Chapter 15

Exploration of the Rhapsodie corpus: Data structure, formats and query tools

A. Lacheret-Dujour, S. Kahane, R. Bawden, S. Fleury, I. Wang

Abstract

This chapter describes the data structure of the Rhapsodie Treebank and discusses methodological issues stemming from the complexity of this structure, articulated around three independent, non-aligned, hierarchies: microsyntactic, macrosyntactic, and prosodic, as well as the challenging questions to be resolved in this context. It discusses the specific problems posed by the simultaneous processing of the phonological stream (prosodic level) and the orthographic stream (syntactic level), which are often far from being isomorphic in French, and the related problem of the processing of disfluent and/or overlapping strings, which do not have the same representation in the syntactic and the prosodic hierarchy. Then, it present the formats adopted to encode prosodic and syntactic annotations and query them simultaneously, given that the prosodic architecture is a non-recursive time-aligned representation while the syntactic one is a recursive tree-based representation.

1. Introduction

The linguistic analyses presented in the last part of this monograph have three main objectives. First, drawing up an inventory of the syntactic and prosodic constructions that can be found in the Rhapsodie corpus by pointing out expected and unexpected patterns, and checking their usage and their frequency. Second, enhancing our knowledge of intonosyntactic articulation, with particular regard to segmentation of the speech flow and the alignment of syntactic and prosodic units' boundaries. Finally, observing the distribution of prosodic, syntactic and intonosyntactic constructions across discourse genres.

These different issues raised two methodological challenges: (i) the creation of data formats and databases in which information about every level of annotation is encoded; (ii) the development and the adaptation of query tools capable of exploring the Rhapsodie data flexibly and quickly, even for intonosyntactic structures, that is, structures combining macro and microsyntactic features on the one hand, prosodic and syntactic features on the other hand. In other words, the issue of formats and query tools that simultaneously combine symbolic information (orthographic text, tokenisation, features on tokens, links between tokens, segmentation into units) and temporal information (time alignment of units), is a crucial one, which was for the most part unexplored until the launch of the Rhapsodie project. We present the formal structure used to encode Rhapsodie data and the difficulties associated with the intonosyntactic querying (Section 2). Then, we describe the various encoding formats and query tools developed and/or adapted for querying the Rhapsodie database (Section 3).

2. The complex data structure in Rhapsodie

In spite of the abundance of studies on the annotation and querying of spoken data, as far as we know, none of the previous attempts to process ordinary speech has taken into account overlaps (i.e. at least two people speaking at the same time) and disfluencies (filled-pauses, repetitions, self-repairs, false starts and truncations of segments), or enabled phonetic and syntactic annotations to be simultaneously queried. Consequently, the methodological challenges for Rhapsodie can be summarized in three issues:

1) How to annotate, encode, query and analyze disfluent and overlapping segments;

2) How to formally encode prosodic and syntactic annotations;

3) How to query on units of a different nature – a phonetic time-aligned sound file for prosodic units and an ordered list of lexemes for syntactic units.[1]

In other words, complex formal relations exist between prosodic and syntactic units and multiple problems arise in the complex process of partial alignment. For a start, we have three non-congruent hierarchies to study, presented in Part 2 and summarized in Section 2.1: prosodic, microsyntactic, and macrosyntactic hierarchies. However, the prosodic hierarchy is not explicit in our tier-based annotation (Section 2.2). Overlapping between speakers is the source of various problems, for prosody itself (Section 2.3), as well as for syntax and syntax-prosody alignment (Section 2.4). Even the basic component of syntax – lexemes – are not congruent with the basic components of prosody – syllables and metrical feet (Section 2.5).

2.1 Three independent hierarchies

Prosodic annotations were conducted following a tier-based approach (Praat software, Boersma & Weenink, 2010). Perceptual syllabic saliences were manually annotated using a three-scale labelling distinguishing between strong, weak, or zero prominences. The ± disfluent nature of a syllable was also labelled (Chapter 9). Building on these two pieces of information and the location of major prosodic breaks, we were able to automatically generate the totality of the prosodic structure made up of a hierarchy of prosodic segments (speech turns, intonational periods, intonational packages, rhythmical groups, and metrical feet) characterized by more or less prominent boundaries (Chapters 10 and 11). Finally,

[1] We use the term *lexeme* for the word segmentation done by Rhapsodie's syntactic team to avoid any confusion with a possible word segmentation done on phonological/prosodic grounds.

prototypical-stylized melodic contours were generated automatically from the stylized F0 curve for all prosodic and syntactic units (Chapter 13). The availability in the Rhapsodie Treebank of these various contours allows the user to build various lexicons of intonational contours in an extremely flexible way according to his or her research goals.

