine grained level Argument Structure is depicted in Figure 8.

6.1.1 Annotation Guidelines

We annotated the Argument Structures in the CorEA corpus, a subset of 27 articles and blog conversations we collected from Corriere.it, at both levels of granularity. Following recent literature in the analysis of conversations in social media, we defined the labels for the relations of the Argument Structures as Agreement/Disagreement labels between messages and sentences [3], [56], [58].
Two Italian native speaker annotators labelled the data at both granularity levels, in order to evaluate the annotation with a measure of inter-rater reliability. We designed two different but compatible guidelines, one for each level of granularity.

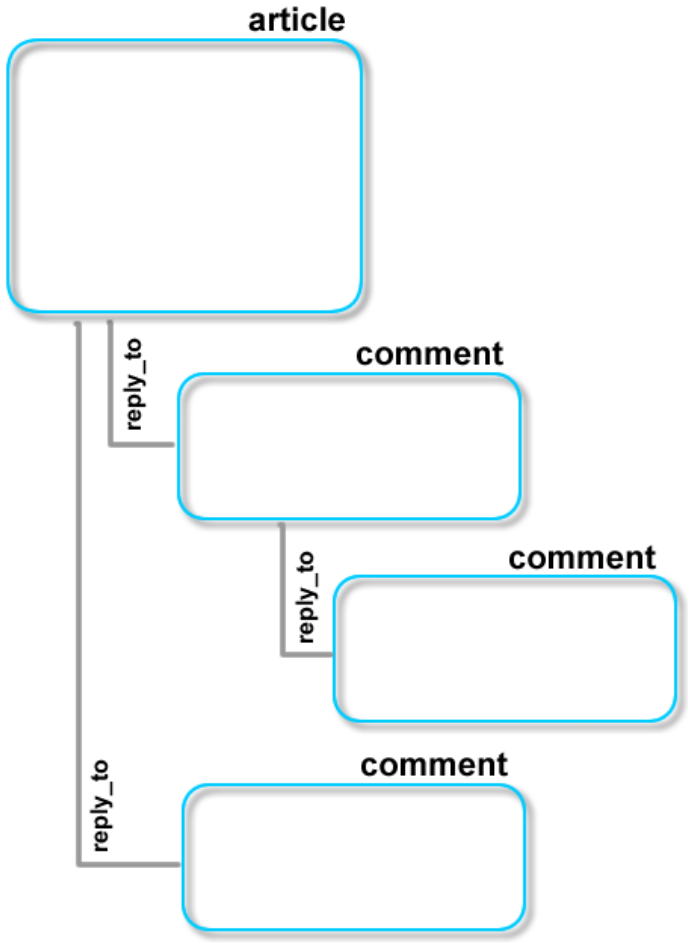

Figure 7: Representation of argument structures at a coarse-grained level

Guidelines for the annotation at Coarse-grained level.

1. Read and understand title and content of the article.
2. Read the comments one by one, sorted by time from the oldest to the newest.
3. For each message pair, check the reply link identifying parent and child messages. (with the term “messages” we refer to article and comments without distinction).
4. Understand the semantics of the relation between the message pair.
5. Annotate with a “NA” label (not applicable) if the relation falls under one or both the following conditions:

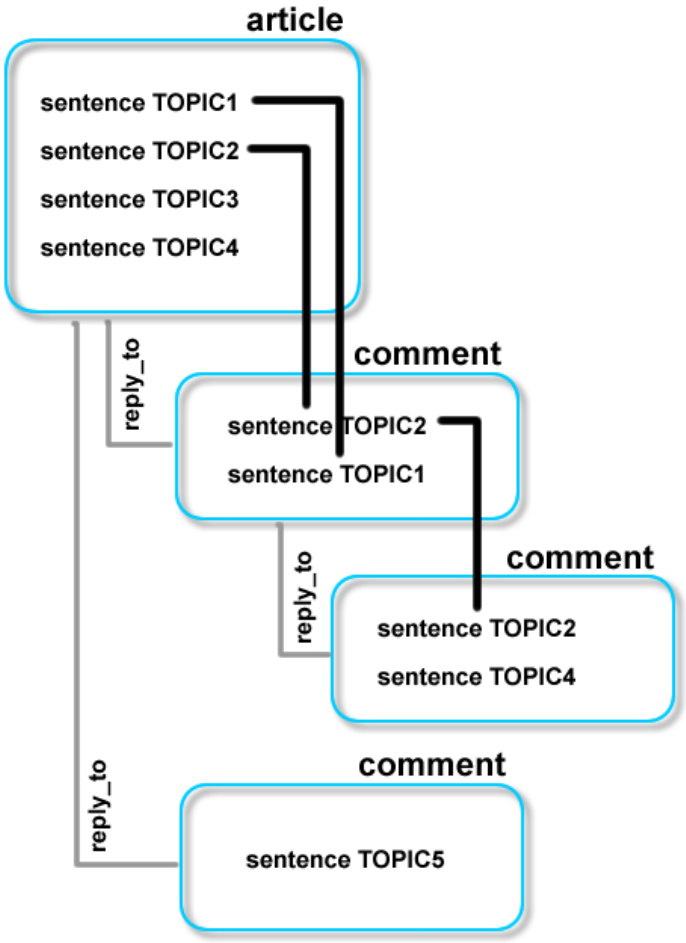

Figure 8: Representation of argument structures at a fine-grained level

(a) **broken reply**: cannot find the parent (e.g., the message is not referred to any other);
(b) **mixed agreement** (e.g., "I partly agree with you but .."");
(c) **unclear utterances**: the content is not related to the conversation: (e.g., links) or the annotator cannot understand the relation between the two messages.

