EU COMMUNITY

ICT-2013.5.4 ICT for Governance and Policy Modelling

EU COMMUNITY MERGES ICT AND SOCIAL MEDIA NETWORKING WITH ESTABLISHED ONLINE MEDIA AND STAKEHOLDER GROUPS TO CULTIVATE TRANSPARENCY, ENHANCE EFFICIENCY AND STIMULATE FRESH IDEAS FOR EU POLICY-MAKING.

Deliverable D4.4.2

Test Cases, Adaptations and Evaluation Results (Second Version)

Editor(s):	Miltiadis Kokkonidis, Aggeliki Androutsopoulou, Yannis Charalambidis
EU COMMUNITY:	INTRASOFT INTERNATIONAL SA
Status-Version:	Final– v1.0
Date:	30.09.2016
EC Distribution:	R

Project Number:	611964
Project Title:	EU COMMUNITY

Title of Deliverable:	Test Cases, Adaptations and Evaluation Results (First Version)
Date of Delivery to the EC:	30/09/2016

Workpackage responsible for the Deliverable:	WP4 –Policy Modelling and Impact Assessment Component
Editor(s):	Miltiadis Kokkonidis, Aggeliki Androutsopoulou, Yannis Charalambidis
Contributor(s):	INTRA, AEGEAN
Reviewer(s):	FRAUNHOFER IGD, I-EUROPA
Approved by:	All Partners

Abstract:	The document describes the tests, evaluation and adaptations performed on first version of the Policy Component Prototype as well as adaptations to be implemented as part of the development of its second version. It examines separately the Ontology Module and the two subsystems of the Predictions Module, namely the Hybrid Predictions Subsystem and the Simulation Subsystem.
Keyword List:	Test planning, test cases, correctness testing, performance testing, results evaluation, UI, System Dynamics, blended expert-machine approach, ontology, design, simulation

Document Description

Document Revision History

Version	*Date*	*Modifications Introduced*	
		Modification Reason	*Modified by*
V0.1	25/03/2016	Initial version, based on D4.4.1	INTRA, AEGEAN
V0.2	24/05/2016	Split the Hybrid Prediction chapter into two chapters, one about the HPS and one about the Statistical Predictor.	INTRA
V0.3	30/05/2016	Amendments to the Statistical Predictor Chapter	INTRA
V0.4	10/06/2016	Minor edits and comments	AEGEAN
V0.5	17/06/2016	First complete draft of HPS chapter	INTRA
V0.6	20/06/2016	Minor edits and comments	AEGEAN
V0.7	12/09/2016	Draft prepared for review	AEGEAN, INTRA
V0.8	21/09/2016	Revised draft prepared for review	INTRA, AEGEAN
V0.9	30/09/2016	Revisions on the basis of review	INTRA, AEGEAN
V1.0	30/09/2016	Final version – Submitted to Commission	INTRA

Contents

List of Figures

List of Tables

Definitions, Acronyms and Abbreviations

Table 1: Definitions, Acronyms and Abbreviations

Acronym	Title
API	Application Programming Interface
OWL	Web Ontology Language
RDF	Resource Description Framework
SD	System Dynamics
SKOS	Simple Knowledge Organisation System
UI	User Interface
XML	Extensible Markup Language

Executive Summary

Testing and evaluation work in WP4 has been an evolving process, following the evolutionary development process of the Policy Component and the overall project progress and needs. D4.4.2 reports the results of testing and evaluation of the Second Prototype of the Policy Component (described in D4.3.2), and more specifically of its two modules, the Ontology Module and the Predictions Module. The results of this testing and WP-internal evaluation are reported separately for the Ontology Module (Chapter 2), and for the two subsystems of the Predictions Module; Chapter 3 concerns the Simulation Subsystem and Chapters 4 and 5 concern the Hybrid Predictions Subsystem (HPS). Chapter 4 focuses on the Statistical Predictor which is part of the Hybrid Predictions Subsystem, but at the same time it represents an alternative to it, whereas Chapter 5 focuses on the effectiveness of the game-based design of the Hybrid Predictions Subsystem. In each case, different objectives were set and an entirely different approach was taken given the nature of the involved software, its intended users and the availability of data required for its implementation and/or evaluation.

Ontology Module: The Ontology Builder UI was tested extensively. The aim was to investigate if it functioned correctly and performed adequately and whether the claim made during the Year 1 review that the Ontology Builder's design facilitates the input of large segments of ontologies at a fraction of the time this would take using other tools could be substantiated. The results were positive on all fronts.

Statistical Predictor: The Statistical Predictor was tested and evaluated on the basis of the same data it relies on for its predictions, using a cross-validation evaluation methodology. Indeed, it was possible not only to evaluate the original Statistical Predictor originally implemented, but also to provide a significantly improved version on the basis of experimentation and adaptations.

Hybrid Predictions Subsystem: Getting sufficient amounts of human expert opinions to perform credible analysis and evaluation will require a large-scale commercial deployment of PolicyLine or a tool with a comparable focus and the ability to collect such input. The results will depend on the quality of human input. What we aimed to evaluate here was the extent to which the HPS makes the best use of whatever human input it gets. We followed up an internal review of the original design and initial improvements with an evaluation involving the project's External Experts.

Simulation Subsystem: The correctness and performance characteristics of the implementation of the Simulation Subsystem were tested and are deemed satisfactory. As the Simulation Subsystem aims to simulate the behavior of a large on-line community actively engaged with a tool such as PolicyLine, evaluation and adaptation of the Simulation Subsystem will require significant quantities of data obtained in the context of post-pilots operation. A qualitative evaluation on the basis of external experts was performed.

1 Introduction

1.1 Objectives and Purpose

The objective of the current deliverable is to describe the tests, evaluation results and adaptations performed on the second version of the Policy Component Prototype.

D4.4.2 and its predecessor, D4.4.1, aim to ensure WP4 delivers quality through transparency. They describe the WP4-internal testing and evaluation of the Policy Component (see Section 1.2 for a clarification of the aims of different relevant evaluation deliverables belonging to different work packages); in doing so, they reveal a different side to the development of the Policy Component. Whereas D4.3.1 and D4.3.2 present, in the best possible light, the result of the T4.2 design and implementation efforts, namely the Policy Component's first and second prototype, respectively, D4.4.1 and D4.4.2 are responsible for demonstrating the effort that was put into ensuring that the Policy Component meets expectations. This involves discovering and reporting the corresponding version's shortcomings.

It also involves describing adaptations that address, where possible, those shortcomings. Most of these adaptations were performed as part of the work towards the completion of D4.4.1. D4.4.2 additionally covers the effort to improve the design of the Hybrid Prediction Subsystem.

1.2 Relation to other Deliverables

The current Deliverable D4.4.2 is the second of two deliverables produced as outcomes of T4.3 ("Policy Component Testing and Adaptations"). Those two deliverables, D4.4.1 and D4.4.2, follow and complement T4.2 ("Policy Component Design and Development") deliverables D4.3.1 ("Policy Component Prototype (first version)") and D4.3.2 ("Policy Component Prototype (second version)"), respectively. More specifically, they describe testing, evaluation and adaptations performed or planned for the first and second versions, respectively, of the Policy Component.

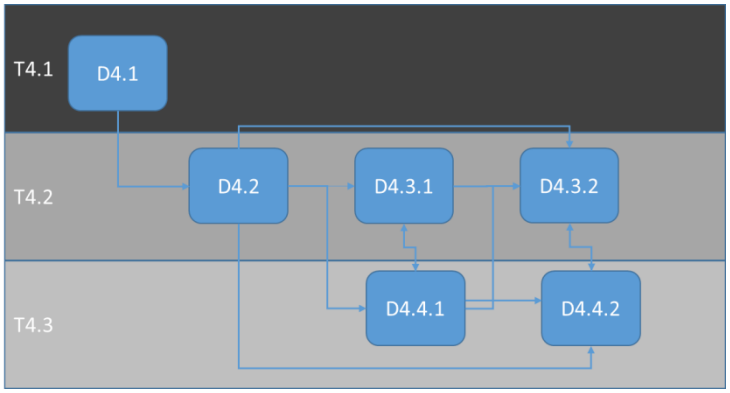

Figure 1: WP4 Tasks and Deliverables

1.3 Structure of the Deliverable

The deliverable is structured as follows:

Chapter 2 presents unscripted UI testing, timed task-based UI testing, browser compatibility, memory usage, UI responsiveness and network testing performed on the Ontology Module and small changes to it that the tests indicated were necessary.

Chapter 3 presents testing, evaluation and adaptations performed on the Statistical Predictor, based on large data quantities crawled from EurLex.

Chapter 4 analyses the game modelling-based design of the Hybrid Predictions Subsystem. This analysis was subjected first to internal evaluations which led to an improved version of the HPS. The improved analysis, presented in this Chapter, was then subjected to external evaluation involving the EU Community External Experts.

Chapter 5 presents the results of technical tests performed on the Simulation Subsystem's implementation and an evaluation based on feedback received from WP4 external experts.

Chapter 6 provides the present report's conclusions and lists possible further steps in relation to exploitation efforts.

Appendix A and **Appendix B** relate to the tests and experimentations that resulted in an improved Statistical Predictor (Chapter 4); they contain the description of different ideas, respectively, for alternative outcome and duration prediction algorithms that aim to improve on the corresponding original outcome and duration prediction algorithms of D4.3.2.

2 Ontology Module

2.1 Introduction

The Ontology Module comprises the Ontology Builder (also known as the Policy Domain Ontology Builder) and the Ontology Server (consisting of the Policy Domain Ontology Server which is responsible for communicating with the Builder and persisting the domain ontologies and the Policy Process Semantic Web Publication Agent).

The Ontology Builder is a web-based user-facing component, albeit one that ordinary users of the EU Community Platform will never use or even be aware of.

The Ontology Builder has functionality for:

1. Maintaining the list of Sections (Policy Domains) and Topics (Topics within Policy Domains) of the EU Community Platform
2. Maintaining an ontology of concepts per each policy domain (Section)
3. Associating concepts with Topics they are relevant to.

The Ontology Server is the server-side of the Ontology Module.

The testing of the Ontology Module focuses on ensuring that the said prototype and especially the Ontology Builder delivers the functionality described in D4.3.1 and D4.3.2 both correctly and with performance and UI effectiveness characteristics compatible with the claims made therein.

2.2 Testing, Evaluation and Adaptations Plan

On the basis of the above aims, a test mix was decided. The text mix included three main categories of tests:

- Unscripted UI Testing
- Timed Task-Based UI Testing and Comparative Evaluation
- Technical Testing
    - Browser Compatibility Tests
    - UI Responsiveness Tests
    - Browser Memory Usage Tests

2.3 Unscripted UI Testing

Unscripted UI Testing involved testing functionality available to each tester without a specific script dictating what action the tester must perform next (and what its output should be). It helped discover bugs which were subsequently corrected.

2.4 Browser Compatibility Testing

Browser compatibility testing was performed to ensure the Ontology Builder could be used on a wide range of browsers. This is indeed the case as summarised in Table 2. Furthermore, it is expected that newer versions of the tested browsers will retain compatibility with the Ontology Builder in the foreseeable future.

Table 2: Browser Compatibility Test Matrix

BROWSERS	Microsoft Internet Explorer& Edge	Mozilla Firefox	Google Chrome	Apple Safari	Opera
VERSIONS	Edge ✓ IE11 ✓ IE10 ✓ IE9 ✓	FF 40✓ FF 38 ✓ FF 36✓ FF9 ✓ FF4✓	GC 30 ✓ GC 42 ✓	AS 6.2✓ AS 5.1✓	O 28 ✓ O12✓

2.5 Browser Memory Usage Testing

The purpose of the Browser Memory Usage test is twofold:

- To ascertain that peak memory demands per action are reasonable
- To ascertain that long-term memory effects per action are reasonable and that there are no memory leaks

The Ontology Builder proved to be particularly economic in its memory use. In order to have a measure for comparison we also tested WebProtégé[1], possibly the most commonly used tool of this kind and found the Ontology Builder to be significantly more memory-efficient.

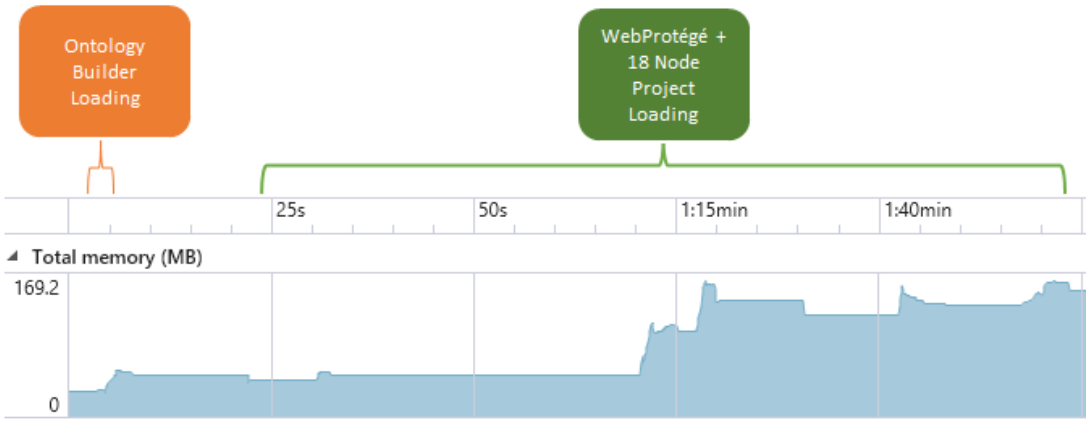

Figure 2: Ontology Builder's vs WebProtégé Loading Time and Memory Requirements

2.6 UI Responsiveness and Network Tests

UI response times always have a component that stems from the UI element rendering required to reflect the changes that must be displayed to the user, but, in the case of client-server applications, are typically determined by perceived server-response times (time for client-sent request to arrive at the server + time to process request and produce reply + time for server-sent response to arrive at the client).The former depends on the user's system capabilities, while the latter depends to a larger extent on the server system's capabilities and the network connecting the client and the server.

Table 3: UI Responsiveness Measurements

Initial Loading	
Loading of Ontology Builder Page	Approx. 1 sec
Activation of Javascript	Approx. 1 sec
Initial Loading of Data (Sections &Topics)	Approx. 1 sec

[1]Protégé (the desktop version) together with its web-based edition known as WebProtégé is among the most widely used ontology building tools; in fact, it probably is *the single* most widely used such tool. It is open-source software (both the desktop and the web edition) and has been developed by the Stanford Center for Biomedical Informatics Research. More information can be found at http://protegewiki.stanford.edu/wiki/Main_Page (Last accessed: 14 June 2015)

Total time before user can interact with Ontology Builder	< 3 sec
Interaction Responsiveness	
Changing a node (section, topic or concept) description (including node removal)	UI Response Time: Instantaneous Server Confirmation Time: < 1 sec
Initiating the addition of a node (section, topic or concept)	UI Response Time: Instantaneous Server Confirmation Time: N/A
Completing the addition of a node (section, topic or concept)	UI Response Time: Instantaneous Server Confirmation Time: < 1 sec
Switching between the three main Sections of the UI (Sections & Topics, Concepts, Topic-Concepts Linking)	UI Response Time : < 1 sec
Switching between the various tabs whilst in the Concepts or Topic-Concepts Linking section of the UI	UI Response Time: < 1 sec (currently; will increase when the ontology gets significantly larger; update to be provided in D4.2.2)

For the purposes of the UI responsiveness and network tests, a VM with 4GB of RAM and two cores on a remote 3GHz Xeon X5472-based system dating back from 2008 served as the server and an i3 (2.2GHz, 2nd Generation) 4GB RAM laptop served as the client. The client was connected to the internet (and therefore to the server) on line with an Ookla-measured 5MBps of download / 0.32 Mbps upload speed. The client setup was purposefully very modest especially in terms of the network connection speed characteristics, in order to best demonstrate that the Ontology Builder UI delivers on its promise of being highly efficient and performant; this would have been much less convincingly conveyed by measurements in a setup where an ultra-powerful server or server-farm was communicating with an ultra-powerful client over an ultra-fast local area network.