The tier-based approach used in the prosodic labelling and based on time-aligned annotation can simulate constituent structures by time-aligning larger segments in one tier with smaller segments in another tier (Figure 1). Nevertheless, this does not allow for an explicit encoding of constituent structure, because one segment cannot be linked to another segment (we only know that the boundaries of one unit are inside the boundaries of the other, but the fact that the former is an immediate constituent of the latter is not explicit). Therefore, neither constituency-based nor dependency-based syntactic structures can be formally encoded in the commonly used tools for prosodic annotation.

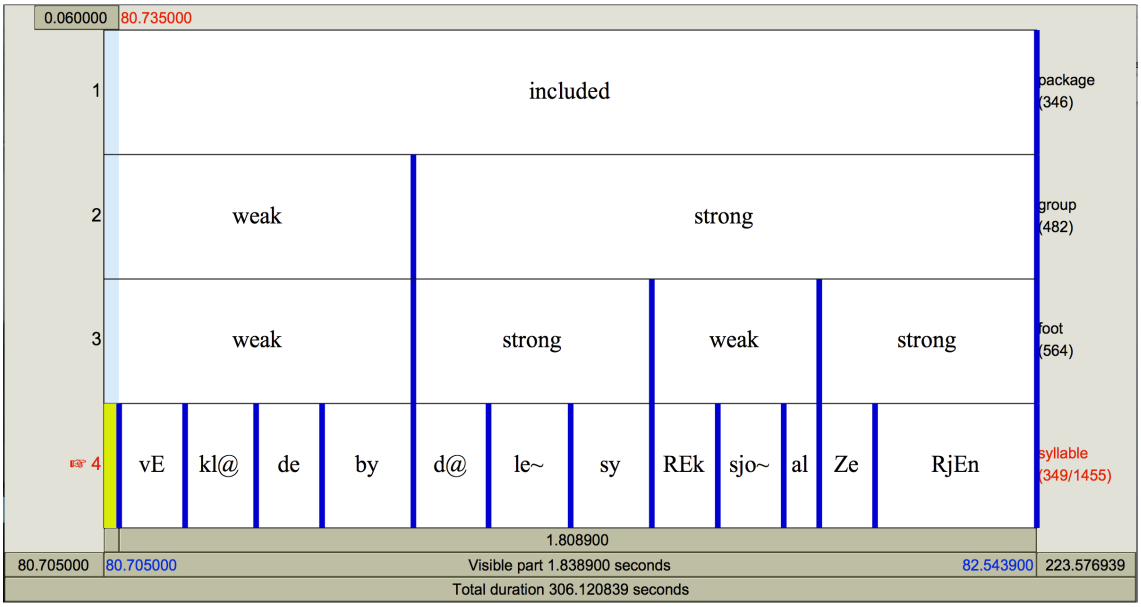


Figure 1. Representation of prosodic constituent structure by time-aligning larger segments in one tier with smaller segments in the tier below, [Rhap-D2005, Lacheret]

As explained in Chapter 3, microsyntactic and macrosyntactic phenomena were encoded independently from one another in a modular approach. Microsyntactic phenomena were encoded in a partially computer-aided approach relying on collaborative online tools, while macrosyntactic segmentation was done entirely manually by marking up the transcription. The microsyntactic level describes the kind of syntactic relations that are usually encoded through dependency trees or phrase structure trees. These relations are annotated in all the major syntactic treebanks, such as the Penn Treebank, the Prague Treebank, the French Treebank, the Copenhagen Dependency Treebank, etc. (Chapter 4). The main originality of our annotation is the fact that the whole text has been parsed, including disfluencies, which are encoded as list phenomena, like reformulation and coordination, that is to say the multiple realization of one and the same governed position (Chapter 5). Macrosyntax can be regarded as an intermediate level between syntax and discourse. This level describes and classifies the sequences that make up one and only one illocutionary act as well as the relations holding between their components (Chapter 6). As a result, illocutionary units [IUs], and even speech turns, are not considered as boundaries of microsyntactic dependencies, which means that microsyntax cannot be considered as a refinement of macrosyntax, and the two levels of annotation define two non-congruent hierarchies. For instance in dialog (1), the interviewer L1 questions L2 by adding to L2's IU a prepositional phrase which piles on L2's prepositional phrases and L2 answers by repeating the prepositional phrase for confirmation. This results in a microsyntactic tree that stretches over three speech turns and as many IUs.