6. Judge the agreement/disagreement expressed in the child message with respect to the parent. Annotate the child pair with the corresponding label: agree (1), disagree (-1) neutral (0).

The annotation at coarse-grain level has been performed manually. An example of annotation follows:

```
1:  5 Stars Movement party returns 2.5 million Euros to Italian citizens.
2:  great!!!.  [agree(2,1)=1]
```

```
3:  http://xyz.com see this :)  ha ha [NA]
4:  what has to do this link with the topic?  [agree(4,3)=-1]
5:  if only every party did it!..  [agree(5,1)=1]
6:  would not change anything.  [agree(6,5)=-1]
7:  what do you mean?  [agree(7,6)=0]
```

Guidelines for the annotation at Fine-grained level. The annotation task at fine-grained level has been performed in three stages: sentence splitting [23], topic extraction and matching [29] [51], and candidate sentence pairs annotation. Following previous literature [3], we used a tool improved the User Interface (UI) of the annotation tool and tailored for our task. A screenshot of the tool is shown in Figure 9. The tool is a web application where the

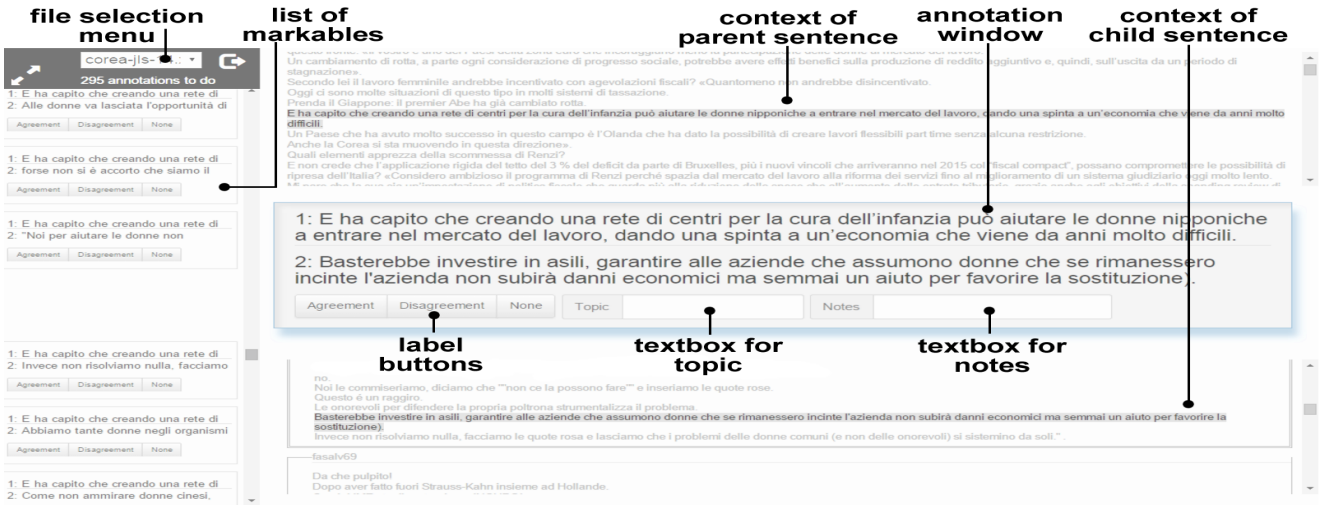

Figure 9: Screenshot of the interface of the tool for the annotation of Agreement/disagreement relations at sentence level. On the top left corner there is the news article selection menu, on the left is the column of the markables to be annotated. The right column is divided into three parts: in the centre (focus) there is the annotation window with label buttons and textboxes, that highlights the pair of sentences to be annotated; on the top and the bottom the contexts of the parent and child sentences respectively.

UI is designed to maximise the annotator attention over the single annotation task. We display the sentence pairs to be annotated at the centre of the annotation space, as well as the intra-message context above and below (see figure 9) the annotation space. The fields to be annotated include labels (agree/disagree/none), topic, free text notes. The topic field displays the keywords in common between the two sentences and it can be edited by annotators.
The guidelines for this task are:
1) Read and understand title and content of the article.
2) Read each automatically extracted sentence pair and its context.

3) Understand the semantics of the relation between the sentences.
4) Annotate with a “NA” label (not applicable), “agree” or “disagree”, according to the following definitions: **Agreement**: sentence B (child) express the same opinion of A (parent) on the same topic or has a positive, supporting tone. E.g. *sentA: I am sure that the boy will make a lot of money from this game! sentB: if he developed the game on his own, for sure he is very smart!*.
Disagreement: sentence B (child) do not express the same opinion on the same topic or has a negative tone towards sentence A (parent). E.g. *sentA: This guy had a great intuition in game design!. sentB: I never said the boy is a genius and I never compared him to Steve Jobs or Bill Gates, this game is bullshit compared to an OS*.
None: there is no relation between sentence A and B. This case happens in the following conditions: **a) not clear** the annotator cannot find or understand the relation between sentences (e.g. *sentA: this boy is smart, I think he should take a degree, it is a pity that he does not want to go to the university. sentB: perhaps the boy is lucky*); **b) mixed agreement** sentence B contains both agreement and disagreement (e.g. *sentA: this game is awesome! sentB: I played the game, it's funny for the first hour, but then is very boring*); **c) wrong topic**: sentences are not about the same topic (e.g. *sentA: The boy wrote his first program when he was 8 years old. sentB: I think he is not so intelligent, if he does not attend any university program*).
5) Report notes about the labelling decision (for example the type of “NA”) and correct the topic, if wrong.