As Table 3 above shows the Ontology Builder is very responsive by modern web application standards, partially thanks to the fact it has been designed to be light-weight and partially because it relies on asynchronous communication with the server.

2.7 Timed Task-Based UI Testing

One of the goals of the testing and evaluation performed was to substantiate the claim made during the Year 1 review that the Ontology Builder UI has a very significant advantage over existing ontology editors in terms of how quickly a user can input concept ontologies capturing the kind of information needed in the EU Community Platform.

The initial exhibit in supporting this claim was the timed manual data entry of the energy industry part of EuroVoc 6606 Energy Policy[2] into the EU Community Ontology Builder. This was completed in 8 minutes and 2 seconds using the Ontology Builder in a preliminary test conducted by its principal designer.

Subsequently, a timed task-based test was conducted involving a smaller task (to enter the concepts and relations of Figure 3), the first prototype of the Ontology Builder and the current version of WebProtégé (July 2015), this time involving four testers none of whom had any prior contact with either of the two tools. Prior to the test, the testers were walked through the task and given a chance to start entering data in the tool they were about to test; additionally, they were asked to find ways they thought they could make the task faster and given some suggestions as to how this may be achieved in both tools. The time allocated to the preparation prior to the Ontology Builder test was 2 minutes in the case of the Ontology Builder and 5 minutes in the case of WebProtégé. This difference, as well as the results obtained from the test, are due to the complexity of the interaction patterns necessary in WebProtégé to achieve what can be achieved effortlessly in the Ontology Builder. Two of the testers performed the task first on the Ontology Builder (Tester 1 and Tester 3), whereas the other two performed the task first on WebProtégé. In either case, the testers performed the task with an expert in both tools sitting next to them pinpointing any errors in the data-entry process and advising them whenever needed.

Table 4: Data Entry Test Results

Tester	**EU Community Ontology Builder**	**WebProtégé**
Tester 1	3.2 minutes	10.1 minutes
Tester 2	4.4 minutes	10.6 minutes
Tester 3	3.2 minutes	11.3 minutes
Tester 4	3.1 minutes	10.2 minutes
Average	**3.5 minutes**	**10.6 minutes**

[2] URL:http://eurovoc.europa.eu/drupal/?q=request&mturi=http://eurovoc.europa.eu/100263&language=en&view=mt&ifacelang=en Last Retrieved: 20 July 2015

Here are some observations about the two tools:

- Creating a new concept and entering a label for it is equally simple in both tools.
- Creating alternative labels require additional steps in WebProtégé: the user has to go to the class description for the concept, add a skos:altLabel annotation with the alternative label as its value. This has to be repeated for every additional alternative label. In the Ontology Builder, the user simply writes any alternative labels right after the preferred one separating labels with the '|' character e.g. ("RES | renewable energy sources | renewables").[3]
- The skos:narrowerThan and skos:relatedTo relations between concepts are entered as properties of the left-handside concepts, in WebProtégé, with the right-hand concept as their value. In the Ontology Builder, the moment a new concept is entered, it is declared as being the right-hand side part of the skos:narrowerThan relation, with the left-hand side being its parent concept. If what the user wants to record instead is a skos:relatedTo relationship between the parent and the child concept, this is simply achieved by entering a tilde ('~') character in front of the child concept's preferred label.

By no means is WebProtégé a badly designed or poorly implemented tool; the reason the Ontology Builder is so much more efficient for data entry of EuroVoc data is because it was designed for efficient data entry of specific attributes and relations. We believe that even our preliminary comparative task-based test results show there is room for specialised ontology tools that do not aim to implement generic semantic web standards, but to serve specific purposes, for which those generic semantic web standards may be of use, albeit in the background.

[3]Unfortunately, the test did not include any alternative labels, but it is obvious they would have given a further advantage to the Ontology Builder over WebProtégé.

Figure 3: EuroVoc data used in the comparativetimed data entry tests

3 Statistical Predictor

3.1 Introduction

For any given legislative procedure, the Statistical Predictor answers two questions:

- What will be its outcome with regards to its proposal (Pass with amendments, Pass without amendments, or Fail)
- When will it be completed (i.e. what will its duration be (in months)?)

The aim of the testing and evaluation work on the first prototype of the Statistical Predictor was to:

- Get an estimate of how accurate the Statistical Predictor's predictions can be expected to be.
- Implement or plan accuracy improving adaptations.

The key enabler for the evaluation and adaptation work on the Statistical Predictor presented here was the availability of large quantities of crawled data about past completed legislative procedures.

On the one hand, what was needed was a scientifically valid way of using these data, as they are both what the Statistical Predictor feeds on and what the evaluation of its accuracy is based on.

On the other hand, understanding the characteristics of large quantities of data was necessary in order to explain the behaviour of the Statistical Predictor and decide on improvements; this involved experimenting with data and algorithms.

The present chapter presents our efforts in both providing a suitable evaluation methodology and the results of going through various alternatives to the original algorithms in our effort to increase accuracy.

3.2 Understanding the Statistical Predictor

A **legislative procedure** is a formally defined procedure for the adoption of policy proposals as EU legislation, in accordance with the EU Treaties. It starts with a proposal being adopted by an EU institution, usually the Commission, and then going through a number of steps in a semi-predefined (according to the legislative procedure type) workflow.

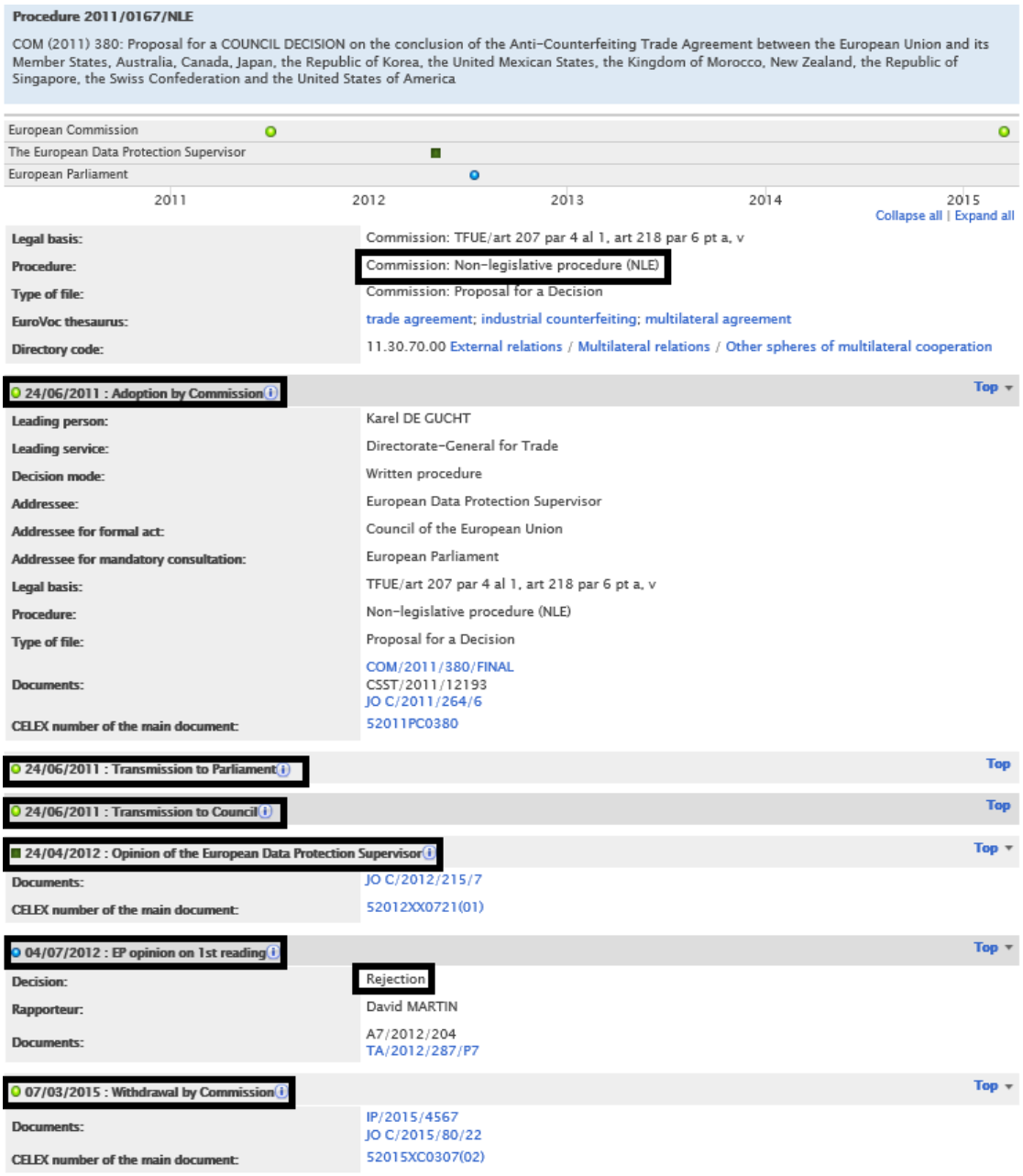

Figure 4: Example of legislative procedure with information used by the Statistical Predictor highlighted

The key idea behind the statistical outcome and duration predictors of D4.3.1 is that if a new legislative process is sufficiently similar to existing legislative processes, its outcome and duration are likely to be the same or similar. What is important to note here is the concept of similarity used.

The Statistical Predictor does not know the content of the proposal nor the content of any other documents related to it. It relies on knowledge of the type of the legislative procedure, as well as of the sequence, types and possible results of any steps in the legislative procedure (see Figure 4). This is a fairly superficial view of legislative procedures but also one that is, as it proves, particularly effective.

Note: Other types of information that could be used include the general policy domain of the policy proposal and possibly the sentiment of various relevant documents. Whether those possibilities will be explored will be decided at a later point. Here we will report on results obtained without the potential benefit of such information.

3.3 Evaluation Methodology

3.3.1 Initial Evaluation Methodology

In order to compute the accuracy of outcome and duration predictions of the statistical predictor, a set of 1,500 completed legislative procedures (for which the outcome and durations are known) was used; all of these were selected to have one of the current EU legislative procedures types.

The available data set was divided into a training and a test set in three ways, with the test set numbering 500 items and the training set 1,000, by breaking down the 1,500 legislative procedures into three 500-item sets and letting the test set be the first, second and third of those sets respectively (Figure 5).

Figure 5: Dividing the crawled legislative procedures into 3 training and test sets

Therefore, three tests were conducted on the basis of the above division of available data into test cases and training data. Because of the way the Statistical Predictor algorithms work, the tests simulated the progress of each test case over time, or to be literally accurate, over the length of its progress history, i.e. the number of steps that are assumed to have been made. As expected, in some cases, the prediction changes once information about an additional step becomes available.

Unless we combine the results of the three tests, what we have is three possible cases of splitting the available data (legislative procedures) into a test and a

training set. One common evaluation methodology, amounts to doing exactly that: separate, say, one third (33.$\bar{3}$%), one fourth (25%), or one fifth (20%) of the available data out in order to use them as test data and use the rest as the training data. We will see that this would not have been a good enough choice of evaluation methodology.

Having three tests instead of just one and combining their results goes some way, as we will see, towards avoiding drawing conclusions about the data, that in fact were caused by the way it happened to be split into test and training data.

With three independent tests, for each step count there are three accuracy metrics (computed as the number of successful predictions divided by the number of test cases for the given step count in that particular test). A single combined accuracy metric for each step count is defined as the sum of the number of successful predictions in the three tests divided by the sum of the number of test cases for the given step count in that particular test.

To get the overall accuracy of the Statistical Predictor when predicting the outcome or duration of a legislative procedure, the number of correct predictions (irrespective of number of steps) is divided by the number of all predictions made in the three tests.

3.3.2 A Closer Look at the Initial Evaluation Method

It is interesting to see how the number of test cases and of relevant training data changes as the number of steps considered increases in the three tests.

All three tests started with 500 test cases for which, we first tested the accuracy of predicting their outcome and duration at a point that not even the first step was assumed to be known, and then we repeated the same tests assuming one more step had been made (a different test case) and so on and so forth.

As the number of steps for which we were conducting the test grew larger and larger, there were fewer test cases for the test; this is because fewer test cases involved at least that many steps. So, for instance, there were, 59, 76 and 51 test cases available for testing with 10 steps, in test 1,2 and 3, respectively, there were 49, 62 and 36, respectively, for testing with 11 steps; this was because 10, 15, and 15 test cases, belonging to the test sets of tests 1, 2, and 3 respectively, were complete after 10 steps and had no 11^{th} step. The longest test case considered, appeared in Test 2 and had 26 steps.

Figure 6 shows the rapid decline of available test cases per step count. The start is at 500, but falls to below the 50 test cases mark after step 12 for Test 2 and after step 10 for Test 1 and Test 3.

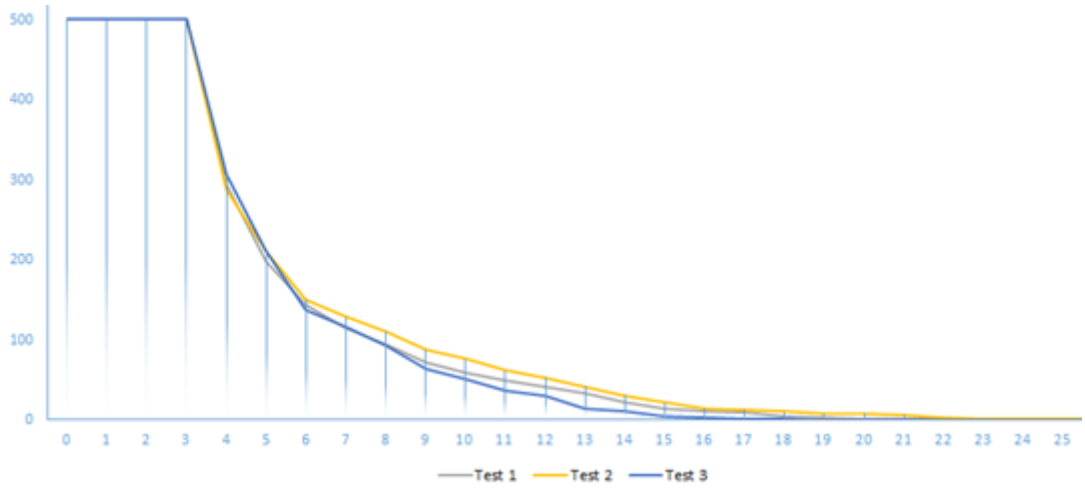

Figure 6: Test Cases Availability per Test and Step Count

Whereas the corresponding 1,000-strong data set made available to the Statistical Predictor in each of the three tests remains available to it in its entirety irrespective of the step count, there is a question of relevance; for making a prediction on a legislative process that currently has X steps, only legislative processes in the training set with X or more steps are relevant. Figure 7 shows that the number of relevant training data also falls rapidly as the number of the steps in the process for which the prediction is to be made increases.