(1) [Rhap-D2001, Mertens]

L2*: je crois que je ne me suis pas conduit d'une façon conforme à ce qu'on attend euh d'une jeune fille d'abord et d'une femme ensuite*
L1 : *d'une jeune $- bourgeoise*
L2 : *dans la -$ disons d'une jeune bourgeoise voilà*

L2: 'I think my behaviour wasn't what was expected um from a girl first and from a lady next
L1: from a young $- bourgeoise
L2: in the -$ let's say from a young bourgeoise that's it'

As shown in Figure 2, the three speech turns of (1) form only one microsyntactic unit where the two prepositional phrases *d'une jeune bourgeoise* 'from a young bourgeoise' pile up on the previous coordinated phrase *d'une jeune fille d'abord et d'une femme ensuite* 'from a girl first and from a lady next' and are linked by paradigmatic relations of negotiation (*para_negot*, specifically asking for confirmation and providing confirmation, cf. Chapter 5).[2]

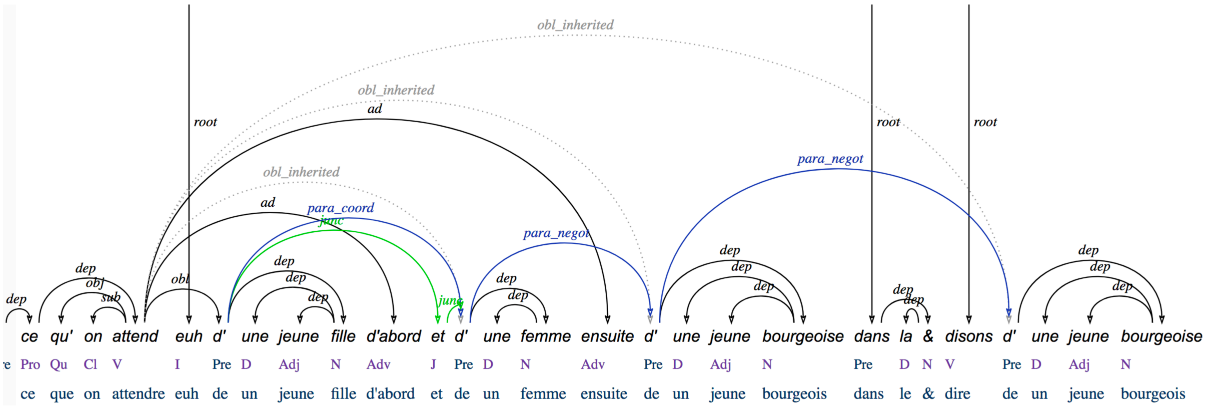

obl_inherited
obl_inherited
ad
root
root
root
para_negot
obl_inherited
ad
para_coord
junc
para_negot
dep
dep
obj
sub
obl
dep
dep
dep
junc
dep
dep
dep
dep
dep
dep
dep
dep
dep
dep
dep
ce qu' on attend euh d' une jeune fille d'abord et d' une femme ensuite d' une jeune bourgeoise dans la & disons d' une jeune bourgeoise
Pro Qu Cl V I Pre D Adj N Adv J Pre D N Adv Pre D Adj N Pre D N V Pre D Adj N
ce que on attendre euh de un jeune fille d'abord et de un femme ensuite de un jeune bourgeois dans le & dire de un jeune bourgeois

Figure 2. Microsyntactic structure of (1)

The fact that micro- and macrosyntactic structures are independent is not a problem in itself, but we must be aware that we have three different hierarchies on the same data, with the prosodic, microsyntactic and macrosyntactic structures. It is even possible to consider a fourth hierarchy, since list phenomena define their own hierarchy. Lists form a subpart of the microsyntactic structure, but as lists are in some sense orthogonal to the government relation (a paradigmatic pile is the multiple realization of the same governed position), list embeddings can be studied individually.

2.2 Overlaps in prosody and syntax

As regards prosody, monologues were directly segmented into intonational periods (henceforth IPE), but dialogs required a prior segmentation into speech turns according to the person who is speaking (Chapter 10). This segmentation into speech turns is not trivial due to

[2] The analysis of the adverbs *d'abord* 'first' and *ensuite* 'next' has not been resolved satisfactorily. We decided to analyze them as adjuncts on the verb, but they are also clearly part of the layers of the list, which is not captured by our dependency structure.