6.1.2 Evaluation of the Annotation

To evaluate the annotation we measured the Inter Annotator Reliability (IAR) and the Intra-annotator Reliability (iAR). Results are reported in table 20.

Table 20: inter-annotator reliability (IAR) and intra-annotator reliability (iAR) scores on the annotation coarse-grain and fine-grained levels.

task	examples	classes	*k*
IAA-msg	100	3	0.57
IAA-msg	50	2	0.85
IAA-sent	93	3	0.66
IAA-sent	51	2	0.88
iAA-msg	100	3	0.87
iAA-msg	100	2	0.91
iAA-sent	166	3	0.64
iAA-sent	53	2	0.80

6.2 Argument structure in the Online Forum Summarisation shared task

Identifying argument structure is currently an active area of research [33; 46]. In the context of the Online Forum Summarisation (OnForumS) shared task, the view of argument structure we adopted was that of articulating a closed set of argument labels for the linking of sentence pairs from reader's comments and news articles. On one hand, linking comment sentences to article sentences is a useful step towards summarising the mass of comments. For instance, comment sentences linked to the same article sentence can be seen as forming a "cluster" of sentences on a specific point or topic. On the other hand, having labels capturing argument structure and sentiment enables computing statistics within such topic clusters on how many readers are in favour or against the point raised by the article sentence and what is the general 'feeling' about it. Consider the following example from our corpus:

(1) S_A**:** In September the environment secretary, Owen Paterson, assured us that climate change "is something we can adapt to over time and we are very good as a race at adapting".
$\hookrightarrow C_1$**:** Human adaptability!!!!!!!!!!!!!! Tell that to ther first dynasty of Egypt (the ones with the pyramids), who died from hunger due to a 30-year drought, the Minoans (volcanic eruption and tsunami), Babylonians (drought), ...
$\rightarrow C_2$**:** Patronising and cynical comment by the Government. I daresay we can 'adapt' to a certain extent but there are limits.

In example 1, the first comment (C_1) links to article sentence S_A through 'human adaptability' and it expresses a view against the quote given in S_A and then the second comment (C_2) seconds the viewpoint of C_1 (it is actually a reply to C_1).

Such clusters of linked sentences are not summaries in themselves, but can be seen as digests of the mass of comments and key points covered in news articles (to an extent resembling the idea of 'capsule overview' put forward in [4]).

The argument labels are: *in_favour*, *against*, *neutral* and *not_applicable*. The choice of modelling argument structure with a closed set of labels is a rather pragmatic choice driven, firstly, by the need to capture both argument structure and sentiment whilst modelling these in an integrated manner[11] and, secondly, by the objective to define a feasible shared task cast as a classification problem that can be tackled with standard machine learning algorithms.

Adopting a more pragmatic view on argument structure also has the advantage that it is suitable for annotation and/or validation of automatic output using crowdsourcing[12], which is a commonly used method for evaluating HLT systems [9; 43]. Thus, the crowdsourcing HIT illustrated in Figure 10 was designed as a validation task (as opposed to annotation), where each system-proposed link and labels are presented to a human contributor for their validation with

[11] The sentiment labels parallel the argument ones and are: *positive*, *negative*, *impartial* and *not_applicable*.
[12] We used CrowdFlower: `http://www.crowdflower.com`

