Financé par ANR

EPAC

Exploration de masses de documents audio pour l'extraction et le traitement de la parole conversationnelle

WP5 – Natural Language Processing on the Speech Transcripts

Formats d'échanges de données

version 2.2.1 – 4/01/2008

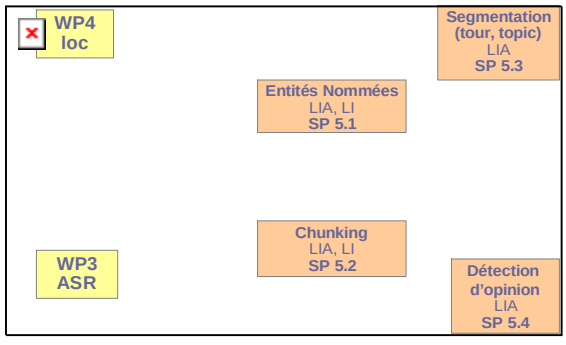
Rapport Technique EPAC_WP5_2008_01

1. Contexte

Un des objectifs du WP5 est l'annotation des transcriptions avec les informations suivantes :

- étiquetage en parties du discours des énoncés transcrits
- étiquetage et segmentation des énoncés en segments syntaxiques non récursifs (chunks)
- détection et typage des entités nommées
- caractérisation du topique des tours de parole
- détection d'opinion (positif, négatif, neutre) par tour de parole.

La figure ci-dessous illustre les flux de données entre les différents modules d'annotation qui seront réalisés dans le cadre du WP. Ce document décrit les formats d'échanges entre ces modules.



2. Données de référence pour les traitements du WP5: transciptions tokenisées

2.1. Transcription orthographique : formats

Les outils de TAL développés dans le cadre du projet travailleront sur des transcriptions orthographiques brutes. Au cours de l'évolution du projet, nous considererons des données différentes en entrée :

1. dans un premier temps, travail sur le corpus de parole transcrit manuellement par le LIUM,
2. ensuite, travail sur meilleure hypothèse sortie par la reconnaissance de la parole (même format)

3. enfin, passage à un travail sur les treillis d'hypothèses

Pour les deux premières étapes, les formats d'entrée seront identiques. On reprendra le format TRS défini comme format de sortie de transcription de Transcriber et correspondant à un découpage en tours de parole. Ce format (DTD XML) ne sera pas décrit dans ce document. On pourra se reporter à la documentation Transcriber. Lorsque celle-ci a un sens (parole préparée), la ponctuation pourra être conservée.

2.2. Format de référence des niveaux de TAL : tokenisation

Plutôt que de rajouter à chaque niveau de traitement une couche d'annotation XML au corpus, il a été décidé que chaque outil fournirait en sortie une annotation correspondant uniquement à son niveau de description. L'alignement entre toutes les annotations se fera en référence à une segmentation de la transcription orthographique en tokens.

Le format retenu pour cette référence en entrée s'inspire du projet LUNA auquel a participé le LIA. A la transcription orthographique Transcriber de base est rajouté une segmentation en phrase et tokens.

Par défaut, chaque tour de parole ne contient qu'une phrase, mais un tour peut-être découpé en plusieurs phrases si nécessaire. Chaque phrase est décrite par une balise `<sentence>` avec comme attribut un identificateur qui correspond à l'ordre d'apparition de la phrase dans le corpus. Un champ <text> décrit ensuite la transcription orthographique correspondant à cette phrase. Viennent ensuite le champ <token> qui donne la segmentation de la phrase en tokens (et donne au passage le nombre d'entités ainsi segmentées.

Un token peut-être un tag XML (par exemple un `<sync time>` de la DTD Transcriber), un caractère de ponctuation, un espace, ou une suite de caractères correspondant à une entité lexicale simple. Cette information est portée par l'attribut `type` associé à la balise `<token>`. On distingue ainsi les types `sgmltag, space, wtoken` et `poncts`. A chaque token est associé un identificateur `id`. Cet identificateur porte le numéro de la phrase où se trouve le token suivi du numéro du token (précédé de la lettre `t`), suivant son ordre d'apparition, dans la phrase considérée.