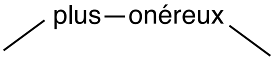
frequent overlaps between speakers, that is, parts where two or more persons are speaking together. In overlapped segments, we can process at best one and only one speaker: The segment produced by this speaker is handled just like any other non-overlapping part. Consequently, the timeline can be divided into segments attributed to exactly one speaker: the only speaker in the case of monological sequences, the speaker who dominates in the case of an overlap that can be processed, and the speaker who was speaking just before in the case of a break between two speech turns.[3] The problem of overlaps has in fact been eliminated from the data structure and each point of the time line is thus attributed to exactly one speaker. Hence, the time line can be partitioned into speech turns, which are divided into intonational periods and so on. The price paid for that is that some intonational periods are truncated.[4]

The syntactic annotation is a lexeme-based annotation. Linear precedence relations between syntactic units are based on linear precedence between lexemes. These linear precedence relations are symbolic and do not necessarily correspond to time-based precedence relations. In particular, lexeme order differs from the time-based order in overlaps, where the production of the speakers overlapping each other has to be ordered separately. For instance in (2) (where $- and -$ indicate the beginning and the end of the overlap), *onéreux* ’expensive’ has no order relation with *ouais* ‘yeah’, even though *onéreux* was produced after *ouais*.

(2) [Rhap-D0009, PFC]

L1: *là par contre ça doit être $- plus onéreux*
L2: *ouais il faut -$ faut compter autour de soixante soixante-dix*

L1: ‘there on the contrary it must be $- more expensive’
L2: ‘yeah you got -$ got to count around sixty seventy’



[3] We consider that a break between two speech turns must be attributed to the previous speaker due to the fact that it arises because the previous speaker stops and not because the next speaker has not yet started.
[4] It occurs even when there is no overlap and the speaker is interrupted by another speaker.

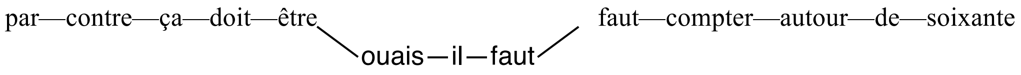


Figure 3. Partial order on the lexemes for (2)

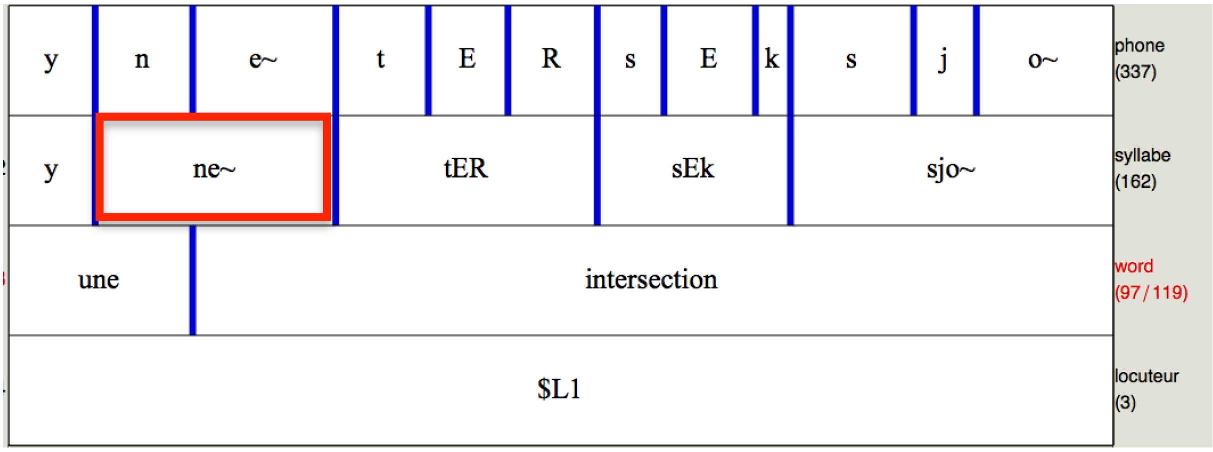
Moreover, the time-alignment is only partial in overlaps and cannot be used for compiling lexeme order. As explained in Section 2.1, for technical reasons linked to signal processing, time-alignment is performed for at most one of the speakers.

Strictly speaking the text of a dialog with overlaps is not linear. There is only a partial order on the lexemes (Figure 3). It is only a convention, based on the continuity of speaker productions and IUs, which led us to adopt a linear order for lexemes and to decide that L1's production precedes L2's production for the lexeme order.