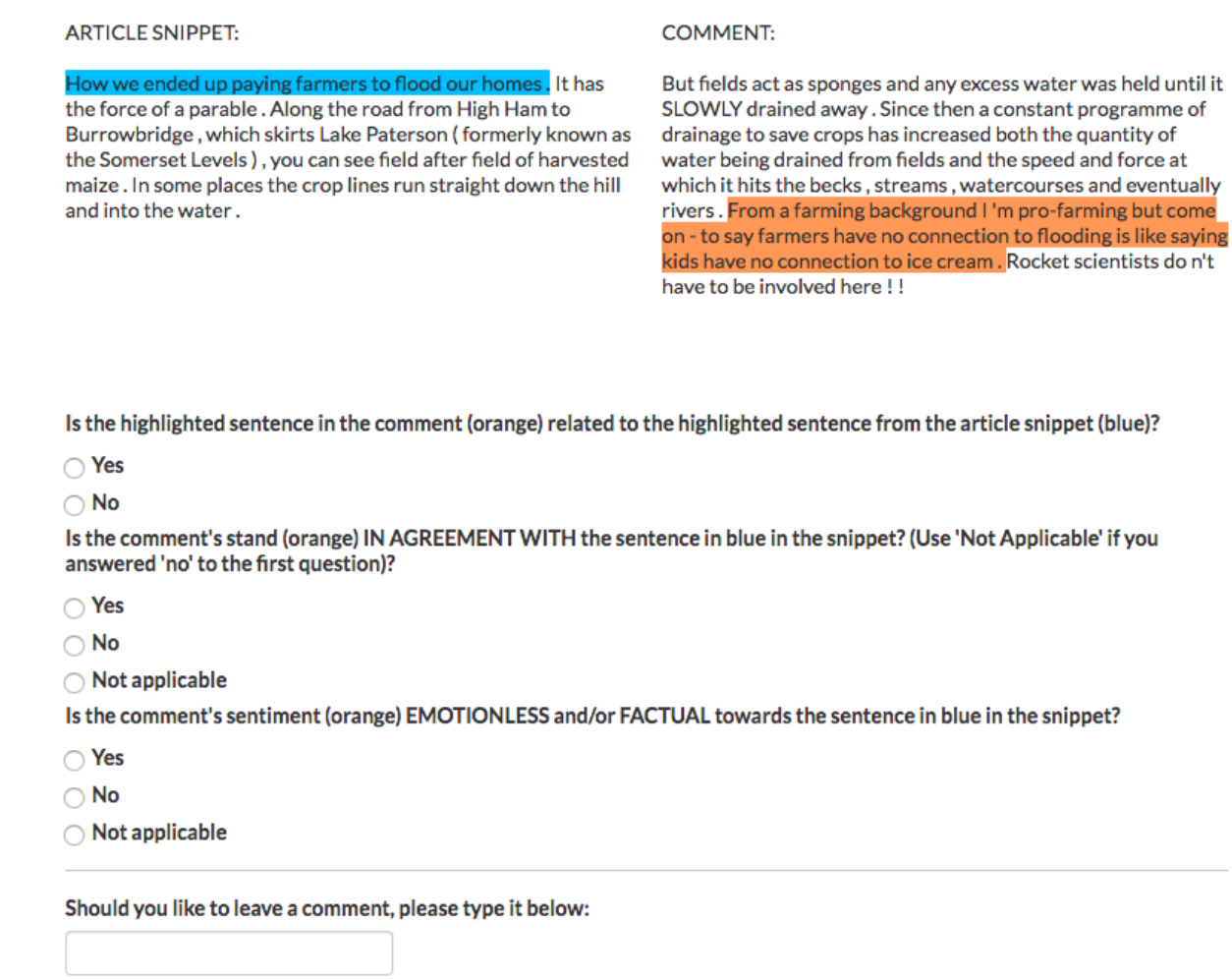

Figure 10: Validation HIT on CrowdFlower.

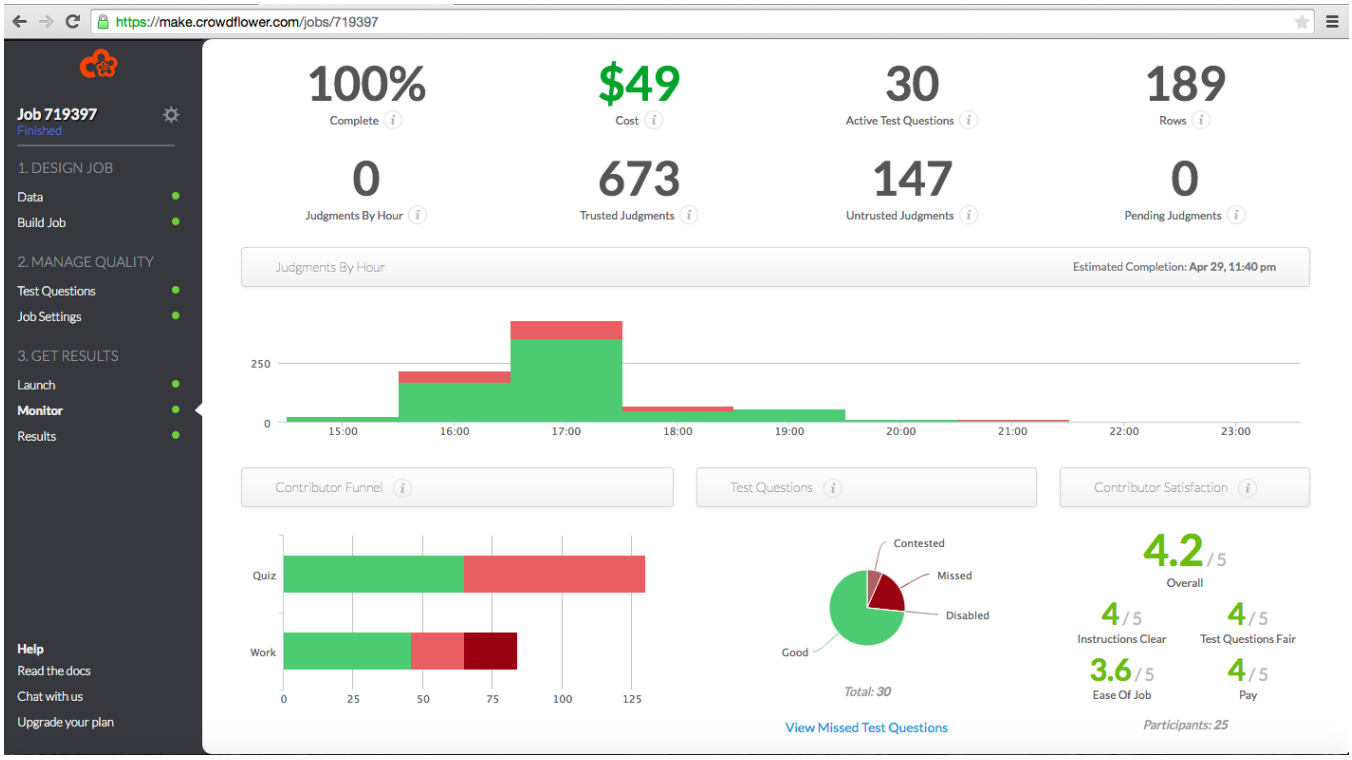

Figure 11: Example of finished CrowdFlower project.

both article sentence and comment sentence placed within context (see Fig. 10).