Voici un exemple de corpus tokenisé pour illustrer le format. On notera que la segmentation en mots pointe également sur la segmentation en tokens.

```
<Turn startTime="0" endTime="2.933" speaker="spk1" mode="spontaneous" fidelity="high"
channel="studio">
<sentence id="s0001">
<text>bien .</text>
<tokens count="5">
<token id="s0001_t0001" type="sgmltag">
  <Sync time="0" />
  </token>
  <token type="space" id="s0001_t0002" />
  <token type="wtoken" id="s0001_t0003">bien</token>
  <token type="space" id="s0001_t0004" />
  <token type="poncts" id="s0001_t0005">.</token>
  </tokens>
</sentence>
<sentence id="s0002">
...
```

Le fichier de segmentation reprendra la racine du nom du fichier transcrit, avec extension `.tk.xml`. Pour rappel, cette racine est également la même que celle du fichier audio. C'est pourquoi l'entête du fichier de segmentation précisera ces informations en reprenant l'entête définie par la DTD Transcriber.

```
<Token audio_filename="200409_0455_0535_INTER_ELDA" version="10" version_date="070522">
```

3. Segmentation en mots et étiquetage en partie du discours

Au contraire des tokens, un mot ne peut être qu'un signe de ponctuation, une entité lexicale simple ou une entité polylexicale (*pomme de terre*). La segmentation en mots de la transcription consiste donc à définir les limites de ces entités par référence à la segmentation princeps en tokens. A chaque mot est ensuite associé sa partie du discours.

L'annotation en parties du discours (par la suite, POS pour *Part of Speech*) sera réalisée par le LIA à partir d'un outil existant (LIA_TAGG : `www.lia.univ-avignon.fr/chercheurs/bechet/download_fred.html`) ou du toolkit MACAON (`www.lif-sud.univ-mrs.fr/~nasr/macaon/index.html`) développé par Alexis NASR du LIF. Les POS retenus pour l'annotation reprendront donc ceux définis par le LIA, en les adaptant aux besoins du projet et en y rajoutant des catégories propres au langage oral (fragments et marques

d'hésitations pour pauses remplies).

3.1. Liste des POS du projet EPAC

- *Catégories utilisées actuellement par LIA_TAGG*

ADV	adverbe
ADVNE	adverbe "ne"
ADVPAS	adverbe "pas"
AFP	adjectif feminin pluriel
AFS	adjectif feminin singulier
AMP	adjectif masculin pluriel
AMS	adjectif masculin singulier
AINDFP	adjectif indefini feminin pluriel
AINDFS	adjectif indefini feminin singulier
AINDMP	adjectif indefini masculin pluriel
AINDMS	adjectif indefini masculin singulier
CHIF	chiffre ou nombre
COCO	conjonction de coordination
COSUB	conjonction de subordination
DETFP	determinant feminin pluriel
DETFS	determinant feminin singulier
DETMP	determinant masculin pluriel
DETMS	determinant masculin singulier
DINTFP	interrogatif feminin pluriel
DINTFS	interrogatif feminin singulier
DINTMP	interrogatif masculin pluriel
DINTMS	interrogatif masculin singulier
MOTINC	mot inconnu (comprend également les inachèvements)
NFP	nom feminin pluriel
NFS	nom feminin singulier
NMP	nom masculin pluriel
NMS	nom masculin singulier
PDEMFP	pronom demonstratif feminin pluriel
PDEMFS	pronom demonstratif feminin singulier
PDEMMP	pronom demonstratif masculin pluriel
PDEMMS	pronom demonstratif masculin singulier
PINDFP	pronom indefini feminin pluriel
PINDFS	pronom indefini feminin singulier
PINDMP	pronom indefini masculin pluriel
PINDMS	pronom indefini masculin singulier
PINTFP	pronom interrogatif feminin pluriel
PINTFS	pronom interrogatif feminin singulier
PINTMP	pronom interrogatif masculin pluriel
PINTMS	pronom interrogatif masculin singulier
PPER1S	pronom personnel premiere personne du singulier
PPER1P	pronom personnel premiere personne du pluriel
PPER2S	pronom personnel deuxieme personne du singulier
PPER2P	pronom personnel deuxieme personne du pluriel
PPER3FS	pronom personnel troisieme personne du feminin singulier
PPER3FP	pronom personnel troisieme personne du feminin pluriel
PPER3MS	pronom personnel troisieme personne du masculin singulier
PPER3MP	pronom personnel troisieme personne du masculin pluriel
PPOBJFP	pronom personnel objet feminin pluriel
PPOBJFS	pronom personnel objet feminin singulier
PPOBJMP	pronom personnel objet masculin pluriel
PPOBJMS	pronom personnel objet masculin singulier
PREFFP	pronom reflechi feminin pluriel
PREFFS	pronom reflechi feminin singulier
PREFMP	pronom reflechi masculin pluriel
PREFMS	pronom reflechi masculin singulier
PRELFP	pronom relatif feminin pluriel
PRELFS	pronom relatif feminin singulier
PRELMP	pronom relatif masculin pluriel