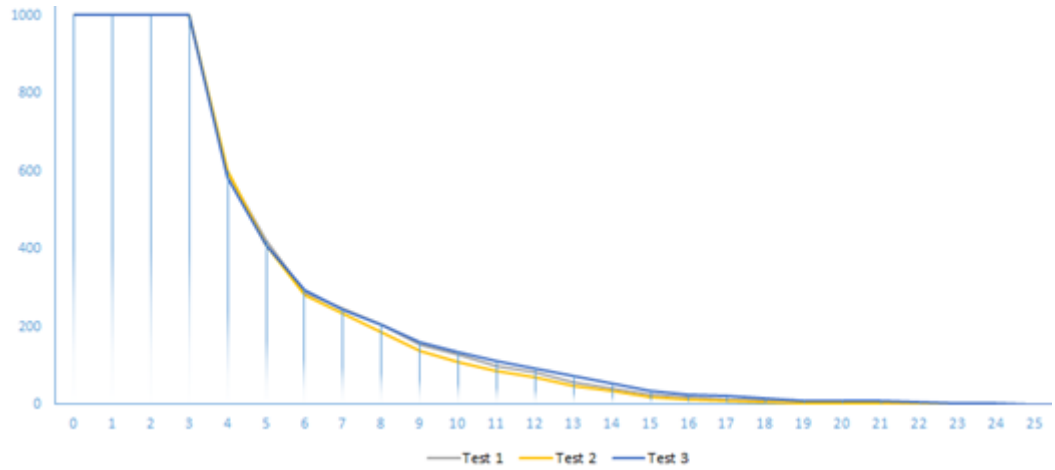

Figure 7: Training Data Relevance per Test and Step Count

Figure 6 and Figure 7 indicate that any results involving steps beyond 10 will not be very reliable, as they will be based on less than 10% of the original test and training sets. Therefore, there are no sufficient grounds to consider as significant either a very good or a very bad result for accuracy of legislative processes of 10 steps or more.

Figure 8 and Figure 9 show the output and duration prediction accuracy, respectively, per number of steps taken in the legislative procedure, for each of the three tests. The differences between the graphs of the three tests demonstrate how problematic it would have been if we had relied on a single test to determine accuracy. The three tests give outcome prediction accuracies of 76.24%, 73.78%, and 81.56% respectively and duration prediction accuracies of 22.5%, 17.4%, and 21.4% respectively. The combined outcome prediction and duration prediction accuracies were 77.08% and 20.4% respectively.

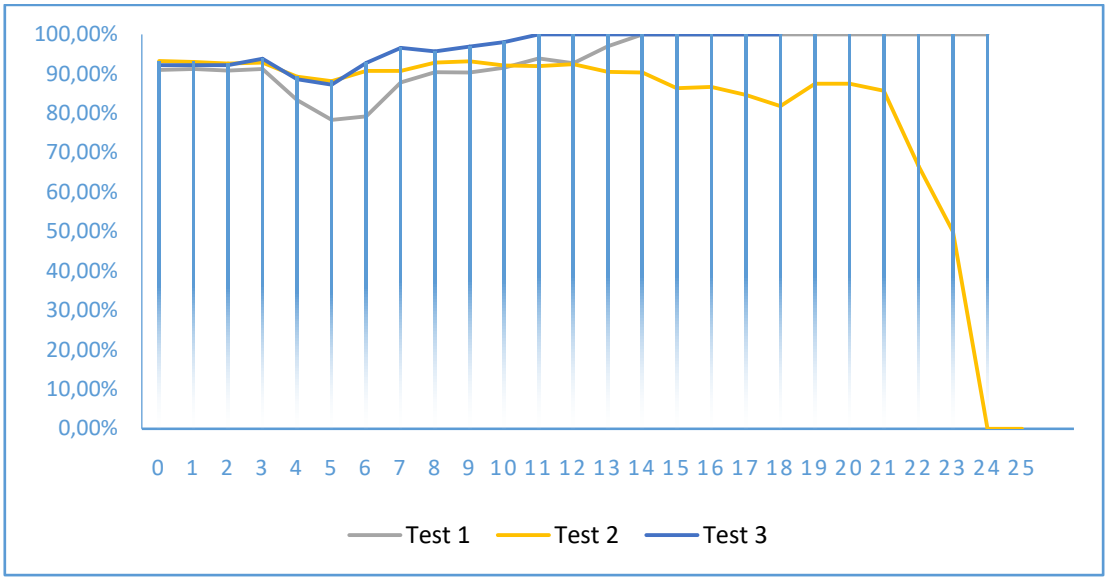

Figure 8: Statistical Predictor: Output Prediction Accuracy for Tests 1, 2 and 3

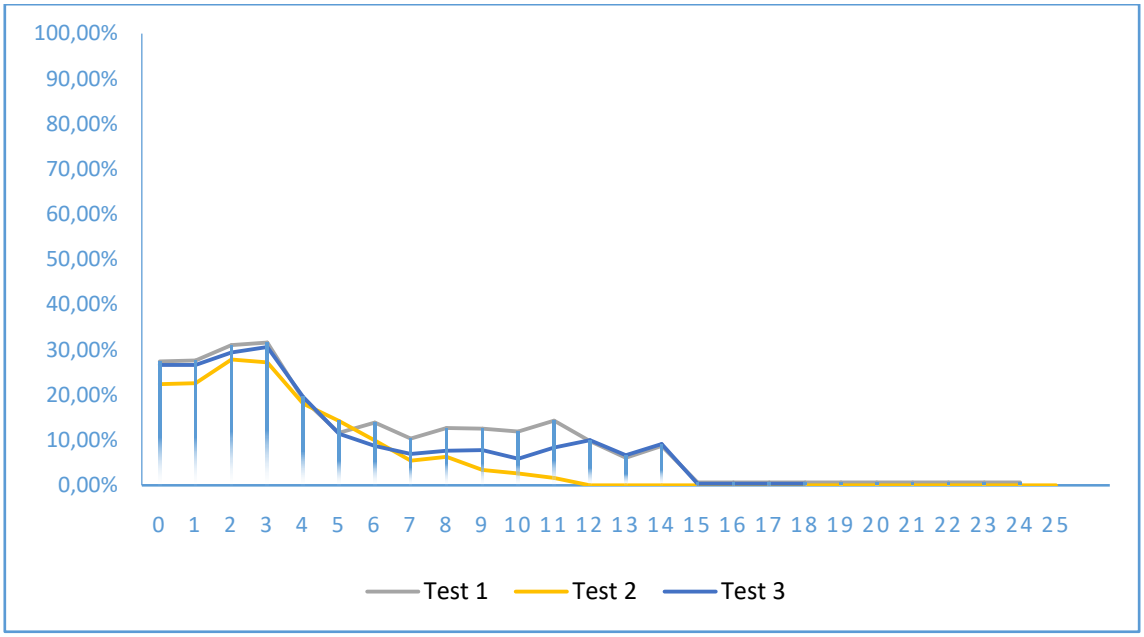

Figure 9: Statistical Predictor: Duration Prediction Accuracy for Tests 1, 2, and 3

3.3.3 Revised Evaluation Methodology

The main problem with the initial methodology is that it sensitive to the ordering of the available data; it matters very much which legislative procedures are in the first, second and third 500 batches, though the order within those batches is immaterial.

For instance, if all legislative procedures of type Ordinary Legislative Procedure (OLP) happen to be in the third batch, then in Test 1, the predictions involving the OLPs will be conducted with no corresponding OLP training data and for Test 2 and Test 3 the available OLP data will be in their training sets but no OLPs will be in the corresponding test sets. A similarly unfortunate situation occurs if legislative procedures are ordered by length.

Standard practice is to ensure the list of data items is shuffled randomly before they are separated into the three batches of Figure 5 for the purposes of conducting the three tests required by the initial evaluation methodology. However, this does not address the problem at its core: after all, the result of randomly shuffling an entire list that may not exhibit one of the above mentioned problematic distributions, may result in one that does.

The primary reason we wanted to have a reliable evaluation methodology is obvious: we wanted to have an evaluation methodology the results of which we could quote with confidence. But for our work, the evaluation methodology was to also serve as a tool that would allow us to explore properties of the available legislative procedure data and the behaviour of the Statistical Predictor on that data. This would be very difficult if we always had to second-guess the results the methodology we were using produced. On more than one occasions we had to wonder if results we had obtained using the initial methodology were the result of an unfortunate segmentation of legislative procedures or a genuine feature of the available data which we would need to understand, explain and/or try to take advantage of.

One way to be reasonably sure that any trends exhibited are genuine and not a by-product of the methodology would be to repeat the three tests a number of times after randomly reshuffling the legislative procedures. This is time consuming and would create a multitude of results and graphs to be examined. Moreover, there is a question of how many times this process would need to be repeated.

The methodology we planned to use did not suffer from those drawbacks, but needed additional implementation effort so the methodology we described earlier was used in order to obtain some first results and conduct a fist set of experiments. When the time came to move past initial experiments, the methodology that will be used in the remainder of the present chapter was implemented. The decision was made to describe the initial methodology and present its results as it was one of a couple of common evaluation methodologies we had rejected and the fact that we had already obtained results using it could be used to explain why.

The difference between the initial and the revised methodology is that the latter involves 1,500 tests of outcome and duration prediction accuracy for a single test

legislative procedure (test set of size 1) with all remaining 1,499 legislative procedures serving as the training sets.[4]

The order of the legislative procedures is entirely irrelevant as it is guaranteed that each legislative procedure will be tested in of the 1,500 tests and will be in the training data of all remaining 1,499 tests.

Moreover, given that for an actual prediction (not a test/evaluation prediction) about an ongoing legislative procedure, all available past data, namely all the 1,500 currently available completed legislative procedures, will be available, an evaluation methodology that uses training sets of 1,499 legislative procedures for its tests is closer to reality than an evaluation methodology that uses only two thirds (1,000) of the relevant past completed legislative procedure data available.

The present and the previous two sections dealt with the question of evaluation methodology. The methodologies examined were variants of well-known standard evaluation methodologies[5]:

- **Simple (Non-Cross-Validated) Hold-out Evaluation:** Involves splitting available data in test and training data. Each of Tests 1, 2 and 3 discussed as part of the Initial Evaluation Methodology could have served as an example of a methodology based on simple hold-out evaluation if its results were held to be definitive on their own (without conducting the other two tests).
- **K-Fold Cross-validation:** Involves splitting available data in Kpartitions and conducting K tests each using a different partition as the test data and the union of the remaining K-1 segments as the training data. This is the basis for both the initial (K=3) and the revised (K=1500) evaluation methodology.
- **Leave-One-Out (Exhaustive) Cross-validation:** The basis of the revised evaluation methodology; same as K-Fold Cross-Validation when K equals the size of the data set.

We will demonstrate the robustness of the revised methodology on the basis of two scenarios. In the original scenario, data are shuffled pseudo-randomly, whereas in the second scenario they are ordered by step count from highest to lowest.

Which legislative procedure is first, which one second, which one last etc. in the list of available data should not influence the result of the evaluation. Indeed, looking at Figure 10 and Figure 11, one notices that the Revised Evaluation Methodology produces the exact same outcome prediction accuracy results in both scenarios (77.89% overall).

[4]Because with 3 tests collecting and combining results semi-automatically is relatively painless, but for 1,500 tests it is not, additional implementation effort was required to make the collection and combination of results fully automatic.

[5]If all we were examining were predictions with no additional step information, we would indeed be using the standard versions of the hereby listed evaluation methodologies; what we do is slightly different though as we consider completed legislative procedures with their duration and outcome to be the items in the dataset, but do not test whether an outcome or duration prediction is correct for a legislative procedure, but rather for a legislative procedure at a specific stage of its progress (0, 1, …, N steps).

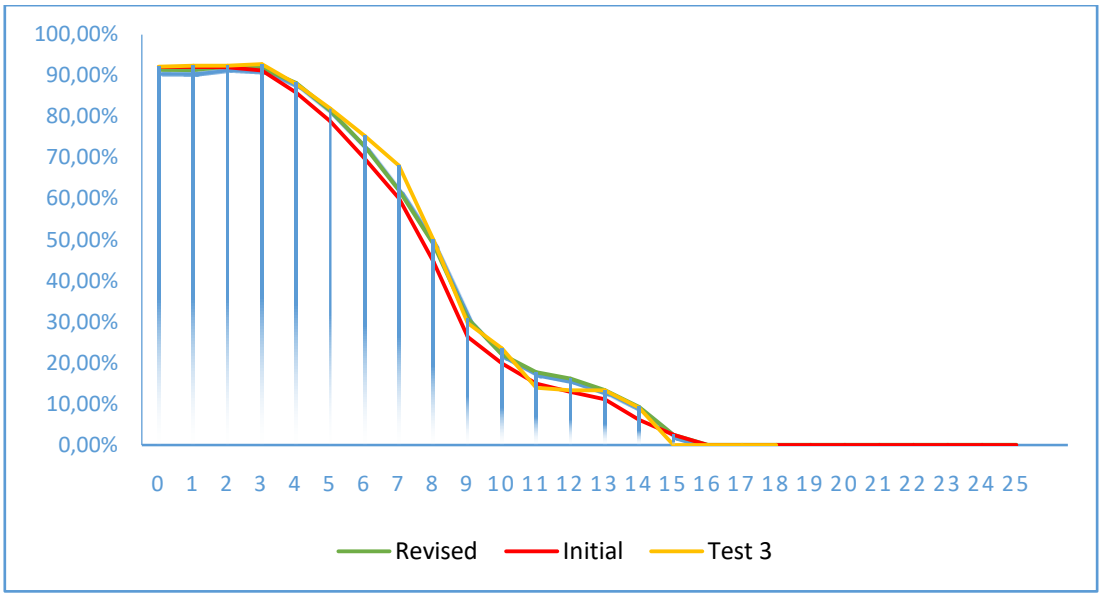

Figure 10: Statistical Predictor: Outcome Prediction Accuracy per Evaluation Methodology (Scenario 1)

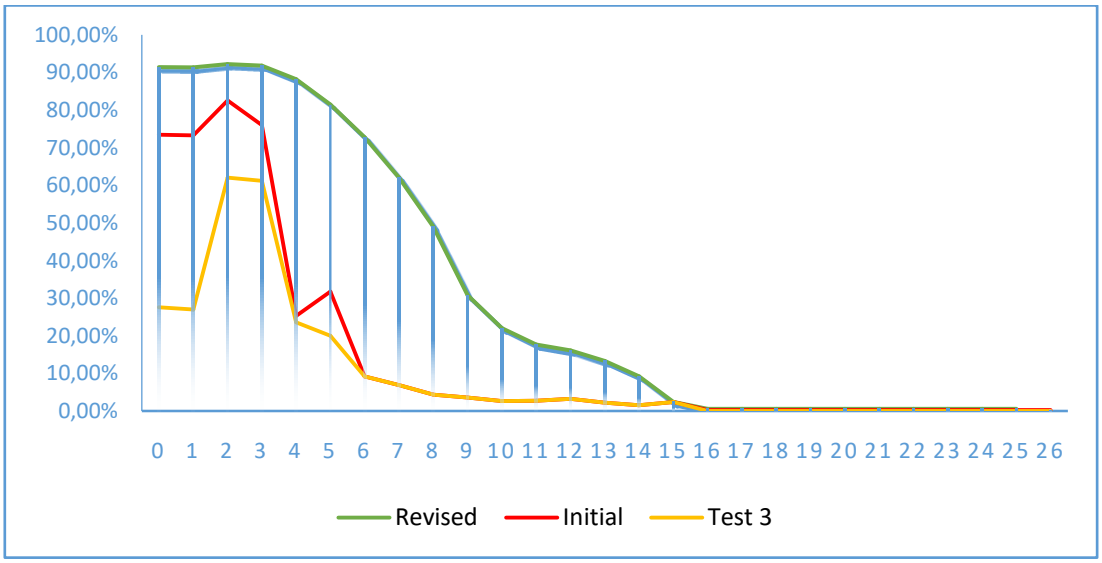

Figure 11: Statistical Predictor: Outcome Prediction Accuracy per Evaluation Methodology (Scenario 2)

The exact same observation can be made about Figure 12 and Figure 13 which display the three evaluation methodologies' duration prediction accuracy results in the two scenarios: the Revised Evaluation Methodology produces the exact same outcome prediction accuracy results in both scenarios (22.28% overall).