Both the HIT and the instructions for contributors were translated to English and Italian, thus targeting two distinct groups of native speakers.

Participation in the crowdsourcing HIT varied between $20 - 40$ contributors approximately. A sample snapshot of a finished project from crowdflower can be seen on Figure 11.

Four research groups participated in the OnForumS shared task, each group submitting two runs. In addition, two baseline system runs were included making a total of ten different system runs.

The approach used for the OnForumS evaluation is IR-inspired and based on the concept of *pooling* used in TREC [44], where the assumption is that possible links that were not proposed by any system are deemed irrelevant. Then from those links proposed by systems, four categories are formed as follows (see Table 21 for the cumulative distribution of each):

(a) links proposed in 4 or more system runs
(b) links proposed in 3 system runs
(c) links proposed in 2 system runs
(d) links proposed only once

Table 21: OnForumS corpus: link statistics.

	English	Italian
Links validated (via crowdsourcing)	2311	1087
All Links	9635	6193
Unique Links and Labels	6576	4138
Unique Links only	5789	4016
Type d Links	3517	2083
Type c Links	2975	2024
Type b Links	63	20
Type a Links	21	11

Table 22: Results in terms of precision, recall and F1: English (top scores in bold).

	In_Favour			Against			Positive			Negative		
GroupAndRun	**R**	**P**	**F1**	**R**	**P**	**F1**	**R**	**P**	**F1**	**R**	**P**	**F1**
BASE-first	7.48	28.27	11.31	2.46	6.01	3.35	24.43	22.99	21.97	1.40	2.28	1.68
BASE-overlap	2.26	35.02	4.18	1.07	19.26	1.90	8.27	39.22	12.76	0.65	9.50	1.22
CIST-run1	67.86	24.49	34.94	0.18	1.03	0.28	45.14	24.35	28.58	2.01	2.27	1.97
CIST-run2	**70.79**	**25.18**	**35.99**	0.18	1.17	0.32	**45.61**	**24.64**	**28.72**	2.01	2.47	2.00
JRC-run1	6.78	34.60	10.78	1.15	8.89	2.00	10.01	29.14	12.77	1.37	6.81	2.24
JRC-run2	9.91	31.11	14.39	0.89	4.60	1.43	12.34	26.57	15.36	1.09	4.70	1.64
USFD_UNITN-run1	0.52	43.89	3.34	**5.44**	**5.15**	**4.39**	13.24	26.86	18.93	**3.00**	**5.83**	**6.21**
USFD_UNITN-run2	0.12	50.00	1.18	1.92	3.97	2.44	7.46	29.19	14.50	1.41	4.64	5.59
UWB-run1	12.91	39.16	17.70	0.06	16.67	0.42	6.69	37.75	11.25	0.00	0.00	0.00
UWB-run2	13.78	21.00	14.97	0.06	8.33	0.42	7.26	18.60	9.28	0.00	0.00	0.00

Due to the volume of links proposed by systems, a stratified sample was extracted for evaluation based on the following strategy: all of the **a** and **b** links[13], one third of the **c** links selected at random and one third of the **d** links also selected at random (see Table 21 for numbers of links validated via crowdsourcing).

Once the crowdsourcing exercise was completed, correct and incorrect links were counted first for the linking task only based on the aggregated judgements provided by Crowd Flower[14] (i.e., number of 'yes' and 'no' answers from contributors). From those links validated as correct, the correct and incorrect argument and sentiment labels were counted (again, number of 'yes' and 'no' answers). Using these counts precision scores were computed and system runs were then ranked based on these precision scores. For the linking task no system surpassed the baseline algorithm based on overlap followed by USFD_UNITN's runs, and scores were substantially higher for English than for Italian.

[13] The popular links (**a** and **b**) were not that many, hence, we chose to include all.

[14] An aggregated judgement is based on multiple judgements using CrowdFlower's "agg" method which returns a single "top" result – AKA the contributor response with the highest confidence (agreement weighted by contributor trust) for every given data point (for more details see: `https://success.crowdflower.com/hc/en-us/articles/203527635-CML-and-Instructions-CML-Attribute-Aggregation`).

There are two ways to create gold standard links and labels from the validated data. One is direct validation which entails taking all 'yes' validations of links as gold links and then all labels for argument and sentiment with 'yes' validations as the gold labels for those links. And the other way is by exclusion, if all possible labels for a given link except for one have a 'no' validation then this makes the remaining label a gold label (e.g., if it is not "against", nor "impartial", then it is "in_favour"). With these criteria in mind we created a small gold standard set from which precision, recall and F1 can be computed.

From Table 22 we can see that for top systems recall ranged between $45-70\%$ and precision, $24-25\%$, for the labels *In_Favour* and *Positive*, and precision, $3-5\%$ and around 5% for labels *Against* and *Negative*, respectively.

More details on the OnForumS shared task, such as participating groups, context, impact, etc., can be found in deliverable D7.4 on dissemination.