The fact that a spoken corpus requires two orders for the annotation, a temporal order and a structural order, partially aligned, was anticipated and formalized by Bird and Liberman (2001). Their formalization was implemented for the AN.ANA.S corpus by Voghera and Cutugno (2009), but without addressing our central problem of the duality of time-aligned and non time-aligned units. Contrarily to Bird and Liberman, we preferred to introduce an order on lexemes rather than to introduce abstract points with only structural order relations with the relevant time points (i.e., the start and end points of the time-aligned units).

2.3. Non-alignment of syntactic and prosodic basic units

Lexemes are the basic components of syntax and syllables the basic components of prosody. These two tokenisations of the speech chain are not congruent, because one syllable can span over two words, such as /nɛ̃/ in ***une in****tersection* 'an intersection' (Figure 4). This first problem was formally solved by considering that a syllable is the child of the word bearing the vowel (*intersection* in our example).



<table>
<tr><td>y</td><td>n</td><td>e~</td><td>t</td><td>E</td><td>R</td><td>s</td><td>E</td><td>k</td><td>s</td><td>j</td><td>o~</td><td>phone
(337)</td></tr>
<tr><td>y</td><td colspan="2">ne~</td><td colspan="3">tER</td><td colspan="3">sEk</td><td colspan="3">sjo~</td><td>syllabe
(162)</td></tr>
<tr><td colspan="2">une</td><td colspan="10">intersection</td><td>word
(97/119)</td></tr>
<tr><td colspan="12">$L1</td><td>locuteur
(3)</td></tr>
</table>

Figure 4. A syllable overlapping two words [Rhap-D0017, Avanzi]

However, even the second level of the prosodic segmentation, defined by prominence annotation, is not congruent with the lexeme segmentation, because prominent syllables are not always final syllables of lexemes and a foot can span over half a word. As shown in Figure 5a, the second syllable of *Olkaloo* /ol.ka.lu/ is prominent and the lexeme is divided into two metrical feet. Conversely, in Figure 5b, the preposition *pendant* 'during' has no prominent syllable and is in the same metrical foot as *quatorze* 'fourteen'.

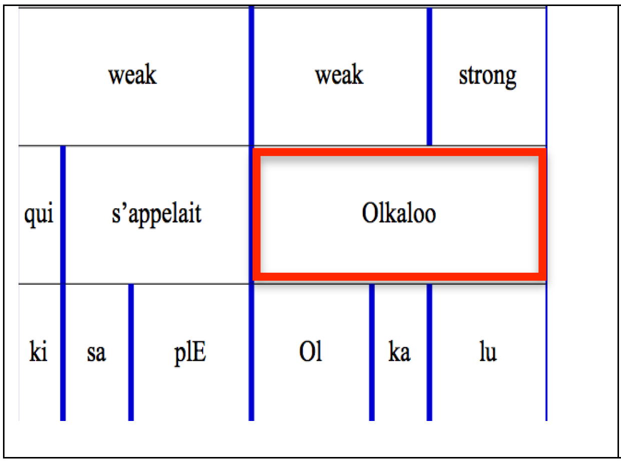


Figure 5a
A lexeme including two metrical feet
[Rhap-D2004, Lacheret]

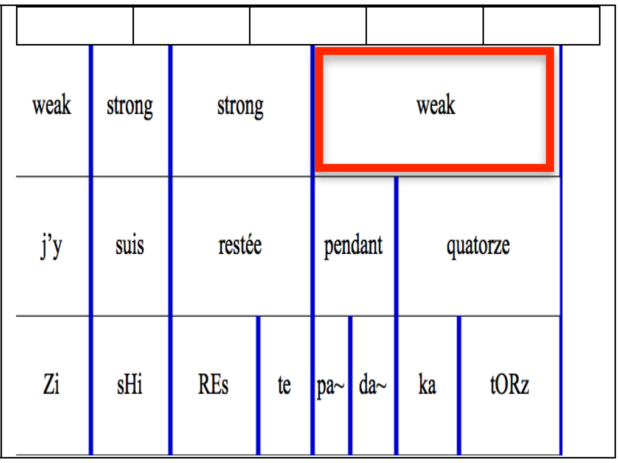


Figure 5b
A metrical foot spanning over two lexemes
[Rhap-D2004, Lacheret]

The fact that feet and words are segmentations of the same level implies that we must handle two constituent trees, one including feet, the other including words. All other nodes are shared between the two constituent trees. Thus, our structure can also be seen as a constituent DAG (a syllable can be a child of both a foot and a non congruent word) rather than as a set of constituent trees.[5]