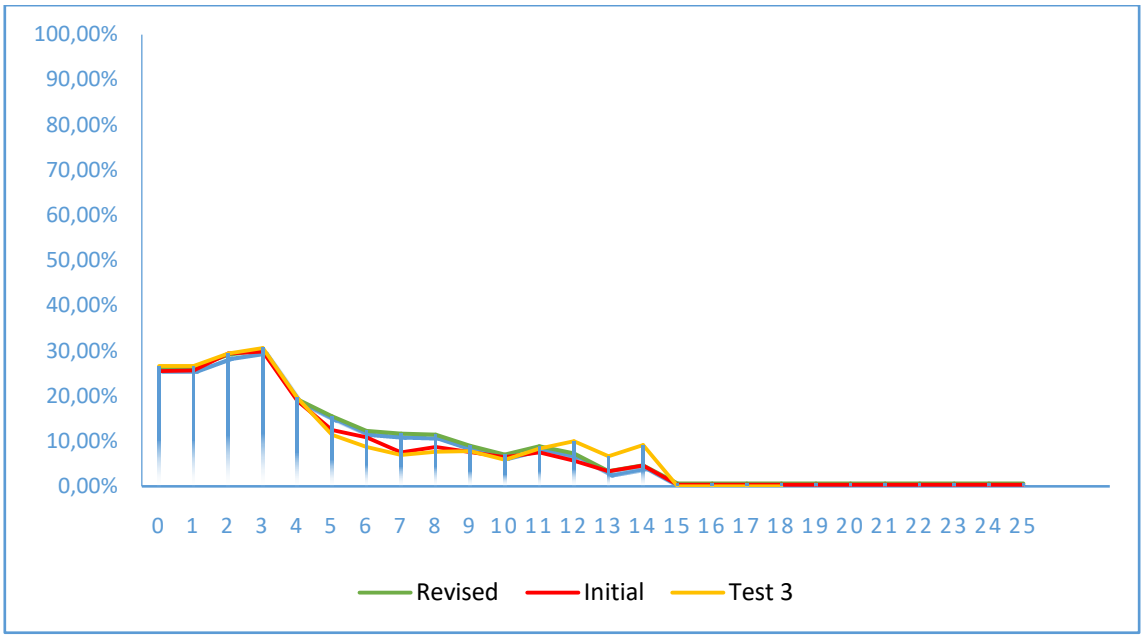

Figure 12: Statistical Predictor: Duration Prediction Accuracy per Evaluation Methodology (Scenario 1)

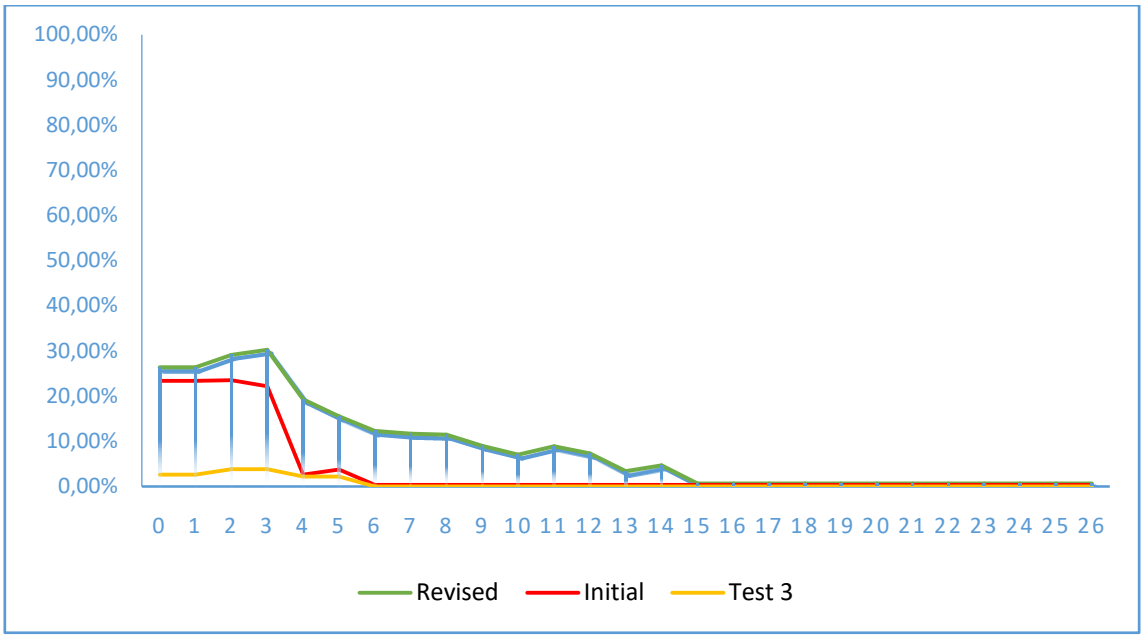

Figure 13: Statistical Predictor: Duration Prediction Accuracy per Evaluation Methodology (Scenario 2)

The Initial Methodology (based on 3-Fold Cross-Validation) does not achieve the same level of consistency across the two scenarios despite the fact that it combines results from three tests; if data happen to be ordered in one way (Scenario 1), it states that the outcome and duration prediction accuracies are 77.08% and 18.51% respectively, but if they happen to be ordered in another way (Scenario 2), it states that the outcome and duration prediction accuracies are 49.26% and 14.95% respectively. Same Statistical Predictor, same data, very different results.

As for the possibility of basing our evaluation methodology on Simple (Non-Cross-Validated) Hold-Out Evaluation, the inconsistency of results of Test 3 in the two scenarios clearly demonstrates the dangers of such a choice are even graver. Compare the discrepancies: 81.56% (Scenario 1; Figure 10) vs.16.40% (Scenario 2; Figure 11) for outcome prediction accuracy and 22.49% (Scenario 1; Figure 12) vs. 1.69% (Scenario 2; Figure 13) for duration prediction accuracy. Same Statistical Predictor, same data, entirely different results. There are three points to be made:

- As it turns out, our initial methodology produces similar results to the revised methodology; this means that the pseudo-random shuffling produced a favourable distribution of the available data.
- Both the Initial Methodology and Test 3 used as an example of Simple Hold-Out Evaluation are shown to be sensitive to a factor that should not have played any role, namely the ordering of the available data prior to its partitioning; however, the Initial Methodology, based on 3-Fold Cross-Validation is significantly less sensitive.
- The Revised Methodology takes cross-validation to its extremes and is entirely insensitive to the order of available legislative procedures (exactly as it should be).

Having demonstrated the key advantage of the revised evaluation methodology, we will now use it in investigations of prediction accuracies.

3.4 On the Accuracy of the Original Statistical Predictor

The Statistical Predictor has two questions to answer:

- What will the outcome of a legislative procedure X be?
- What will the duration of a legislative procedure X be?

So, the Statistical Predictor implements two separate functions, one delivering an outcome prediction and one a duration prediction, each with a different accuracy.

3.4.1 Specification of Original Statistical Predictor

The key underlying idea of the original specification was that two legislative procedures with a similar history (i.e. legislative procedures that have the same type and have gone through the same steps so far) are more likely to have the same outcome and duration than two randomly picked legislative procedures.

The history of a legislative procedure is formed by its type (e.g. Ordinary Legislative Procedure), followed by the steps that have so far been taken (EP Opinion on 1st Reading) together with the outcome of the step (e.g. Approval with amendments).

More specifically, the original Statistical Predictor delivers predictions about a legislative procedure X that so far has a history H by:

1. Selecting the set of all legislative procedures R in the training set that have histories that start with H (i.e. legislative procedures for which it can be said that H has followed in the procedural footsteps thus far)

2. Predicting that the outcome of X will be the most frequent outcome of legislative procedures in R (one of: Pass with Amendments, Pass without Amendments, Fail)

3. Selecting the set of all legislative procedures R' in R that have a total duration no shorter than the duration of X so far.

4. Predicting that the duration of X will be the most common duration of legislative procedures in R' (in months).

3.4.2 Prediction Accuracy and Evaluation

One way to appreciate whether the outcome prediction accuracy represents an achievement of some magnitude is to establish a baseline. There are three possible outcomes for legislative procedures. If they were equally likely, there would be a one in three change of a legislative procedure having one of them as their outcome, but they are not. Out of the 1,500 legislative procedures examined, 1,044 concluded with the proposal passing without amendments, 379 concluded with the proposal passing with amendments, and only 77 resulted in the proposal failing. The best performing of the following three constant predictors would be a suitable baseline predictor: a predictor always predicting a legislative process's outcome is "Pass without amendments", a predictor always predicting a legislative process' outcome is "Pass with amendments", and a predictor always predicting a legislative process' outcome is "Fail".The three aforementioned constant predictors achieve outcome prediction accuracies of 47.86%, 46.65% and 5.49% respectively. Therefore, it is the constant predictor that always predicts "Pass without amendments" that sets the bar; any outcome predictor that does not achieve over 47.86% outcome prediction accuracy is no better that the constant predictor that always predicts "Pass without amendments".

The original Statistical Predictor achieves an outcome prediction of 77.89%. This is a prediction accuracy that is significantly higher than the 47.86% mark. This signifies that there is merit to the approach outlined in D4.3.1 for legislative procedure outcome prediction.

The duration of legislative procedures is far less predictable; the shortest concluded within less than 15 days since the day they commenced and the longest took over117 months, at least according to the way we compute a legislative procedure's duration which involves taking the difference (in months and using rounding) between the date of the last step and the date of the first step in a legislative procedure.[6]The majority of legislative procedures take less than 6 months to complete (871 out of 1,500), two thirds take less than 12 months to complete (1,083 out of 1,500) and all but around 200 take less than 24 months to complete (1,308 out of 1,500).The best performing constant predictor in this case

[6] One good example for calling our choice into question is the legislative procedure for ACTA (see Figure 4); the relevant proposal was adopted by the Commission in June 2011 and was rejected by the Parliament in July 2013, but the last step is the withdrawal by the Commission in March 2015. While we did contemplate an alternative algorithm that would consider the rejection rather than the withdrawal date as the date marking the end of such legislative procedures we decided against it as (a) only 4 out of 1,500 legislative procedures exhibited this pattern and (b) there were cases where after a Rejection steps other than a withdrawal were taken which we took to support the argument that a withdrawal happens when it happens for a reason and that it has its own significance which should not be hastingly ignored.

would be one that always predicts a duration of 2 months; it would have an accuracy of 15.64%.

The original Statistical Predictor achieves a duration prediction of 22.20%. This is a prediction accuracy that is significantly higher than the 15.64% mark, despite being quite low.

While there is merit to the approach outlined for legislative procedure duration prediction, the accuracy achieved would seem to indicate that predicting the duration (in months) of a legislative procedure in this manner is unlikely to be a useful feature as users will not be able to rely on it.

This is not entirely true though, as for duration prediction there is a degree of tolerance. If we allow a +/- 1 month tolerance in our evaluation, accuracy rises to 34.42%. With a +/- 2 months tolerance, accuracy rises to 41.78%. Henceforth, we will allow a +/- 3months tolerance. As a result, accuracy rises to 47.29%.While nowhere close to the outcome prediction accuracy, duration prediction can be accurate enough to give users a prediction that is accurate enough to serve as an indication of how long the legislative procedure is going to take.

The present deliverable aims not to simply report on the accuracy of the original Statistical Predictor but also to find ways of improving the Statistical Predictor on the basis of data and tests that were not available at the time its initial design was being proposed. The previous section focused on the evaluation methodology, the present section on examining whether the initial Statistical Predictor's design had merit, and the following section will present improvements made as part of T4.3.

3.5 Revised Statistical Predictor

One of the aims of T4.3 was to produce a specification of an improved Statistical Predictor, as the result of testing and adaptations on the original one as it was specified originally in T4.2. The original Statistical Predictor of Section 3.4.1 (also presented in T4.2 deliverables) attempts to solve two problems: to provide outcome *and* duration predictions for ongoing legislative procedures. It achieved prediction accuracies of 77.89% and 47.29% (with a +/- 3 months tolerance for duration accuracy) respectively.

A number of variations of the original algorithms were tested as part of the current phase of the testing and adaptations task; see Appendix C and Appendix D. For most of them, it can be proved mathematically (without requiring tests on real data) that they will always make predictions that will be at least as accurate as the corresponding (outcome or duration prediction) original algorithm's. In some cases, it is also possible to prove one variation as better than another. Whether theoretical superiority can be established or not, testing on real data serves to demonstrate how much better one variation performs than another or than the original algorithm.

The reason the variations of the original algorithms were thought of, implemented, and tested was so that they could replace the original algorithms, resulting in what we call the Revised Statistical Predictor (more accurately the revised first prototype of the Statistical Predictor). On the basis of the results obtained for the various variations we tried, we have exactly that.

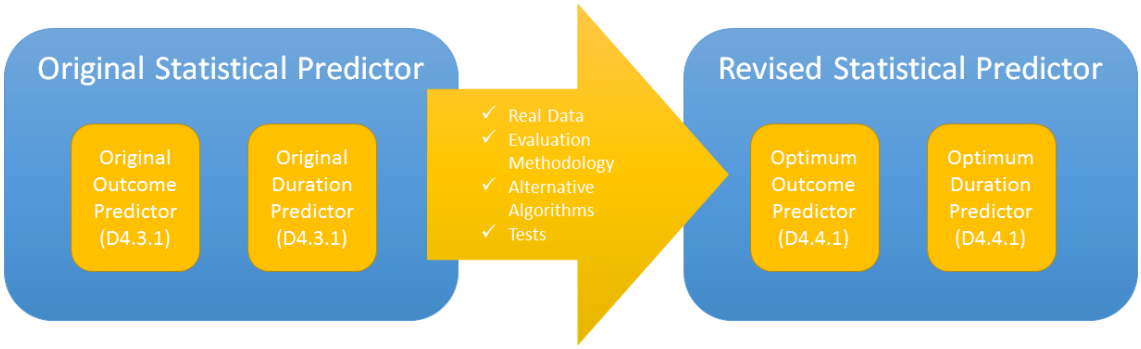

Figure 14: From the Original to the Revised Statistical Predictor

The Revised Statistical Predictor uses Variation O4 (with parameter a strength threshold of 2) (see Appendix A) of the original outcome prediction algorithm for outcome prediction and achieves an accuracy of 91.50%. For duration prediction, it uses Variation D5 (with a maximum fallback step of 4) (see Appendix B) of the original duration prediction algorithm and achieves an accuracy of 56.26% for duration prediction with a +/- 3 months tolerance.

Because decisions were based on tests on real data, there may be a concern that had the data been different, perhaps different variations and/or different parameters for the same variations would have been chosen. If another set of specially crafted artificial data were to be tried, indeed, it is possible that the chosen variations could perform worse than others. If another set of real data of similar characteristics was used instead, most probably the chosen variations would perform similarly well. If a set of real data of otherwise similar characteristics to the current one but five times the size was used, perhaps some of the choices made here could turn to be sub-optimal, but it would be unlikely that the difference in prediction accuracy would be significant.