7 Conclusion

During the second year of the project on Discourse Parsing of Conversations (Task 4.1), the parser pipeline developed by [48] was augmented to cover in addition non-explicit relations and the scope of discourse parsing was extended to include Dialogue Act and Overlap Classification tasks. Also, all third party tools used previously were replaced by in-house trained Conditional Random Fields and AdaBoost models.

During the same period a tool was implemented for Event Extraction (Task 4.2) a tool for event detection, a component which is integrated with the project's conversational repository.

On intra-document coreference (Task 4.3), during year two of the project further experiments on intra-document domain adaptation were carried out using the OnForumS corpus and various data sets from the ARRAU corpus. The findings from the experiments were that training on standard domain (news) and testing on target (online forums) is better than training on similar domain (dialogues) and testing on target. The main reason perhaps is that our OnForumS corpus is a collection of news articles with the corresponding online forum discussions that these evoked, hence naturally, has a substantial overlap with the news domain. And the second key reason is corpus size, the corpus of dialogues we used is substantially smaller in size than the news corpus.

Also during the same period, on inter-document coreference (Task 4.4), we got hold of the JRC-Names resource developed at the Joint Research Centre of the European Commission and ran preliminary experiments on the OnForumS corpus. We found that JRC-Names is able to identify roughly between 8% and 22% of the entities represented by the annotated coreference chains, however, we note that many of the coreference chains do not represent persons or organisations and, hence, are beyond the scope of the JRC tool.

During Period 2 of the project on Argumentation Structure in Conversations (Task 4.5) we defined and implemented a shared task on Online Forum Summarisation (OnForumS). The shared task was grounded on three main pillars, one of which was argument structure in online conversations. The key novelty in the evaluation of system submissions was to bring in crowdsourcing to the evaluation of systems for summarisation and argumentation mining. And on another line of work, we also designed a suitable annotation scheme and carried out annotation of argument structure in Italian.

In the third year of the project we plan to put an emphasis on disseminating the results of the previous two years by publishing in appropriate conferences and journals. In particular, we will target the conference on Language Resources and Evaluation (LREC'16) to publish the work on extending the coreference system BART to French and the work on the data set created for the shared task on Online Forum Summarisation (OnForumS). We will also target core NLP conferences such as ACL to publish the ongoing work on domain adaptation for coreference and an upcoming Special Section of the ACM Transactions on Internet Technology specifically

on Argumentation in Social Media with the aim of publishing the work on argumentation and crowdsourcing.

It is also part of our goals for the third year to follow up on the shared task OnForumS which proved to be a successful pilot track at MultiLing 2015 held jointly with SIGDIAL 2015. We plan to collect more data, refine and extend the definition, methodology and infrastructure of and for the task and hold in the near future a second chapter of the OnForumS campaign jointly with the MultiLing team and tracks. Also, interesting discussions arose at the MultiLing 2015 event with potential for future collaborations, as for instance, to include more languages in future OnForumS campaigns, in particular, Chinese through the CIST group at the University for Posts and Telecommunications of Beijing who were one of the participants of OnForumS'15.

Bibliography

[1] Ahmet Aker, Fabio Celli, Adam Funk, Emina Kurtic, Mark Hepple, and Rob Gaizauskas. Sheffield-Trento System for Sentiment and Argument Structure Enhanced Comment-to-Article Linking in the Online News Domain. `http://multiling.iit.demokritos.gr/file/download/1577`, 2015. [Online; accessed 06-August-2015].

[2] Ahmet Aker, Emina Kurtic, Mark Hepple, Rob Gaizauskas, and Giuseppe Di Fabbrizio. Comment-to-article linking in the online news domain. In *Proceedings of SIGDIAL*, pages 245—-249, Prague, Czech Republic, 2015.

[3] Jacob Andreas, Sara Rosenthal, and Kathleen McKeown. Annotating agreement and disagreement in threaded discussion. In *LREC*, pages 818–822. Citeseer, 2012.

[4] Braninimir Boguraev and Christopher Kennedy. Salience-based content characterisation of text documents. In Inderjeet Mani, editor, *Proceedings of the Workshop on Intelligent and Scalable Text Summarization at the Annual Joint Meeting of the ACL/EACL*, Madrid, 1997.

[5] Sabine Buchholz. chunklink.pl. `http://ilk.uvt.nl/software/`, 2000.

[6] Harry Bunt. A framework for dialogue act specification. In *In Proceedings of SIGSEM WG on Representation of Multimodal Semantic Information*, 2005.

[7] Harry Bunt, Jan Alexandersson, Jean Carletta, Jae-Woong Choe, Alex C. Fang, Koiti Hasida, Kiyong Lee, Volha Petukhova, Andrei Popescu-Belis, Laurent Romary, et al. Towards an ISO standard for dialogue act annotation. In *Seventh conference on International Language Resources and Evaluation (LREC'10)*, 2010.

[8] Harry Bunt, Jan Alexandersson, Jae-Woong Choe, Alex C. Fang, Koiti Hasida, Volha Petukhova, Andrei Popescu-Belis, and David R Traum. ISO 24617-2: A semantically-based standard for dialogue annotation. In *LREC*, pages 430–437, 2012.

[9] C. Callison-Burch. Fast, cheap, and creative: Evaluating translation quality using amazon's mechanical turk. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP'09)*, volume 1, pages 286—-295, 2009.

[10] Diego Castán, Alfonso Ortega, Antonio Miguel, and Eduardo Lleida. Audio segmentation-by-classification approach based on factor analysis in broadcast news domain. *EURASIP Journal on Audio, Speech, and Music Processing*, 2014(1):1–13, 2014.