3. Encoding formats and query tools

As recalled in the introduction of this chapter, the main goal in the Rhapsodie project is to explore the interface between prosody, syntax, and discourse, and particularly to understand and model the role of syntactic and prosodic features in the categorization of discourse types. Such an objective requires a tool that simultaneously combines queries on metadata, time-aligned units, and hierarchies of units, in order to access all the situational variables and all the prosodic information (all prosodic units even syllables and phonemes, temporal

[5] A DAG (directed acyclic graph) is a directed graph with no cycles of directed edges.

information such as durations of segments and pauses, etc.), and all the syntactic information including symbolic precedence relations. Since such a generic tool does not exist, two solutions were explored: first, developing specific tools; second, projecting the data in an existing format and using existing tools. The first strategy is still under development and will not be presented here (see Beliao 2016 for a specific tool, OOPS, exploiting an object-oriented encoding of the Rhapsodie database). The second strategy used for the following chapters is presented in Sections 3.1 (projection of the Rhapsodie data structure in a tabular format *à la* ConLL) and 3.2 (use of an existing tool initially dedicated to textometry and adapted to our needs).

3.1. Tabular format

A dependency treebank can be easily encoded in a table, such as the CoNLL format (Nilsson et al. 2007), which has now become a standard (Table 1). Each token is on a different line and receives an identifier (first column). Its governor is identified by the governor's identifier (fifth column). Other information can be encoded in other columns: here, for instance, the lemmatization (third column), the part of speech (fourth column) and the grammatical function (sixth column).[6]

Table 1. Dependency tree of Figure 1 in a tabular format

[6] In our case, the dependency structure is a graph: A word can have several governors (such as the first *d'* in Figure 2, which is *junc*, *para_coord*, and *obj_inherited*). But a word can never have more than three governors, which means that the whole graph can be encoded by trebling the Governor and Function columns.

ID	Word	Lemma	POS	Governor	Function
19	ce	ce	Pro	18	dep
20	qu’	que	CS	22	obj
21	on	on	Cl	22	sub
22	attend	attendre	V	19	dep
23	euh	euh	I	0	root

Other information attached to the word can be easily encoded, such as morphosyntactic features or start and end times. It is also possible to encode units in a tabular format. The most common encoding mode is the BILOU format (corresponding to Begin, In, Last, Out, Unique). We can for instance introduce a column to encode nuclei. The given value indicates if a word belongs to a nucleus (B at the beginning, I in the middle, L at the end and U when the word is a nucleus in its own right) and if it does not belong to a nucleus, it has the value O. A tabular format was used for the encoding of macrosyntactic and prosodic constituents (Table 2; see the description of our tabular files with their 63 columns in Wang & Bawden 2015).

Table 2. Excerpt from the tabular file, showing the annotation of nucleus, illocutionary unit and intonational period with the BILOU format ([Rhap-M2004, Rhapsodie])

ID	Word	Speaker	Nucleus	Illocutionary unit	Intonational period
1	mes	$L1	O	B	B
2	chers	$L1	O	I	I
3	compatriotes	$L1	O	I	L
4	je	$L1	B	I	I
5	voudrais	$L1	I	I	B
6	d’abord	$L1	I	I	L
7	exprimer	$L1	I	I	I
8	ma	$L1	I	I	I
9	sympathie	$L1	I	I	L

A tabular format is very easy to handle, but has two main limitations. First, it rests on a particular tokenization. In our case the level of tokenization, the word/lexeme, was imposed by syntax. As a result, we cannot access the prosodic properties of smaller units such as syllables and phonemes with such a tool. Second, all the information is attached to tokens and, even if units can be delimited, it is not convenient to associate features to units (syntactic category, tonal shape, etc.) and it is impossible to encode the various relations between units: linear relations (inclusion, adjacency, etc.) as well as domination relations (immediate constituency in the prosodic and macrosyntactic trees). Some tools dedicated to searching treebanks, such as ANNIS (Krause and Zeldes 2016), allow that, but they do not take temporal information into account. Nor is it possible to adopt a format entirely based on temporal information, such as Praat, because the temporal information (time intervals and boundaries) is only partial in the tabular format exploited by Trameur. For instance, the tabular format can give information about the beginning and the end of a unit, but its duration is not easy to query. Therefore, in addition to the Trameur, the temporal features associated to the prosodic units were computed with a Praat script to run part of the prosodic statistics presented in Chapter 17.[7] Our tabular format was used to train a macrosyntactic segmenter (Wang et al. 2014) and to study the corpus with the Trameur.