The fact of the matter is that while concerns about data-sensitivity are valid, they can be rendered immaterial. A solid evaluation methodology has been devised which can decide which variation is best and with what parameterisation, in a matter of minutes. A legislative procedures crawler has also been implemented in order to provide the data on which the current discussion of the Statistical Predictor is based; this can be used to record data on newly completed legislative procedures. Combining these capabilities, it is possible to have a Statistical Predictor that:

- Learns about newly completed legislative procedures every day and, as a result, can have the entire set of completed legislative procedures as its training set when faced with the task of making predictions for an ongoing legislative procedure
- Automatically evaluates the available competing outcome and duration prediction algorithms on the basis of all available data at the time, recording the optimum choice of algorithms and parameters for when a prediction is to be made
- Uses the outcome and prediction algorithms that have been determined to be optimal on the basis of the above step (with their optimal configuration parameters, if any) when asked to make a prediction about an ongoing legislative procedure.

Such a Statistical Predictor will be only as good as its best alternative prediction algorithms. For the currently available dataset, which was frozen for the purpose of experimentations and evaluations that were taking place in a semi-automatic manner, the optimal variations are the aforementioned ones.

3.6 Prediction Accuracy and Beyond

The Statistical Predictor was proposed on the suspicion that the described superficial similarity-based prediction methodology would work well for predicting the outcome of legislative procedures. It was expected that the granularity, range and distribution of the possible answers to the question of how long a specific legislative procedure is going to take to complete would make having acceptably accurate duration predictions significantly harder, but that it would be beneficial to the project to attempt duration predictions in spite of their inherent difficulty.

It is evident from the outcome accuracy scores of both the original and the revised Statistical Predictor (77.89% and 91.50% respectively) that our first suspicion was correct.

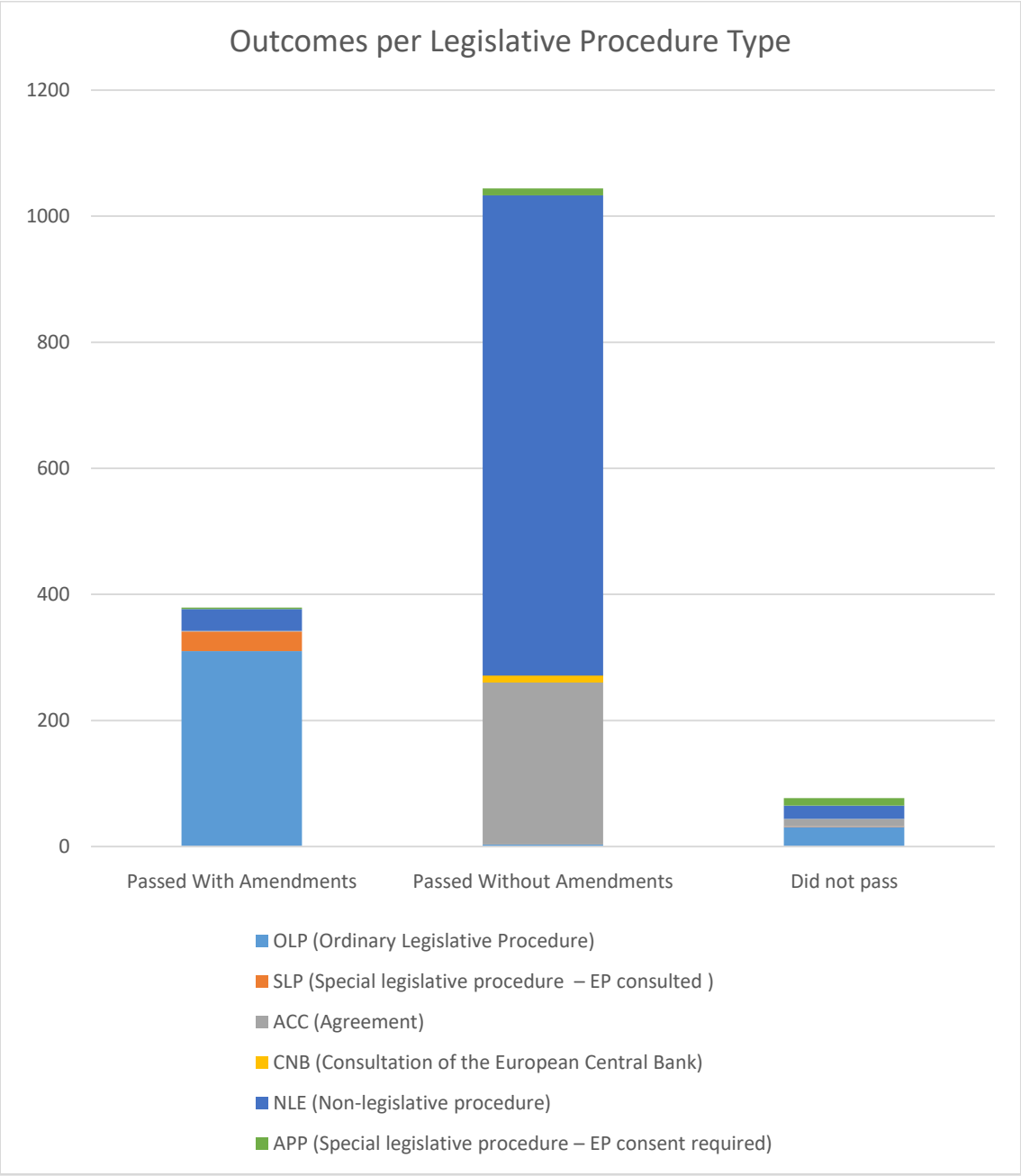

Figure 15: Outcome Frequency and Legislative Procedure Types

Both the original and the revised Statistical Predictor have very high initial outcome prediction accuracies (91.40%). This may seem somewhat odd given that very little is known about legislative procedures when they are only starting out, but in fact the type of a legislative procedure, which is the first thing recorded in a legislative procedure's history, is quite informative. More specifically, there is a very high probability for a certain outcome in each of the three most common legislative procedure category types:

1 The outcome of 762 out of 818 (93.15%) of Non-legislative (NLE) procedures was "Pass without amendments".

2 The outcome of 310 out of 344 (90.12%) of Ordinary Legislative Procedures (OLP) was "Pass with amendments".

3 The outcome of 258 out of 271 (95.20%) of Agreement (ACC) procedures was "Pass without amendments".

Therefore, even just knowing the legislative procedure type is enough to make a fairly confident guess of its outcome.

The reason for the rapid decline in the accuracy of the original Statistical Predictor as the step count increases is that the longer a legislative procedure's history is, the more unique it is also; this means that that as the step count increases, the chance that there will be a shortage of relevant training data also increases.[7]

The Revised Statistical Predictor does a very good job at addressingthe cases where the original Statistical Predictor fails to make a prediction. The result is not only a higher overall accuracy, but more importantly a consistently high accuracy across different step counts (see Figure 16).

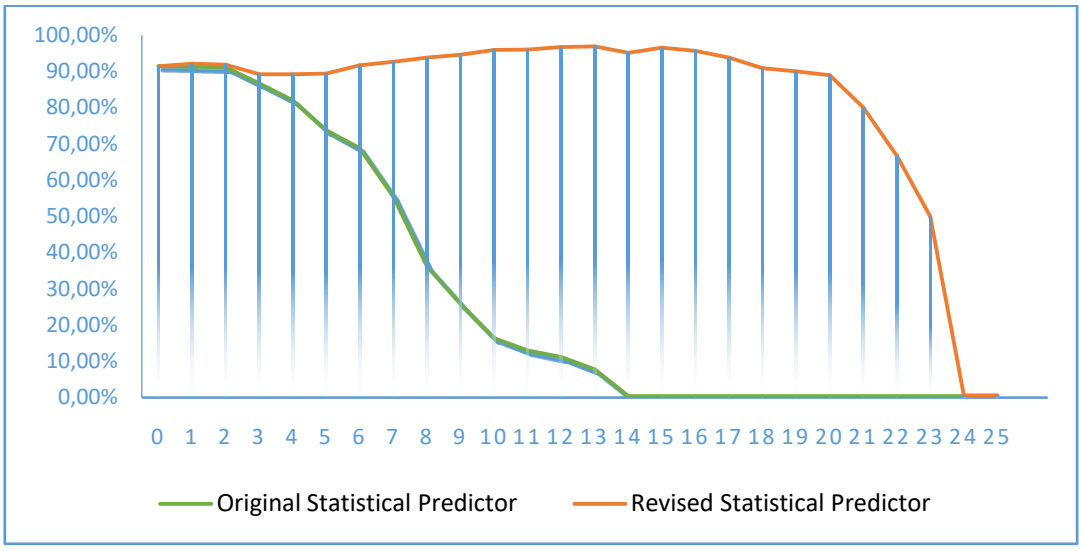

Figure 16: Output Prediction Accuracy: Original vs Revised Statistical Predictor

[7] In 1,117 out of the 8,088 test cases it is requested to provide outcome predictions for, the original Statistical Predictor fails to do so due to lack of relevant training data. The problem starts becoming noticeable at step 4 (51 out of 650 test cases receiving no outcome prediction), peaks at step 8 (143 out of 224) and remains till the end (for step count 25, the one test case receives no prediction for this reason). There are additionally 69 test cases for which the original Statistical Predictor produces no prediction not because there are no relevant training data but because it cannot decide between two or three equally supported outcomes; this problem was far more common for small step counts (<6).

Table 5: Revised Statistical Predictor: Outcome Prediction Accuracy per Step Count

Step Count	Outcome Predictions Requested	Correct Outcome Predictions		Outcome Prediction Accuracy	
		Original Statistical Predictor	Revised Statistical Predictor	Original Statistical Predictor	Revised Statistical Predictor
0	1,500	1,371	1,371	91.40%	91.40%
1	1,500	1,369	1,382	91.27%	92.13%
2	1,500	1,366	1,378	91.07%	91.87%
3	889	770	793	86.61%	89.20%
4	620	508	553	81.94%	89.19%
5	433	319	387	73.67%	89.38%
6	361	248	331	68.70%	91.69%
7	298	165	276	55.37%	92.62%
8	224	81	210	36.16%	93.75%
9	186	48	176	25.81%	94.62%
10	147	24	141	16.33%	95.92%
11	124	16	119	12.90%	95.97%
12	90	10	87	11.11%	96.67%
13	65	5	63	7.69%	96.92%
14	41	0	39	0.00%	95.12%
15	29	0	28	0.00%	96.55%
16	23	0	22	0.00%	95.65%
17	16	0	15	0.00%	93.75%
18	11	0	10	0.00%	90.91%
19	10	0	9	0.00%	90.00%
20	9	0	8	0.00%	88.89%
21	5	0	4	0.00%	80.00%
22	3	0	2	0.00%	66.67%
23	2	0	1	0.00%	50.00%
24	1	0	0	0.00%	0.00%
25	1	0	0	0.00%	0.00%
TOTAL	**8,088**	**6,300**	**7,405**	**77.89%**	**91.56%**

Indeed, the Revised Statistical Predictor's outcome accuracy only drops significantly below 90% for step counts greater than 19. The fall is quite sharp, but at the same time without a significant impact as for instance for step 22 where accuracy drops to 66.67%, this corresponds to one of three test cases receiving the wrong prediction; likewise, the 50% accuracy of step 23 corresponds to failure to predict the outcome of one of two test cases and the 0% accuracy for step counts 23 and 24 to failure to make a correct prediction for the only test case that had that many steps.

The unavailability of relevant training data also causes a significant drop in the duration prediction accuracy of the original Statistical Predictor as the step count increases. An additional problem is that the range of possible answers also tends to increase as the step count increases. The duration prediction problem is significantly harder even with a +/-3 months imprecision tolerance. Again, the Revised Statistical Predictor not only provides a better overall accuracy, but quite importantly retains its accuracy levels as the step count increases.[8]

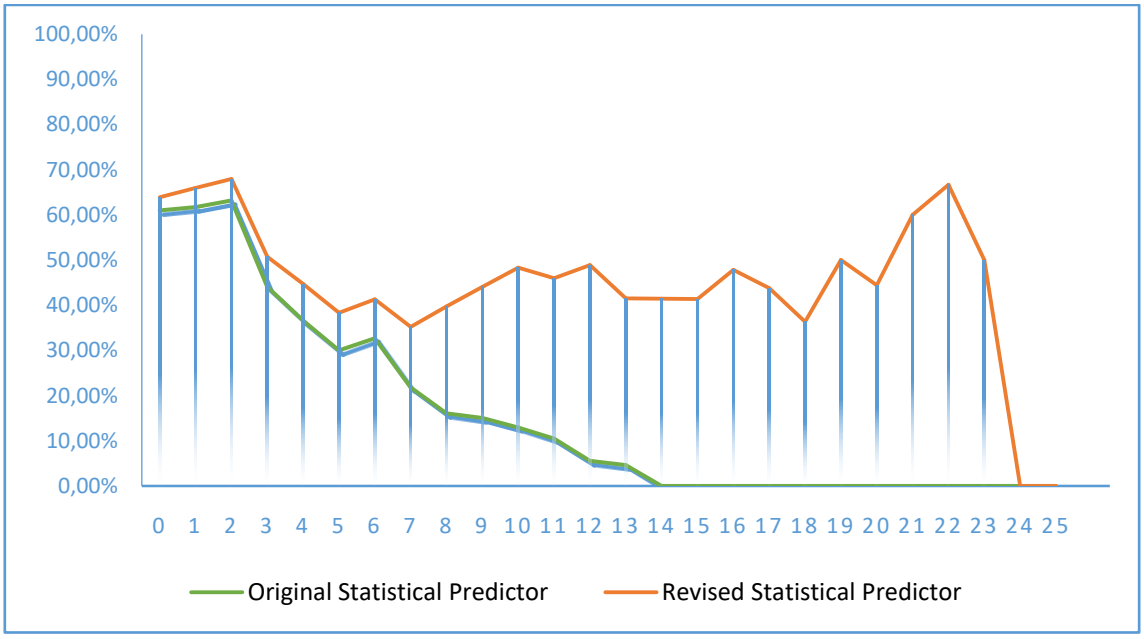

Figure 17: Duration Prediction Accuracy: Original vs. Revised Statistical Predictor

[8] The temporary sudden jump in accuracy after step 20 is not any more significant than the sudden drop of outcome accuracy, as results for step counts above 20 concern only a very small number of test cases (see Table 6).