[11] Shammur A. Chowdhury, Morena Danieli, and Giuseppe Riccardi. The role of speakers and context in classifying competition in overlapping speech. In *Proc. of Interspeech*, 2015.

[12] Shammur Absar Chowdhury, Morena Danieli, and Giuseppe Riccardi. Annotating and categorizing competition in overlap speech. In *ICASSP*. IEEE, 2015.

[13] Mark G. Core and James F. Allen. Coding dialogs with the damsl annotation scheme. In *Proceedings of AAAI Fall Symposium on Communicative Action in Humans and Machines*, 1997.

[14] Marco Dinarelli, Silvia Quarteroni, Sara Tonelli, Alessandro Moschitti, and Giuseppe Riccardi. Annotating spoken dialogs: from speech segments to dialog acts and frame semantics. In *Proceedings of EACL Workshop on the Semantic Representation of Spoken Language*, Athens, Greece, 2009.

[15] Florian Eyben, Felix Weninger, Florian Gross, and Björn Schuller. Recent developments in opensmile, the munich open-source multimedia feature extractor. In *Proc. of the 21st ACM international conference on Multimedia*, pages 835–838. ACM, 2013.

[16] Alex C. Fang, Jing Cao, Harry Bunt, and Xiaoyue Liu. The annotation of the Switchboard Corpus with the new ISO standard for dialogue act analysis. In *Workshop on Interoperable Semantic Annotation*, 2012.

[17] Benoit Favre, Dilek Hakkani-Tür, and Sebastien Cuendet. Icsiboost. `https://github.com/benob/icsiboost/`, 2007.

[18] Yoav Freund and Robert E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1): 119–139, August 1997.

[19] Sucheta Ghosh, Richard Johansson, Giuseppe Riccardi, and Sara Tonelli. Shallow discourse parsing with conditional random fields. In *Proceedings of the 5th International Joint Conference on Natural Language Processing (IJCNLP 2011)*, 2011.

[20] Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H Witten. The weka data mining software: an update. *ACM SIGKDD Explorations Newsletter*, 11(1):10–18, 2009.

[21] Shafiq Joty, Giuseppe Carenini, and Chin-Yew Lin. Unsupervised modeling of dialog acts in asynchronous conversations. In *IJCAI Proceedings-International Joint Conference on Artificial Intelligence*, 2011.

[22] Mijail Kabadjov, Josef Steinberger, Udo Kruschwitz, and Massimo Poesio. OnForumS MultiLing 2015 Task. `http://multiling.iit.demokritos.gr/pages/view/1531/task-onforums-data-and-information`, 2015. [Online; accessed 19-July-2015].

[23] Philipp Koehn. Europarl: A parallel corpus for statistical machine translation. In *MT summit*, volume 5, pages 79–86, 2005.

[24] Taku Kudo. CRF++. http://taku910.github.io/crfpp/, 2013.

[25] Ziheng Lin, Min-Yen Kan, and Hwee Tou Ng. Recognizing implicit discourse relations in the Penn Discourse Treebank. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing (EMNLP 2009)*, 2009.

[26] Ziheng Lin, Hwee Tou Ng, and Min-Yen Kan. A PDTB-styled end-to-end discourse parser. *Natural Language Engineering*, 20:151 – 184, 2014.

[27] Diana Maynard and Adam Funk. Automatic detection of political opinions in tweets. In Raul Garcia-Castro, Fensel Dieter, and Antoniou Grigoris, editors, *The Semantic Web: ESWC 2011 Workshops*, pages 88–99. Springer, 2012.

[28] Diana Maynard, Gerhard Gossen, Adam Funk, and Marco Fisichella. Should I care about your opinion? Detection of opinion interestingness and dynamics in social media. *Future Internet*, 6(3):457–481, 2014.

[29] Andrew K McCallum. Mallet: A machine learning for language toolkit. Technical report, 2002.

[30] Judith Muzerelle, Anaïs Lefeuvre, Jean-Yves Antoine, Emmanuel Schang, Denis Maurel, Jeanne Villaneau, and Iris Eshkol. Ancor, premier corpus de français parlé d'envergure annoté en coréférence et distribué librement. In *TALN'2013, 20e conférence sur le Traitement Automatique des Langues Naturelles*, pages 555–563, 2011.

[31] Judith Muzerelle, Emmanuel Schang, Jean-Yves Antoine, Iris Eshkol, Denis Maurel, Aurore Boyer, and Damien Nouvel. Annotations en chaînes de coréférences et anaphores dans un corpus de discours spontané en français. In *SHS Web of Conferences*, volume 1, pages 2497–2516. EDP Sciences, 2012.

[32] NIST. *The 2009 RT-09 RIch transcription meeting recognition evaluation plan*. NIST, 2009.

[33] Raquel Mochales Palau and Marie-Francine Moens. Argumentation mining. *Artificial Intelligence and Law*, 19(1):1–22, 2011.

[34] Emily Pitler and Ani Nenkova. Using syntax to disambiguate explicit discourse connectives in text. In *Proceedings of the ACL-IJCNLP Conference*, pages 13–16, 2009.

[35] Emily Pitler, Annie Louis, and Ani Nenkova. Automatic sense prediction for implicit discourse relations in text. In *Proceedings of the 47th Annual Meeting of the ACL and the 4th IJCNLP of the AFNLP*, pages 683–691, 2009.