3.2. Trameur: a statistical query tool

Trameur (Fleury 2015) (from Fr. *trame* 'grid, raster, framework') is a statistical tool that can be used to analyze the behavior of a given linguistic unit or feature in a corpus. It can be applied to any tokenized text with labels on tokens and was applied to Rhapsodie's tabular format. It was initially developed to study word distribution in textometry and discourse analysis (Née et al. 2012).

Trameur allows us to cut the corpus into sections. The Rhapsodie corpus is basically cut into samples. It is thus possible to compare various samples and to study the specificity of a particular piece of information (for instance labels in a given column of the tabular file: see

[7] http://www.haskins.yale.edu/staff/gafos_downloads/AcouToyPraat(1).pdf.

above Table 2) for each sample. But most importantly, it is also possible to cut the corpus into sets of samples rather than individual samples, by following the metadata attributed to the samples. In other words, Trameur allows us to combine information and to study a particular feature or combination of features as a function of a particular genre.[8]

Trameur evaluates the specificity of a given form in a section according to the distribution of this form in the whole corpus and its probability of occurrence in the section (see Lafon 1980 and Lebart & Salem 1994 for the computation of specificity indexes).[9] Non-specific forms, that is, which do not exceed a given threshold, are filtered.[10] It is thus possible to compute forms that are over-employed (positive specific forms), as well as forms that are under-employed (negative specific forms). The specificity of a genre is then evaluated by its characteristic forms, as well as the forms that do not characterize it at all. Figure 6 shows the specificity indexes of the number of included and tail prosodic packages for the partition *social context* (private vs. public); the tail package is a positive specific form in private speech (over-represented) and a negative specific form in public speech (under-represented); the reverse is true for the included package, although the number of included packages is less specific for this partition (Chapter 17, Section 4.1).

[8] While, in the Rhapsodie project, we focused on discourse genres, any type of metadata can in fact be explored (geographical information, sociolinguistic information, etc.).

[9] The specificity is the logarithm of the probability for the given form to have a frequency in the given section that is at least as far from a uniform distribution on the whole corpus as the observed frequency. The uniform distribution is given by the hypergeometric law, which is also used in Fisher's exact test (Fisher 1922). In other words, if x has a specificity index equal to 5, the probability that x has the observed frequency or a greater frequency in the given section is less than 10^{-5}. If x has a specificity index equal to -5, the probability that x has the observed frequency or a lesser frequency in the given section is less than 10^{-5}.

[10] The threshold is generally fixed at ±2, as in Fisher's exact test. It could also be interesting to study the forms that have a uniform distribution among the sections and could be considered to be specific for the corpus as a whole.

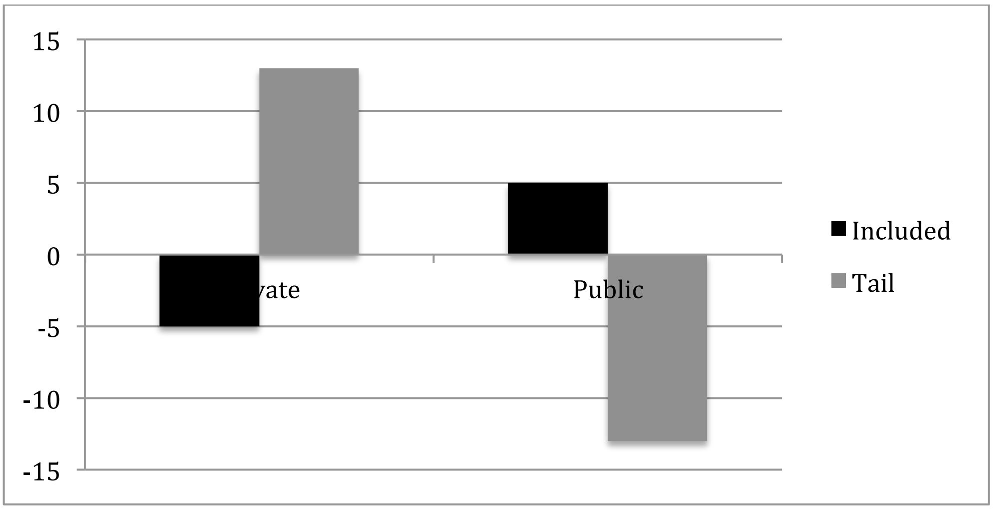


Figure 6. Specificity indexes of the number of included and tail prosodic packages for the partition 'type of speech' (private vs. public)

Furthermore, Trameur can cut the sections into sub-sections, providing an elegant display of the distribution of a given feature or combination of features in the sub-sections of a text.