Table 6: Revised Statistical Predictor: Duration Prediction Accuracy per Step Count

Step Count	Outcome Predictions Requested	Correct Outcome Predictions		Outcome Prediction Accuracy	
		Original Statistical Predictor	Revised Statistical Predictor	Original Statistical Predictor	Revised Statistical Predictor
0	1,500	915	958	61.00%	63.87%
1	1,500	926	989	61.73%	65.93%
2	1,500	948	1,019	63.20%	67.93%
3	889	392	451	44.09%	50.73%
4	620	227	277	36.61%	44.68%
5	433	130	166	30.02%	38.34%
6	361	118	149	32.69%	41.27%
7	298	65	105	21.81%	35.23%
8	224	36	89	16.07%	39.73%
9	186	28	82	15.05%	44.09%
10	147	19	71	12.93%	48.30%
11	124	13	57	10.48%	45.97%
12	90	5	44	5.56%	48.89%
13	65	3	27	4.62%	41.54%
14	41	0	17	0.00%	41.46%
15	29	0	12	0.00%	41.38%
16	23	0	11	0.00%	47.83%
17	16	0	7	0.00%	43.75%
18	11	0	4	0.00%	36.36%
19	10	0	5	0.00%	50.00%
20	9	0	4	0.00%	44.44%
21	5	0	3	0.00%	60.00%
22	3	0	2	0.00%	66.67%
23	2	0	1	0.00%	50.00%
24	1	0	0	0.00%	0.00%
25	1	0	0	0.00%	0.00%
TOTAL	**8,088**	**3,825**	**4,550**	**47.29%**	**56.26%**

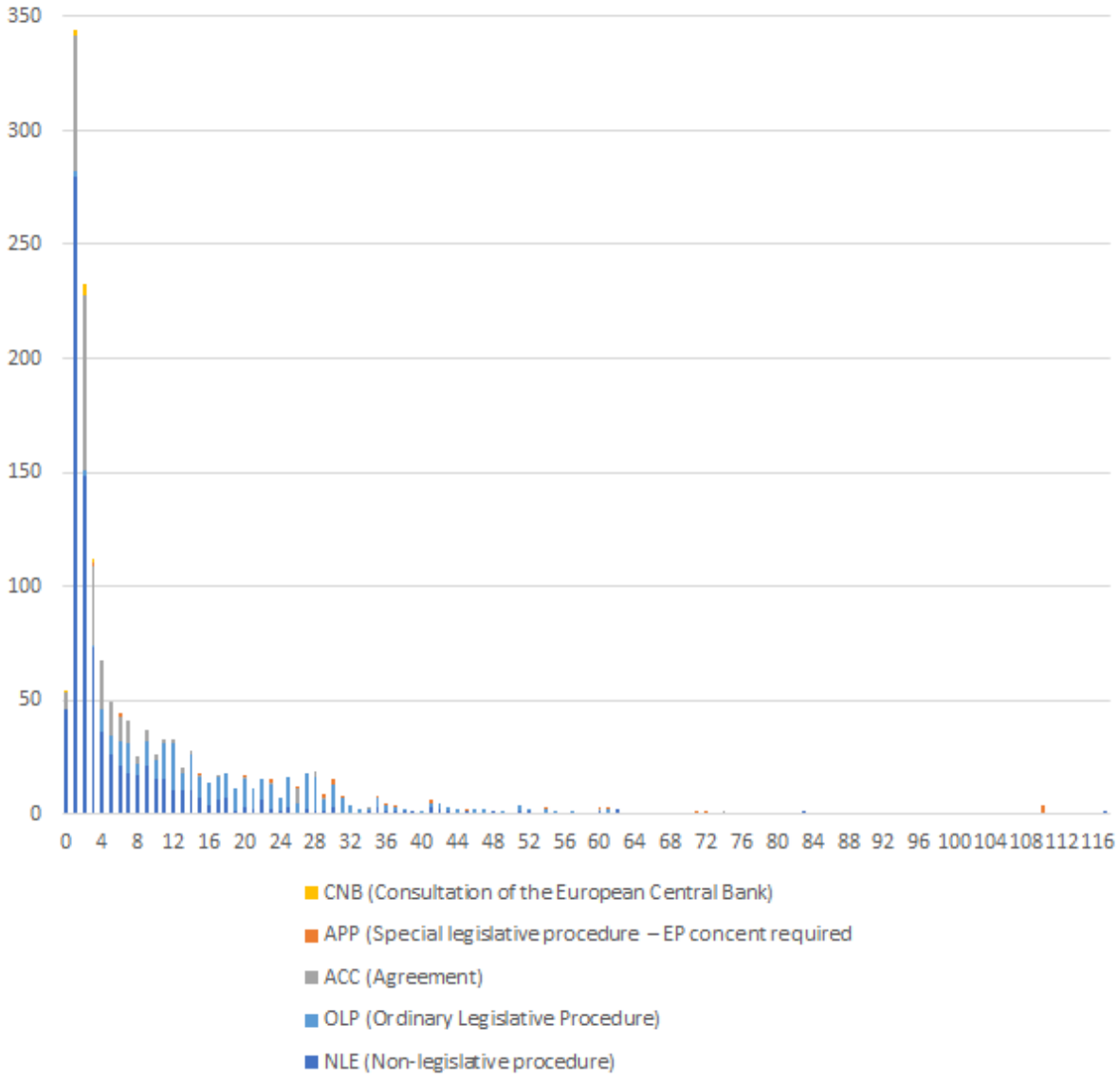

Figure 18: Duration (months) frequency per Legislative Procedure Type

We consider the achieved accuracy scores to be quite an important achievement, even with a +/- 3 month tolerance given the inherent difficulty of the task (partly captured in Figure 18). The fact that accuracy is only 56.26% and that this figure assumes a +/-3 imprecision tolerance, means that these predictions are not to be relied on for certain uses; on the other hand, they could prove to be a useful tool for other uses, so long as their limitations are understood and it is ensured that they are fit for the purpose they are used for.

By contrast, the outcome prediction accuracy is particularly high. One concern we had, was that it might be overly biased towards certain outcomes, and specifically a would perhaps fail to make correctly predictions "Fail" outcomes, as they represent the correct prediction for only a fraction of the test cases (444 out of 8,088) making it easy to score high overall without ever correctly predicting a "Fail" outcome. That concern was laid to rest as, the Revised Statistical Predictor correctly predicted the "Fail" result in 363 out of the 444 relevant test cases (i.e. in 81.76% of the

cases).While this does indicate a certain bias against producing the "Fail" prediction, it is still within reasonable limits. It is expected that if input from policy experts is appropriately combined with statistical predictions, this bias will be eradicated or at least significantly minimised.

4 Hybrid Predictions Subsystem

4.1 Introduction

The Hybrid Simulation Subsystem was not foreseen in the DoW, but rather came about as a result of:

- a Consortium/WP4 initiative, first mentioned in D4.2, to add to the Policy Component a prediction system for legislative procedures based on past EURLex data (the Statistical Predictor), and
- a very stimulating discussion between the Reviewers and the Project Partners during the Year 1 review during which the concept of combining expert knowledge with past data emerged; the design and implementation work aimed at turning this concept into a predictions engine, the Hybrid Simulation Subsystem, was described in D4.3.1 and D4.3.2.

The Statistical Predictor can be used on its own or as part of the Hybrid Simulation Subsystem. D4.3.1 only dealt with the evaluation of the Statistical Predictor. Both the Statistical Predictor and the Hybrid Predictions Subsystem are examined here.

4.2 Human Experts Input

The big advantage of the Hybrid Simulation Subsystem over the Statistical Predictor, namely that it takes into consideration input from human experts that may have specific insights about a specific legislative procedure, not merely statistical knowledge about previous legislative procedures poses additional challenges.

The evaluation and adaptations of the Statistical Predictor proceeded on the basis of large quantities of data on past completed legislative procedures. The aim of the evaluation and of the adaptations was accuracy. For the Statistical Predictor, we examined over 8,000 outcome and duration predictions concerning each and every stage of the 1,500 completed legislative procedures that formed the dataset we worked with. An evaluation of the same kind for the Hybrid Simulation Subsystem will be possible only when a significant number of human predictions will become available.

Interesting questions can be raised when human expert opinions coexist with a statistics based methods:

- Are human experts better at making predictions than the Statistical Predictor?
- Is the Hybrid Predictions Subsystem better than the Statistical Predictor?

The answer to these questions will depend both on who makes the predictions and based on what knowledge and how well the Hybrid Prediction Subsystem utilises the input it gets. The HPS has been defined using game concepts aiming to gradually make predictions by more knowledgeable human experts weight more.

4.3 Raw, Normalised Score and Reputation

Turning outcome and duration predictions into games in D4.3.1 and D4.3.2 came with having a score for those games. This score may not have an upper and lower bound, or it may have a lower bound of 0, depending on the choices discussed below. When we need to distinguish this score from the normalised score, we will be referring to it as the *raw score*.

The *normalised score* is a score proportionally calculated from the raw score, ensuring that it fits in the range 0 to 100. So if the maximum raw score is 1000, the minimum raw score is 0 and a player's raw score is 500, his/her normalised score will be 50.

There are two games, the outcome and the prediction game, so there are two sets of raw scores and two sets of normalised scores. A new criterion was added in the third prototype of the Reputation Management System (D3.2.3) that would allow reputation to depend on the success of a policy expert on the prediction games. This is a weighted average of the normalised outcome predictions score and the normalised duration predictions score (which presently has two equal weights).

4.4 The Effect on Correct and Incorrect Predictions

In D4.3.1 and D4.3.2, we presented one way in which predictions affect a player's score. Here we will take one step back and consider both the choices we made and alternative choices we did not. While no specific changes have resulted from the exploration of these alternatives (i.e. they are not adaptations, but rather potential adaptations), presenting these alternatives helps bring an understanding of the choices made and their consequences.

A prediction can be successful or not. Obviously, a correct answer will have a positive impact on the player's score. One question is whether a wrong answer results in the player score's being lowered. Penalising wrong predictions has the effect of discouraging predictions a player is not sufficiently sure of, which would mean fewer butmore reliable inputs.

If wrong predictions are to be penalised, another question is by how much. Should they result in the player losing the same amount of points they would stand to win if they had been correct, fewer or more?

Another feature of the game could be that more points are awarded to a player that makes a prediction "against the odds". This would be the case, for example, if the likelihood of a prediction (as specified D4.3.1 and D4.3.2) is 90% and a correct prediction is rewarded with only 10 points whereas a correct prediction with a 25% likelihood would receive 75 points. The alternative is that all predictions have the same value, and if a correct prediction is made, say, 100 points are always awarded irrespective of the prediction's likelihood. The latter alternative is conceptually simpler for the players and does not rely on a concept of prediction likelihood, whereas the former, in the long run, rewards those players who give predictions about ongoing legislative they understand better than most.

D4.3.1 and D4.3.2 introduce an additional factor in the computation of the score increment for successful predictions. According to the proposed mechanism therein, it makes a difference if the same prediction is made by one or more human experts. In particular, if a prediction that would award the single human expert that

made it 95 points was made by five human experts, this would lead to them being awarded 95/5=19 points. The rational for this adjustment was that if many experts make the same prediction, this would mean that there may be common knowledge that this is going to be the desired outcome, therefore the value of the prediction is not as high as it would have been if fewer experts made it. In the case that a mistaken prediction is made by many experts, the effect of this design would be that they would be penalised less than they would otherwise have been. This seems to us like an interesting feature of the scoring system, but there seems to also be merit in the simpler alternative of not trying to scale down the value of a prediction on the basis of how many experts made this prediction.

To summarise, we have considered here a number of alternatives. As noted earlier, we decided not to change the HPS changes players' scores on the basis of correct and incorrect decisions. The choices we made were:

- Incorrect predictions are penalised which has the effect of discouraging predictions a player is not sufficiently sure of, which should translate into fewer but more reliable inputs.
- An expert's score is decreased by an incorrect prediction by the same number of points as it would have been increased had the prediction been correct. So given the net effect on the score of two cases of predictions identical in every aspect except that one is correct and the other one is not would be zero.
- The HPS rewards more, correct predictions that go "against the odds". This way, when an expert correctly makes a prediction which the Statistical Predictor would consider a bad choice, exemplifying the key advantage experts may have over statistical predictors based on past data i.e. knowledge about the specific legislative procedure, they are rewarded with more points than when they simply agree with the statistical predictor's view of things. On the other hand, given the two above choices, if a user makes a wrong prediction which the Statistical Predictor would consider a bad choice, he/she is more severely penalised for making a bad prediction than for making a wrong prediction that the Statistical Predictor would have considered a good one.
- If the correct prediction is made by N users, its value is shared between them. This means that those who know something others do not know are rewarded more for their correct prediction than those that know something everybody knows. Likewise, to make a mistake that many other users make will result in a smaller negative effect on their score than making a mistake few or only they made.

Formalising the above, for each answer, its value $v = 100 \times (1\text{-}p)$ where p is the probability of the answer on the basis of past data statistics and for each player that answered it correctly, v/c points are added to their score (where c is the number of players that answered it correctly), whereas for each player that answered it incorrectly, v/i points are subtracted from their score (where i is the number of players that answered it incorrectly).

Note: One additional factor that we considered and was indeed proposed also by the WP4 External Experts was the confidence a user attributes to their own

prediction. This would require additional input, which we initially wanted to avoid. Additional input generally has the effect of decreasing the number of responses obtained when requesting data from users, but in this particular instance we believe it would encourage users to provide predictions as without being able to provide an indication of their certainty they may be discouraged from making a prediction. This confidence factor,*f*, which could take values from 0 to 1, would be used to compute how much the player is rewarded or penalised respectively for a correct or incorrect prediction. With this additional factor, a player that made a correct prediction with confidence *f would have f×v/c* points added to their score (where *c* is the number of players that made the same correct prediction), whereas for each player that made an incorrect prediction, *f×v/i* points are subtracted from their score (where *i* is the number of players that made the same incorrect prediction).

4.5 Combining Human Expert Predictions

When instead of relying on the Statistical Predictor, the aim is to produce a prediction by means of combining all the available predictions of human experts, there is a question of how this is to be done. The problem here is simple: the human experts will more often than not make different predictions, so a choice will have to be made.

D4.3.1 and D4.3.2 proposed a way for making this choice. Here we will consider also alternatives.

The simplest algorithm is C1:

Choose the most heavily supported prediction (on the basis of the number of players making each prediction)

The algorithm proposed in D4.3.1 and D4.3.2 is C2:

Choose the most heavily supported prediction (on the basis of the sum of scores of players making each prediction)

Another algorithm proposed discussed by the consortium C3:

Choose the most heavily supported prediction (on the basis of the sum of the reputation of players making each prediction)

A fourth option, which subsumes C2 and C3, uses both the predictions score and the overall reputation score of human experts, is expressed as C4:

Choose the most heavily supported prediction (on the basis of the sum of the weighted average of the normalised prediction score and the reputation of players making each prediction). Note: The weight of the reputation and the normalised predictions score can be user-configurable parameters.

The newly introduced alternatives could be implemented instead of the originally specified prediction combination algorithm. It would be interesting to see their relative performance (and in the case of C4 try different weights).

As part of the work on adaptations on the HPS, C2 was replaced by C4 (with equal weights for the reputation and the normalised predictions score). The reason for this adaptation/improvement is that it allows the HPS to place more confidence on experts with a higher reputation when the first predictions are made in which case

the prediction score is 0 or based on very few predictions. It also allows the weights to be used to change the relative emphasis of reputation and normalised prediction score, so that the HPS can be optimised in accordance with the experience gained from its fine-tuning.

4.6 Combining Human Expert and Statistics-Based Predictions

Another adaptation/improvement brought forward in the current deliverable is the change of the way human expert and AI predictions are combined.

The problem with the original specification of the HPS is that all players are treated in exactly the same manner. However, there is a fundamental difference in the behaviour of human agents and the Statistical Predictor and similar AI players: whereas a human expert may produce some predictions, he/she is unlikely to provide a prediction for all ongoing legislative procedures at their every stage; the Statistical Predictor can do exactly that. The consequence is that the Statistical Predictor, assuming it continues to be successful more than half of the time will soon start having a much higher score than any human player, potentially to a degree that even if all human experts make prediction A and the Statistical Predictor makes prediction B, the Hybrid Prediction Subsystem will choose to make prediction B.

There could be ways that would ensure that a large number of successful predictions would not lead to large scores, but it would not be desirable to have a solution that avoids the aforementioned problem but does not work well for human players.

A simple way of balancing the opinion of human experts against the opinion of AI players such as the Statistical Predictor is to follow a two stage process:

Stage 1:

- Combine the different predictions human experts have made using algorithm C4 described in the previous section.
- Combine the different predictions AI players such as the Statistical Predictor have made using algorithm C2 described in the previous section.

Stage 2:

- Combine the above two predictions treating them as coming from two higher-level players, HUMAN-EXPERTISE and AI (for whom scores are kept separately than for ordinary players), using algorithm C2 described in the previous section.

4.7 External Experts' Feedback

In order to get an outside perspective and evaluation of the ideas that form the cornerstone of the Hybrid Predictions Subsystem's algorithmic design, we enlisted the help of the EU Community External Experts.