PRELMS	pronom relatif masculin singulier
PREP	preposition
PREPADE	preposition "a/de"
PREPAU	preposition "au"
PREPAUX	preposition "aux"
PREPDES	preposition "des"
PREPDU	preposition "du"
V1P	verbe premiere personne du pluriel
V1S	verbe premiere personne du singulier
V2P	verbe deuxieme personne du pluriel
V2S	verbe deuxieme personne du singulier
V3S	verbe troisieme personne du singulier
V3P	verbe troisieme personne du pluriel
VA1P	auxiliaire "avoir" premiere personne du pluriel
VA1S	auxiliaire "avoir" premiere personne du singulier
VA2P	auxiliaire "avoir" deuxieme personne du pluriel
VA2S	auxiliaire "avoir" deuxieme personne du singulier
VA3P	auxiliaire "avoir" troisieme personne du pluriel
VA3S	auxiliaire "avoir" troisieme personne du singulier
VAINF	auxiliaire "avoir" infinitif
VE1P	auxiliaire "etre" premiere personne du pluriel
VE1S	auxiliaire "etre" premiere personne du singulier
VE2P	auxiliaire "etre" deuxieme personne du pluriel
VE2S	auxiliaire "etre" deuxieme personne du singulier
VE3P	auxiliaire "etre" troisieme personne du pluriel
VE3S	auxiliaire "etre" troisieme personne du singulier
VEINF	auxiliaire "etre" infinitif
VINF	verbe infinitif
VPPFP	verbe participe passe feminin pluriel
VPPFS	verbe participe passe feminin singulier
VPPMP	verbe participe passe masculin pluriel
VPPMS	verbe participe passe masculin singulier
VPPRE	verbe participe present
XFAMIL	nom de famille
XPAYFP	nom de pays feminin pluriel
XPAYFS	nom de pays feminin singulier
XPAYMP	nom de pays masculin pluriel
XPAYMS	nom de pays masculin singulier
XPREF	prenom feminin
XPREM	prenom masculin
XSOC	nom d'organisation
XVILLE	nom de ville
YPFAI	ponctuation faible
YPFOR	ponctuation forte
ZTRM	marque de debut <s> et fin </s> de phrase

- *Catégories ajoutées pour le projet EPAC*

HES	marques d'hésitation (« euh ») et autres pauses remplies

Attention : d'autres catégories sont appelées à être ajoutées suivant les besoins du projet. Elles devraient avant tout correspondre à un découpage des catégories existantes en sous-catégories.

3.2. Format de sortie: transcriptions annotées

Le format de sortie donne la segmentation de la transcription orthographique en mots, associés à leurs POS. La forme de chaque mot est défini entre une balise `<word>` ouvrante et une balise `</word>` fermante. Les différentes propriétés du mot sont définis par les attributs de la balise `<word>`. Chaque mot se voit attribuer un identificateur défini par l'attribut `id`. Ce dernier porte le numéro de la phrase où se trouve le mot suivi du numéro du mot (précédé de la lettre `w`), suivant son ordre d'apparition, dans la phrase considérée.

L'alignement sur la segmentation de référence en tokens est faite à l'aide de l'attribut `token`, qui décrit la position du début du mot. Enfin, l'annotation en POS est décrite par un attribut `pos`, qui donne l'étiquette associée au mot. La segmentation en tours de parole, phrases et tokens n'est plus présente explicitement dans le format de sortie, dont on donne un exemple :

```
<word id="s0001_w0001" token="s0001_t0003" pos="ADV"> bien </word>
```

Le fichier d'annotation en partie du discours reprendra la racine du nom du fichier de token, avec extension `.tag.xml`. Pour rappel, cette racine est également la même que celle du fichier audio. C'est pourquoi l'entête du fichier de segmentation précisera ces informations en reprenant l'entête définie par la DTD Transcriber. S'y ajouteront les informations concernant l'outil utilisé pour réaliser l'annotation et le type d'annotation (balise `type`) défini parmi les trois choix suivants :

- `type= "AUTO"` annotation automatique
- `type= "REVISED"` annotation automatique avec révision manuelle
- `type= "MANUAL"` annotation manuelle

Voici un exemple d'entête (partie additionnelle à l'entête Transcriber) :

```
<Tags tagger="LIA_TAGG" type="AUTO" audio_filename="200409_0455_0535_INTER_ELDA"
version="10" date="070522">
```

4. Segmentation de l'énoncé : chunking

La segmentation en chunks sera réalisée parallèlement par le LI et le LIA. L'objectif est ici de segmenter l'énoncé en segments syntaxiques non récursifs, mais également de typer ces chunks. Le LIA dispose d'un segmenteur assez simple pour le moment, et le LI va reprendre des travaux réalisés précédemment par JY Antoine au laboratoire VALORIA. L'existant est donc assez limité, d'où le choix de repartir de zéro au niveau des formats d'annotation et de s'inspirer du paradigme d'évaluation EASY/PEAS.