For example, samples can be cut into IPEs and IUs in order to see their synchronization in a sample or in a category of discourse. Hence, each square represents an IPE in Figure 7a. Squares without a cross represent boundaries of IPEs that do not match with boundaries of IUs, such as for instance the first three squares in Figure 7a corresponding to (3); squares with a cross represent boundaries of IPEs that synchronize with boundaries of IUs. Figure 7b represents the final boundaries of IUs. Only one does not match the end of an IPE in a context of syntactic ambiguity: In (4), *riche* ('rich') was attached to the right by syntax while the prosodic segmentation indicates a left dependency.

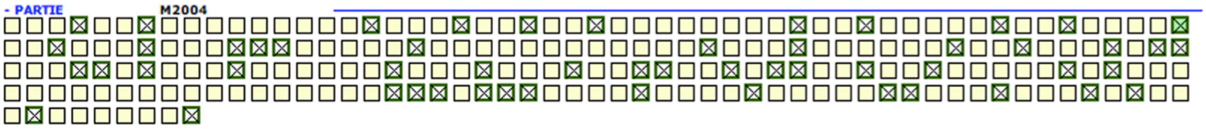


Figure 7a. Distribution of IPEs either synchronized or not with IUs in [Rhap-M2004, Rhapsodie]

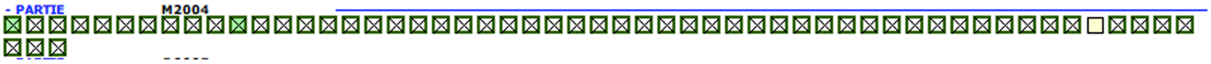


Figure 7b. Distribution of IUs either synchronized or not with IPEs in [Rhap-M2004, Rhapsodie]

(3) [Rhap-M2004, Rhapsodie]

*mes chers compatriotes **IPE** je voudrais d'abord **IPE** exprimer ma sympathie **IPE** à toutes celles et à tous ceux qui vivent ces derniers jours de mille neuf cent quatre-vingt-dix-neuf dans l'épreuve **IPE-IU***

'my dear compatriots **IPE** I would first like **IPE** to express my sympathy to all those who are going through these closing days of nineteen ninety-nine in hardship **IPE-IU**'

(4) [Rhap-M2004, Rhapsodie]
*la France a plus de mille ans **IUF** riche de fièvre de passion **IPEF***

'France has existed for more than a thousand years **IUF** rich with excitement and passion **IPEF**'

Trameur was adapted to take Rhapsodie's data into account. On the one hand, it makes it possible to query multi-layered information and to combine information distributed in different columns of the tabular file. On the other hand, it allows us to take the dependency structure into account, that is, links between words (Fleury & Zimina 2014). Trameur therefore enabled us to query not only prosodic and morphosyntactic features, but also the frequency of various intonosyntactic configurations and their specificity according to genre. For the studies presented in the following chapters, Trameur was mainly used to analyze correlations between discourse genres and prosodic markers (Chapter 17) as well as the intonosyntactic interface in the light of intonational periods and illocutionary units (Chapter 18).

5. Conclusion

In this chapter, after having recapped the main linguistic goals of the Rhapsodie project, we have presented the Rhapsodie data structure and its formal properties. We showed that the main points that require suitable tools to process and query the data are: (i) The non-alignment between the phonological stream on which prosodic structure is anchored and the orthographic stream on which syntactic structure is anchored; (ii) Speech overlaps, which are frequent in our data; (iii) The specificities of each type of linguistic representation: A non-recursive time-aligned prosodic representation and a recursive tree-based representation on a partially ordered list of tokens for the syntactic processing, which cannot be subsumed by each other. Since a generic tool able to query simultaneously both types of structure does not exist, we presented the different tools that have been developed to achieve our purpose: The tabular formatting of Rhapsodie data and the adaptation of Trameur. Chapters 17 and 18, far from being exhaustive about all the points that can be explored regarding the interface between syntax, prosody and discourse, highlight the benefit of such tools for specific and systematic analyses in this field.

It is worth mentioning that Rhapsodie data are freely available and can be exploited by other tools. For instance, they have been included in the on-going project Orfeo (www. projet-orfeo.fr) to train various syntactic parsers and Rhapsodie syntactic data are available on the Orfeo platform based on the ANNIS query system (Krause & Zeldes 2016) as well as on Universal Dependencies (Nivre et al. 2016).