[36] Massimo Poesio and Ron Artstein. Anaphoric annotation in the arrau corpus. In *Proceedings of LREC*, Marrakesh, Morocco, 2008.

[37] Massimo Poesio, Roland Stuckardt, and Yannick Versley, editors. *Anaphora Resolution: Algorithms, Resources and Applications*. Springer–Verlag, 2016. ISBN 978-3-662-47908-7.

[38] Rashmi Prasad, Eleni Miltsakaki, Nikhil Dinesh, Alan Lee, Aravind Joshi, Livio Robaldo, and Bonnie L Webber. *The Penn Discourse Treebank 2.0 Annotation Manual*, 2007.

[39] Rashmi Prasad, Nikhil Dinesh, Alan Lee, Eleni Miltsakaki, Livio Robaldo, Aravind Joshi, and Bonnie Webber. The Penn Discourse Treebank 2.0. In *Proceedings of the 6th International Conference on Language Resources and Evaluation (LREC 2008)*, 2008.

[40] J. Ross Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1993. ISBN 1-55860-238-0.

[41] Attapol T. Rutherford and Nianwen Xue. Discovering implicit discourse relations through Brown Cluster pair representation and coreference patterns. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2014)*, 2014.

[42] Emmanuel Schang, Aurore Boyer-Pelletier, Judith Muzerelle, Jean-Yves Antoine, Iris Eshkol, and Denis Maurel. Coreference and anaphoric annotations for spontaneous speech corpos in french. In *DAARC'2011, 8th Discourse Anaphora and Anaphor Resolu1on Colloquium*, pages 9–pp, 2011.

[43] R. Snow, B. O'Connor, D. Jurafsky, and A. Y. Ng. Cheap and fast – but is it good?: Evaluating nonexpert annotations for natural language tasks. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP'08)*, pages 254—-263, 2008.

[44] Ian Soboroff. Test collection diagnosis and treatment. In *Proceedings of the Third International Workshop on Evaluating Information Access (EVIA)*, pages 34–41, Tokyo, Japan, June 2010.

[45] Hwee Tou Ng Soon, Wee Meng and Daniel Chung Yong Lim. A machine learning approach to coreference resolution of noun phrases. *Computational Linguistics*, 27(4):521–544, 2001.

[46] Christian Stab and Iryna Gurevych. Identifying argumentative discourse structures in persuasive essays. In *Proceedings of EMNLP*, pages 46–56, Doha, Qatar, 2014.

[47] Ralf Steinberger, Bruno Pouliquen, Mijail Kabadjov, Jenya Belyaeva, and Erik van der Goot. JRC-Names: A freely available, highly multilingual named entity resource. In *Proceedings of RANLP*, Hissar, Bulgaria, 2011.

[48] Evgeny A. Stepanov and Giuseppe Riccardi. Comparative evaluation of argument extraction algorithms in discourse relation parsing. In *The 13th International Conference on Parsing Technologies (IWPT 2013)*, pages 36–44, Nara, Japan, November 2013.

[49] Evgeny A. Stepanov, Giuseppe Riccardi, and Ali Orkan Bayer. The UniTN discourse parser in CoNLL 2015 shared task: Token-level sequence labeling with argument-specific models. In *The SIGNLL Conference on Computational Natural Language Learning (CoNLL)- Shared Task*, pages 25–31, Beijing, China, July 2015. ACL.

[50] Maryam Tavafi, Yashar Mehdad, Shafiq Joty, Giuseppe Carenini, and Raymond Ng. Dialogue act recognition in synchronous and asynchronous conversations. In *Proceedings of the SIGDIAL 2013 Conference*, pages 117–121, 2013.

[51] Yee Whye Teh, Michael I Jordan, Matthew J Beal, and David M Blei. Hierarchical dirichlet processes. *Journal of the american statistical association*, 101(476), 2006.

[52] Sara Tonelli, Giuseppe Riccardi, Rashmi Prasad, and Aravind K. Joshi. Annotation of discourse relations for conversational spoken dialogs. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC)*, 2010.

[53] D. Traum and S. Larsson. The information state approach to dialogue act management. In J. van Kuppevelt and R. Smith, editors, *Current and new Directions in Discourse and Dialogue*. Kluwer, Dordrecht, 2003.

[54] David Traum. Conversational agency: The trains-93 dialogue manager. In *Proceedings of Twente Workshop on Language Technology, TWLT-II*, 1996.

[55] Joseph Turian, Lev Ratinov, and Yoshua Bengio. Word representations: A simple and general method for semisupervised learning. In *In ACL*, pages 384–394, 2010.

[56] Lu Wang and Claire Cardie. Improving agreement and disagreement identification in online discussions with a socially-tuned sentiment lexicon. *ACL 2014*, page 97, 2014.

[57] Nianwen Xue, Hwee Tou Ng, Sameer Pradhan, Rashmi Prasad, Christopher Bryant, and Attapol Rutherford. The CoNLL-2015 Shared Task on Shallow Discourse Parsing. In *Proceedings of the Nineteenth Conference on Computational Natural Language Learning: Shared Task*, 2015.

[58] Jie Yin, Paul Thomas, Nalin Narang, and Cecile Paris. Unifying local and global agreement and disagreement classification in online debates. In *Proceedings of the 3rd Workshop in Computational Approaches to Subjectivity and Sentiment Analysis*, pages 61–69. Association for Computational Linguistics, 2012.