Most experts agreed with practically all design choices in the HPS.

Two designs choices proved particularly controversial:

- The suggestion that the HPS would be improved by allowing the user to indicate their confidence in the prediction they are submitting, was warmly accepted by most but not all external experts.
- The choice to deduct an equal amount of points for a wrong answer as the player would stand to win had the answer been correct; whereas technical experts seem to agree with the design decision taken in the HPS, policy experts tended to prefer to have a smaller or a zero penalty for wrong predictions.

One design choice was not warmly received by most experts:

- The choice to divide the worth of a prediction by the number of users that made it, received some support, but was attacked more than any other design decision. Probably the most compelling argument against it is that it minimizes the effect on the score of experts providing predictions in the case of predictions on 'hot' topics (where traffic will be higher and the number of predictions too). In retrospect, it may have been better to aim to achieve the goal we aimed to achieve by dividing by the percentage rather than the number of experts that chose the given answer.

Overall, despite some disagreements with specific design decisions, the majority of the experts found the HPS design to be a good one and agreed that believe the best way forward for the HPS would be continued evolution in the context of a further research project.

5 Simulation Subsystem

5.1 Introduction

The Simulation System is meant to take a snapshot of the current status of the discussion on a policy process and simulate its future evolution by means of iterating over a System Dynamics model transforming a state X_t into a state X_{t+1} in each iteration, on the basis of specific assumptions embodied in a System Dynamics Model. One particularly interesting aspect of EU Community is that its expert users may also enter information about future publications of documents or future events into the EU Community platform database; these will be valuable sources of information for the Simulation Subsystem's predictions and as such will also be taken into consideration.

A state is defined by a number of variables (Documents, Awareness, Engagement, etc.). Taking the value of a variable across states X_0 to X_N produces a time series for that variable.

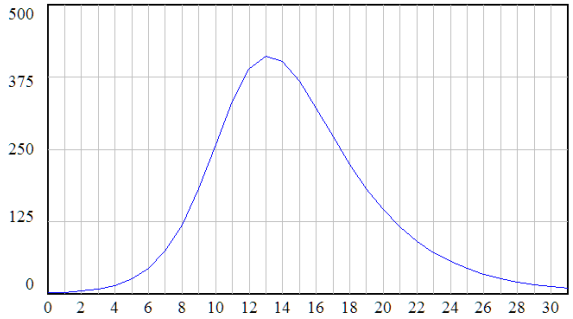

Figure 19: Example time series output (new documents per day)

For example, by providing appropriate initial values to the System Dynamics model, executing the simulation with 30 iterations (each iteration step corresponding to a day passing), and taking the time series predicting the number of new documents added in the discussion each day, the time series data depicted as a graph in Figure 19 was obtained. According to the hypotheses of the current model and the initial values given to its parameters, at the beginning, a small number of authors write about the topic but as documents are published, the amount of people/organisations becoming aware and engaged increases; however, after some days/weeks/months the interaction decreases, due to the phenomenon of saturation, as discussants have expressed their opinions and have nothing to add on the topic in question.

The output of the Simulation Subsystem, is a number of time series, one for each of variables to be visualised.

5.2 Testing the Execution Engine

As explained in D4.3.1, at the very core of the Simulation Subsystem is a System Dynamics model, which is meant to determine its behaviour and results. The model

is an MDL-format specification of how a state X_t can be transformed into a state X_{t+1}. An **execution engine** is needed to load, decode the model, and, given appropriate data input specifying the initial state and the number of steps N to be taken into the simulation, enact the model's state transformation specifications in order to produce a sequence of states X_0 to X_N.

The execution engine is implemented in Python on the basis of PySD[9]. The tests performed on the PySD-based execution engine were meant to ensure:

1. that it executes the model correctly, and
2. that it does not suffer from long running times or excessive memory usage issues that could render it problematic and subject to replacement despite being correct.

5.2.1 Correctness Testing

The Simulation Subsystem's System Dynamics model has been created using Vensim[10]. Vensim includes an execution engine meant to be used interactively by the model designer e.g. to test hypotheses, examine the behaviour of the model as certain input values increase or decrease etc. To test the Simulation Subsystem correctness, we performed tests checking its results' consistency with the results produced by Vensim.

The tests involved checking the consistency of results of the Simulation execution engine with the reference execution engine (i.e. Vensim's execution engine) on two output variables (engagement and document volume) for ten different sets of initial values and on all output variables for two output variables, for 60 iteration steps. Results were deemed consistent if and only if all values in the time series of each tested variable were consistent; individual values were deemed consistent with the corresponding value produced by the reference execution engine if the result of rounding both values to the sixth decimal place produced equal numbers.[11]

For the given tests, the Simulation Subsystem's execution engine was deemed to be consistent with Vensim's and therefore to operate correctly.

5.2.2 Running Time and Memory Usage Tests

Two important questions that the Proof-of-Concept model created helped us answer were questions of implementation feasibility:

1. How quickly will the Simulation Subsystem be able to give back results and how does this vary with the number of simulation iterations?
2. What is the peak memory usage for the Simulation Subsystem and how does this vary with the number of simulation iterations?

[9]PySD Documentation,URL: https://pypi.python.org/pypi/pysd/0.1.0Last accessed: 07/07/2015

[10]Vensim Software Page, URL: http://vensim.com/vensim-software/ Last accessed: 07/07/2015

[11]The two execution engines apparently use different floating point accuracies in their computations and hence their results differ slightly. We do not consider a value 0.123456789 inconsistent with the value 0.123457, for instance. Given the fact that the output variables refer to real world objects (awareness: people, volume: documents), in fact, both those values could be rounded to the nearest integer value (0).

Due to the nature of the simulation, we expected memory usage to be low, irrespective of the number of simulation iterations, but were particularly concerned with execution speed for the model. The test was conducted for 30, 300 and 3000 simulation iterations (time steps).[12]

The results (Table 7) were very positive and re-affirmed the validity of our implementation choices. They suggest that neither computation time nor memory usage issues will be a driving factor in our future designs for changing the time granularity and/or the prediction horizon of our model.[13]

Table 7: Simulation Subsystem: Running time and peak memory usage test results

Iterations Steps	**Simulation Running Time[14]**	**Peak Memory Usage[15]**
30	1.10 sec	40.7MB
300	1.13 sec	41.9MB
3000	1.50 sec	42.2MB

5.3 External Experts' Feedback

Revisions on the model have taken into consideration feedback provided by the EU-Community WP4 experts. In addition to the testing activities that have been described in the previous sections, targeted input was asked from WP4 associated experts, through three concrete steps presented in Figure 20. During the first step, online discussions have been organised in order to introduce to the experts the Policy Component and the work conducted within WP4. Then feedback was gathered through the completion of a questionnaire targeted to specific aspects of the component, by a team consisting of ten experts, with different background such

[12] The number of iteration steps multiplied by the step period equals the prediction horizon. Thirty iteration steps may correspond to a month (when the step period is one day), half a minute (when the step period is 1 second) or seven months (when the step period is a week). Whereas decisions about the step period, the iteration steps and the prediction horizon are an important first step when designing and adjusting the model, for the purposes of the performance test, only the number of iteration steps is significant. By testing for 30, 300, and 3,000 steps we obtained data indicating how the performance of the Simulation Subsystem correlates to the length of the simulation.

[13] For instance, if we had decided that the desired prediction horizon would be 6 months, two options would be (a) 26 iteration steps with a one week time step and (b) 182 iteration steps with a one day time step. If performance degraded rapidly when moving from the 26 to 180 iteration steps, we might have had to base our decision on that. Our test results show us this would not have been the case.

[14] The reported simulation time is the average of the minimum three of ten measurements.

[15] The reported simulation time is the average of the median six of ten measurements.

as policy makers, researchers, business analysts and domain experts. In the final step, a subset of experts identified as more appropriate according to their background have reviewed and provided input in the WP4 deliverables.

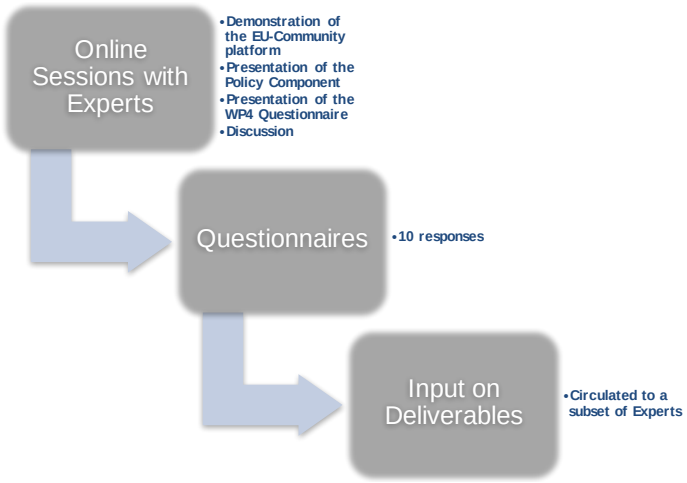

Figure 20: Methodology on Experts' Evaluation Feedback

Experts were asked to respond to questions regarding the critical metrics and factors on the evolvement of a policy debateand the relevant metrics that they would like to be predicted according to how significant they consider them.

5.3.1 Perceived Importance on the Simulation Subsystem metrics

The metrics that are monitored in the first version of the Simulation Engine were listed to be rated by the external experts. The purpose of this was to refine the critical metrics that users want to be monitored and predicted as they represent the stocks of the System Dynamics model, following the principles of Group Model Building.

Awareness and Engagement are considered as the most critical ones, corresponding to the people that are aware of a specific policy process and more over contribute in the process by adding input in terms of documents attached in the process. The number of related documents is considered less critical, whereas what seems to play more importance for the evolvement of the process according to the experts is the polarity of the documents (whether is positive or negative) and whether there is homogeneity among the documents that emerge. The same applies for the type of the author of the documents based on the classification adopted by the project. Finally, even more important are considered the comments correlated with documents than the documents themselves. The detailed results are presented in the following table.

Table 8: External Expert's Feedback on Metrics Importance

To what extent do you consider the following metrics critical within the evolvement of the policy debate?	**Very Uncritical**	**Uncritical**	**Neutral**	**Critical**	**Very Critical**
Awareness: The number of people aware of the policy process	0	0%	10%	60%	30%
Engagement: The number of people whohave contributed in the policy process	10%	0	10%	50%	30%
Volume: The total number of documents associated with the policy process	0	10%	50%	30%	10%
Documents per author type [Institution/Media/Stakeholders]	0	20%	20%	60%	0
Sentiment: The number of documents per sentiment (positive/negative)	0	0	40%	50%	10%
Controversy: The homogeneity/polarity between positive and negative documents	0	0	40%	30%	30%
Comments: The number of comments submitted to the documents associated with the policy process	0	20%	30%	30%	20%

In the question to suggest any other metrics to be predicted, experts proposed to analyse further metrics that are correlated with the awareness, such as referrals, shares, recommendations, media reactions, opinion surveys, media debates, article reviews and political speeches. Furthermore, they suggested to classify the people and organisations involved in the process according to their demographic characteristics, political persuasion, legitimacy and their types respectively. Apart from documents, the polarity of comments are considered crucial. Finally, they mentioned exogenous factors to the specific policy process that we may not be able to use/predict accurately, such as the volume of other ongoing policy processes preventing or allowing for the right focus from all stakeholders to book progress,

alignment and/or urgency of the topic covered by the policy process to current news/focus in media, etc.

5.4 Perceived Importance on the Critical Factors of the Simulation Subsystem

A question on the critical factors affecting the policy process has been also asked to the experts. The question to determine the variables of the simulation models, constituting the respective flows between stocks. According to the responses presented in the following table, weighting of the various auxiliary variables has been reconsidered in the second prototype of the Simulation Subsystem.

Table 9: External Expert's Feedback on Critical Factors

To what extent do you consider the following factors critical to affect the evolvement of a policy process?	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree
The publication of institutional documents (e.g. by EU or local governments)	0	0%	10%	70%	20%
The publication of stakeholders' documents (e.g. by civil society organisation)	0	10%	0	60%	30%
The publication of media articles	0	10%	0	80%	10%
The comments expressed on Social Media (e.g. tweets)	0	10%	20%	50%	20%
The reproductions of documents, articles or Social Media posts (e.g. retweets)	0	10%	40%	40%	10%
The reputation of persons involved in the policy process	0	10%	10%	50%	30%
The organisation of relevant physical events	0	0	30%	60%	10%

To what extent do you consider the following factors critical to affect the evolvement of a policy process?	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree
The existence of an ongoing relevant EU legislative procedure	0	10%	30%	50%	10%

It's notable that the most reinforcing factors are the three types of documents distinguished by the project methodology, rated as of equal importance. According to the responses, this also varies based on the reputation of people involved and can be also affected from the organization of relevant physical events. As the less critical factors are considered by experts, the existence of an ongoing official legislative procedures and the reproductions of the various documents attached to the policy process. Based on these findings, the weightings of the variables have been refined in the simulation model.

Other critical factors suggested by the experts, include the participation of relevant policy makers and citizens' organization, political changes such as election results at national or European level. In addition, the publicity of the debate in terms of media buzz or public events depending also on the countries involved.

5.5 Perceived Usefulness of the Simulation Subsystem

A set of questions was included to evaluate the intention to use to Simulation Subsystem based on the perceived usefulness by experts (Table 10). It is worth to note, that whereas they are reluctant to trust the predictions, they find the results of simulation experimentations useful.

Table 10: External Expert's Feedback on Usefulness of Simulation Subsystem

To what extent do you agree with the following	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree
I find the predictions provided by the Simulation Subsystem useful	0	0%	50%	40%	10%
I would use the Simulation Subsystem to experiment with alternative scenarios and get predictions	0	10%	20%	50%	20%

To what extent do you agree with the following	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree
I intend to trust the predictions provided by the Simulation Subsystem	0	20%	60%	20%	0

6 Conclusions and Next Steps

6.1 Ontology Builder

The Ontology Builder functions according to its specification and meets the performance goals set for it, not only in its internal operations, but also in its user interaction design, which enables particularly fast data entry. To put the various quantitative data collected in perspective, comparisons were made with WebProtégé, one of the best known and most used web-based ontology editors. WebProtégé is a well designed and implemented tool, yet both in terms of loading time and memory use, the Ontology Builder proved to be significantly faster and leaner. Moreover it was conclusively proven that it allows users to perform significantly faster data entry for the kind of data it was designed to handle.

There are thoughts for individual exploitation of the Ontology Builder, albeit not in a commercial context (research projects and later free web-based service). We consider the prototype to show potential, but believe its feature set would need to be expanded in order to be able to serve the needs of the community of individuals building multilingual SKOS vocabularies better than tools like WebProtégé.

6.2 Hybrid Predictions Subsystem& Statistical Predictor

The Statistical Predictor has been thoroughly evaluated and been significantly improved to the extent that outcome prediction accuracy is now slightly above 90%. For duration prediction even with a +/- 3 month tolerance, prediction accuracy was not significantly higher than 55%. At this point we would feel confident including outcome predictions produced by the Statistical Predictor in a commercial service, but would hesitate doing the same with duration predictions. There can be further research to improve both outcome and, more importantly, duration predictions, however, our current belief is that it will be unlikely that either will improve significantly: the 90% accuracy for outcome prediction is already very high and the 55% accuracy for duration predictions with a +/- 3 month tolerance indicates the duration prediction is a difficult task for statistical methods.

The blending of human expertise with statistical methods may help us overcome the limits of statistical methods. The Hybrid Prediction Subsystem paves the way towards a promising research direction exactly because it does not rely exclusively on past data; instead it brings into the picture the knowledge that human experts may have about the particular legislative procedure they are making a prediction about. While a non-expert is not very likely to fair consistently better than the Statistical Predictor, policy experts actively involved with the legislative procedures they make predictions about (and those discussing them with those who are actively involved) will be able to make much more accurate predictions. The challenge then is to ensure the HPS begins to learn who these experts are so that their opinion begins to weight more than other experts'.

We envisage work on the HPS continuing in a further research project.

6.3 Simulation Subsystem

The Simulation Subsystem is a research prototype that achieves its primary design and implementation goal, allowing for easy manipulation of the System Dynamics

model and instant deployment of improved versions. These are important properties when experimenting with models and with real data. The next steps for the Simulation Subsystem (and any machine learning-based or other alternatives) will clearly be in the context of a research project. However, it would have to be a research project working on data from a commercially exploited PolicyLine (or some other tool with a similar features and design aims). The evaluation of the Statistical Predictor was based on 1,500 legislative procedures. What we hope to see is a dataset of Policy Processes of comparable size and quality that can be used for both manually fine-tuning and automatically training prediction models, as well as for various data analyses that will give insights into how Policy Processes evolve.

The EU Community project has set the foundations for such future developments, but further steps will be necessary.

APPENDICES

APPENDIX A: Outcome Predictor Variations

Variation O1: Adding a Standard Default Prediction

Rationale
The original Statistical Predictor's design is an expression of the history-based prediction methodology in its purest form, as the predictor does not make any prediction if there is no support for it from the available data.[16] If the Statistical Predictor were to make a default prediction, any one of the three possible outcomes, instead of failing to make a prediction, there is a chance that this prediction would be the correct prediction. If the three possible outcome predictions are not equally likely to be the correct outcome, choosing the default to be the most likely outcome, not in general, but specifically in cases where the original Statistical Predictor fails to make a prediction, this will result in the best accuracy this idea can produce.[17]
Description of Variation
In cases where the original Statistical Predictor cannot make a prediction, predict "Pass with amendments"; otherwise predict what the original Statistical Predictor predicts.
Accuracy
The difference of Variation O1 in comparison with the original outcome prediction algorithm is that in any case where the original Statistical Predictor would fail to make a prediction, "Pass with Amendments will be returned as the prediction". This increases accuracy to 90.54%, a very significant improvement over the already very respectable 77.89% accuracy of the original Statistical Predictor. Had we preferred Pass without Amendments as the default prediction, accuracy would have risen much less, to 79.64%, whereas if he had chosen Fail as the

[16] In other words, outcome prediction is defined as a *partial* function from (incomplete) legislative procedures to legislative procedure outcomes. The question is what happens when for a certain legislative procedure, the predictor provides no prediction; one option is to ignore this case (count it as neither a success nor a failure), as no prediction was made, and the alternative is to count this as a case where a prediction was requested and the predictor failed to provide the correct prediction, or indeed, any prediction. We have taken the latter approach, as for a user, a failure to provide any prediction is a failure to provide a correct prediction.

[17] If two of the three scores are equally likely, and more likely than the third, choosing either will have the same effect.

default prediction accuracy, accuracy would have risen even less, to only 78.91%.[18]

Variation O2: Fixed Tie-Breaking Preferences

Rationale
Whereas Variation O1 is a significant improvement over the original Statistical Predictor's outcome prediction algorithm, it can be improved further. One shortcoming of Variation O1 is that it produces the same default prediction, Pass with Amendments, irrespective of the situation that prevents the original Statistical Predictor from making a prediction. One particular group of cases where this behaviour is clearly wrong is when the original Statistical Predictor has data that indicate the best possible prediction is either Pass without Amendments or Fail, not Pass without Amendments, but cannot decide which of the two predictions to make. In those cases, Variation O1 would still predict the outcome to be Pass without Amendments, in spite of the evidence in the data.
Description of Variation
Variation O2 is based on the idea of tie-breaking preferences: it prefers the Pass with Amendments outcome, to the other two and the Pass without Amendments outcome to the Fail outcome. It uses those tie-breaking preferences to produce an outcome prediction in cases where the original Statistical Predictor cannot decide between the two best or three possible outcomes with equal support from the training data.[19]
Accuracy
The accuracy of Variation O2 is 90.84%, which constitutes a very small but welcome improvement over the accuracy of Variation O1.

[18]**"**Pass with Amendments" as a default is far better than the other two alternatives as:

- The original Statistical Predictor failed to produce an outcome prediction for 1,246 test cases.
- For the majority of those cases, 1,177 to be exact, it failed to produce a prediction because there were no relevant training data.[18]
- Lack of relevant training data is common for legislative procedures with a complicated history and those are far more likely to pass than fail, but not without amendments.

[19]As a result, in the above examined case where an equally strong case can be made for either Pass without Amendments or Fail (but not Pass with Amendments), being the best outcome, Variation O2 will produce a Pass without Amendments prediction, unlike Variation O1 which would produce a Pass with Amendments prediction.

Variation O3: Per Case Defaults

Rationale
The idea of trying to determine a good default prediction in cases where the original Statistical Predictor fails to provide a prediction is obviously in the right direction. However, there is information that can be taken advantage of to define a number of cases for which a different default prediction might be a better choice than the default choice of Variation O1 (see below).
Description of Variation
Variation O3 constitutes an alternative attempt to improve on Variation O1. Instead of always considering "Pass with amendments" to always be the best prediction in cases the original Statistical Predictor cannot decide which prediction to make, Variation O3 uses a set of rules to determine the default prediction it considers to be best. For legislative procedures of type OLP and SLP (Ordinary and Special Legislative Procedures whereby the Parliament is consulted), Variation O3 considers "Pass with Amendments" to be the best default prediction; for all other legislative procedure types it considers "Pass without Amendments" as the best option[20], except if a step strongly associated with one of the other outcomes is part of the history: ▪ If there is a step whereby an institution accepts a proposal *without* amendments and the preferred outcome on the basis of the legislative procedure type was "Pass *with* amendments", the prediction will instead be "Pass *without* amendments"; this change of prediction will be correct most, but not all, of the times, as a further step may involve amendments. ▪ If there is a step whereby an institution accepts a proposal *with* amendments and the preferred outcome was "Pass *without* amendments", it would change to "Pass *with* amendments"; this change will never be incorrect as an amendment means that only "Fail" and "Pass with amendments" are possible outcomes[21]. ▪ If there is a step whereby an institution rejects a proposal, in which case the prediction changes to "Fail".

[20]It is worth noting that for Special Legislative Procedures requiring the Parliament's consent (APP), while the current data indicates that "Pass without amendments" is the best default prediction, "Fail" comes as a close second best option; indeed more legislative procedures of that type have "Fail" as their outcome. Therefore, this is a choice that might need to change when more training data are considered.

[21]Of course, the fact that a "Pass without amendments" prediction would be guaranteed to be wrong, does not mean that the "Pass with amendments" prediction itself may be wrong as the legislative procedure may result in a "Fail" outcome in the end

Accuracy
Variation O3, the third variation of the outcome prediction algorithm, has a prediction accuracy of 91.28%, which is better than the accuracy of the previous two variations.

Variation O4: Per Case Defaults for Weak Predictions

Rationale
None of the previous variations ever disagree with a prediction the original Statistical Predictor makes; they are only concerned with cases where it fails to provide a prediction, which happens when none of the predictions has more support than the others. Some predictions made by the original Statistical Predictor have much stronger support from training data than the alternatives; others not. In Variation O3 if there are 30 relative procedures supporting outcome A, 30 supporting B and 5 supporting C, the per case defaults are used, but if there happened to be even a single additional relevant legislative procedure in the training set that had outcome A, the mechanism of the original Statistical Predictor would be used instead, meaning that Variation O3 would have A as its prediction. The question is whether the per case default prediction would have been a better bet than a weak prediction. A weak prediction is a prediction that does not seem sufficiently more supported by the relevant available training data than the alternatives. The strength of a prediction is defined as the number of relevant legislative procedures in the training data supporting it, minus the maximum of the number of legislative procedures supporting each of the remaining possible predictions. A prediction of strength 0 is considered a weak prediction. The question is if a prediction of strength 1, 2 and so on and so forth should also be considered a weak prediction. The idea is to explore the possibility that default per case predictions are better than predictions of strength 0, 1, 2, … The strength threshold will be the strength a prediction will need to have in order not to be dropped in favour of a per case default prediction. Experimentation revealed that the optimum strength threshold for our data was 2, meaning that predictions of strength 0 and 1 are not as accurate as the per case default predictions.
Description of Variation
Variation O4 decides on what the default prediction should be on the exact same basis as Variation O3. It differs though in that it does not restrict the use of default per case predictions only in cases where the original Statistical Predictor would have made no prediction whatsoever. Unlike all previous outcome prediction variations, Variation O4 may prefer the default prediction over an actual prediction by the original outcome prediction algorithm of the original Statistical Predictor. It will do so when it considers the prediction to be weak i.e. when it has a strength less than the strength threshold, which for the available

data was determined to be 2.
Accuracy
For the purposes of Variation O4, with a strength threshold of 2 was 91.50%.[22]

[22]When only predictions of strength 0 are considered weak, Variation O4 is behaves identically to Variation O3 and its prediction accuracy is 91.28%. When predictions of strength 0 or 1 are considered weak, Variation 4 has a prediction accuracy of 91.50%. If the threshold is increased further, accuracy falls and Variation 4 becomes a worse option than Variation O3.

APPENDIX B: Duration Predictor Variations

Variation D1: Taking Tolerance into Consideration

Rationale
One problem of the duration prediction algorithm of the original Statistical Predictor is that it aims to find the best possible prediction irrespective of the threshold. If there are 5 relative legislative procedures that have a duration of 10 months, 4 that have a duration of 15 months, 3 that have a duration of 16 months and 2 that have a duration of 19 months, the original Statistical Predictor would have considered 10 months to be the best duration prediction, whereas Variation D1 would instead give a prediction of 16 months. Having prediction tolerance play a part in the evaluation of duration predictions helps judge predictions more fairly; for instance, in the above example, it is reasonable to ask that a prediction of 16 months be evaluated to be better than a prediction of 10 months. However, the original Statistical Predictor would make the latter choice. Variation D1 and all subsequent variations of the original duration prediction algorithm take the tolerance level into consideration in order to produce better predictions.
Description of Variation
For Variation D1, taking the threshold into consideration is the only point of differentiation with the original duration prediction algorithm of Section 3.4.1. The original algorithm, given a legislative procedure X, counts the number of relevant legislative processes in the training set (i.e. legislative procedures in R', see Section 3.4.1) for each duration witnessed in the set and if one of these stands out i.e. has a higher count, it considers it to be the best prediction it can make; Variation D1 does exactly the same except that it counts, for each candidate duration prediction, not only the relevant legislative procedures with that duration, but also legislative procedures with a duration within the given tolerance level. Should the tolerance level be 0, Variation D1 is identical to the original algorithm.
Accuracy
Given a +/- 3 months tolerance, Variation D1 achieves an accuracy of 50.40% which is slightly higher than the 47.29% accuracy of the original Statistical Predictor.

Variation D2: Always Making a Prediction

Rationale
Both the original duration prediction algorithm and variation D1 do not always

make a prediction. They do not make a prediction when they do not find a duration prediction that stands out as better than alternatives.
Description of Variation
There are two remedies that variation D2 tries (in the order they are listed) in order to produce a prediction in such cases: 1. If there are more than one equally good candidate predictions (with strength > 0), it chooses the first (i.e. the smallest duration). 2. Otherwise, it returns a duration prediction that is equal to the legislative procedure's current duration plus the tolerance level (e.g. 3 months).
Accuracy
Given a +/- 3 months tolerance, the accuracy of Variation D2 is 51.89%, a small improvement over Variation D1's accuracy.

Variation D3: Not Earlier than the Optimum Earliest Prediction

Rationale
The default prediction of D2 (current duration plus the tolerance level) was chosen to maximise accuracy on the basis of an observation that the lowest durations correspond to the most legislative procedures (which indicates that a prediction as close as possible to the current duration would be likely to be a good prediction in the absence of additional data) but also of the tolerance level which means that if the prediction is pushed into the future by the amount of months in the tolerance level (e.g. 3), a wider range of possible durations will be covered, including the possibility the legislative procedure end imminently. In other words, the default prediction of Variation D2 is the optimum earliest possible prediction. Therefore, it makes sense that no prediction earlier than that is made; this is something that already applies to the default predictions of D2, but not to the remaining predictions. D3 remedies this.
Description of Variation
Variation D3 adds a final step according to which the prediction made is the minimum of the prediction computed by Variation D2 and the optimum earliest possible prediction.
Accuracy
Given a +/- 3 months tolerance, the accuracy of Variation D3 is 52.26%, a very small improvement over the accuracy of Variation D2.

Variation D4: Backtracking

Rationale
Suppose that in a previous step, the duration of a legislative procedure was predicted to be, say, 6 months, but that after one more step is taken, Variation D1 fails to find a duration prediction that stands out as better than alternatives. Variations D2 and D3 have two remedies for this problem. Variation D4 adds one more remedy: trying to obtain a duration prediction by leaving out the most recent step of the legislative procedure. The reasoning for this additional remedy is that by going back one step one can obtain a prediction more relevant to the current legislative procedure than by trying the per case default predictions. This process of going back in history may be repeated until a prediction of strength > 0 can be found.
Description of Variation
Variation D4 attempts to obtain a prediction in exactly the same manner as Variation D1. If that fails to produce a prediction, there are three remedies that variation D4 tries (in the order they are listed): 1. If there are more than one equally good candidate predictions (with strength > 0), it chooses the first. 2. Otherwise, it tries to find a duration prediction for the history the legislative process had in its prior step (but taking into consideration only legislative procedures with a duration equal or greater than its current duration). If successful, it returns the best (or the first of the best) candidate duration predictions. Otherwise it tries moving further back in the legislative process's history. 3. If that fails also, it returns a duration prediction that is equal to its current duration plus the tolerance level (3 months). Finally, taking cue from Variation D3, the prediction it returns is the minimum of the prediction computed in the above steps and the optimum earliest prediction (duration of the legislative process so far plus the tolerance level).
Accuracy
Given a +/- 3 months tolerance, the accuracy of Variation D4 is 52.90%, which is slightly higher than the accuracy of Variation D3 (52.26%).

Variation D5: Far Back Jumps

Rationale
The second remedial step, in case of no prediction, of Variation D4 looks more promising on paper than it turned out to be. An experiment which a variant of D4's algorithm backtracked to step 0 (when its history consisted only of its type),

instead of the previous step of a legislative procedure, resulted in a better prediction accuracy (53.25%). Further experiments demonstrated there was merit to the idea of jumping back to the first steps of legislative procedures (where the number of relevant legislative procedures is higher).

Description of Variation

Variation D5 closely resembles Variation D4; its difference is that it jumps back not necessarily to previous step in order to find appropriate relevant legislative procedures, but the minimum of that and a step S. Suppose that S=4 (the optimal value according to our experiments) and that the current step count for the legislative procedure in question is 7; whereas variation D4's second remedial step would be to jump back to step 6 and only if it does not find a prediction with strength greater than 0, to step 5 or below, variation D5 would jump back to step 4 and only if it does not find a prediction with strength greater than 0, to step 3 or below.

Accuracy

Given a +/- 3 months tolerance, the accuracy of Variation D5 with S=4 is 56.26%, which is significantly higher than the accuracy of Variation D4 (52.90%); accuracy results for different values of S can be found below:

S	Accuracy
0	53.25%
1	54.15%
2	55.87%
3	56.00%
4	56.26%
5	56.11%
6	55.30%
7	54.13%
8	53.41%
9	53.09%
10	52.97%
11 – 25	52.90%