4.1. Format d'entrée

Les segmenteurs travailleront en entrée sur les transcriptions annotées en POS par le LIA et sur le format de référence en tokens défini au paragraphe 2.1.

4.2. Définition des chunks dans le projet EPAC

Que ce soit au niveau de la granularité de segmentation comme de celui de la catégorisation des segments, on suivra la norme, dans sa version 1.6, établie pour le français par le paradigme d'évaluation EASY / PEAS. Pour plus de précisions, on peut se référer au dernier guide d'annotation (version à rappeler) défini pendant la campagne et disponible à l'adresse suivante : www.limsi.fr/Recherche/CORVAL/easy/

Il faut par contre rajouter quelques définitions de chunks pour adapter le formalisme à l'oral. D'où :

- Catégories de chunks reprises du paradigme PEAS

NV	noyau verbal	verbe et ses clitiques
PV	groupe verbal introduit par une préposition	« il faut vivre *pour manger* »
GN	groupe nominal	
GP	groupe prépositionnel	
GA	groupe adjectival	uniquement adjectifs postposés
GR	groupe adverbial	

- Catégories ajoutées pour le projet EPAC

COO	marque de coordination	non étiqueté dans PEAS, ajout pour cohérence
ED	zone d'édition d'une reprise ou réparation	« une petite [*euh non attends*] une petite fille »
REP	zone fragmentaire avant une zone d'édition	« [*une* petite] euh non attends une petite fille »

Si le reparandum n'est pas fragmentaire et constitue donc un chunk complet, il est étiqueté comme tel.

4.3. Format de sortie

La segmentation en chunks revient à regrouper certains mots et à leur associer une catégorie. Chaque chunk est défini par une balise `<chunk>` qui encadre le type de chunk (catégories décrites au § 4.2.) et à laquelle sont associés plusieurs attributs:

- `token_deb` et `token_fin` font le lien avec la segmentation de référence en token. Ils pointent les tokens qui correspondent au début et à la fin du chunk considéré. Etant donné que la segmentation n'est pas obligatoirement complète (certaines zones du texte peuvent ne pas être parenthésées), il est nécessaire de préciser où s'arrête le chunk.
- `word_deb` et `word_fin` font le lien à toutes fins utiles avec la segmentation en mots. Cette information est redondante avec la précédente. Elle est seulement précisée afin de faciliter d'éventuels traitements ultérieurs.

- `id` est l'identificateur du chunk. Il porte le numéro de la phrase où se trouve le chunk suivi du numéro de chunk (précédé de la lettre `c`), suivant son ordre d'apparition, dans la phrase considérée.

L'exemple suivant illustre la caractérisation de deux chunks successifs mais non jointifs :

```
<chunk     token_deb="s0003_t0008"     word_deb="s0003_w0003"     token_fin="s0003_t0012"
word_fin="s0003_w0006" id="s0003_c0001" > GN </chunk>
<chunk     token_deb="s0003_t00018"     word_deb="s0003_w0008"     token_fin="s0003_t0022"
word_fin="s0003_w0009" id="s0003_c0002" > NV </chunk>
```

Le fichier d'annotation en chunk reprend la racine du nom du fichier de tokens, avec extension `.chk.xml`. Pour rappel, cette racine est également la même que celle du fichier audio. C'est pourquoi l'entête du fichier de segmentation précisera ces informations en reprenant l'entête définie par la DTD Transcriber. S'y ajouteront les informations concernant l'outil utilisé pour réaliser l'annotation et le type d'annotation (balise `type`) défini parmi les trois choix suivants :

- `type= "AUTO"` annotation automatique
- `type= "REVISED"` annotation automatique avec révision manuelle
- `type= "MANUAL"` annotation manuelle

Voici un exemple d'entête :

```
<ChunksDescription chunker="LI_CASSYS" type="AUTO"
audio_filename="200409_0455_0535_INTER_ELDA" version="1" date="070522">
</ChunksDescription>
```

5. Annotation des entités nommées.

Le LI et le LIA doivent développer des outils de détection des entités nommées (EN par la suite). Le LI a développé un système (Cassys) qui privilégie la précision sur le rappel. Les deux systèmes participeront à la campagne d'évaluation ESTER 2. Les formats d'échanges retenus dans le cadre du projet EPAC s'appuient donc sur la typologie des entités nommées retenue pour cette campagne d'évaluation.

5.1. Formats d'entrée

Suivant l'approche utilisée par les systèmes, on considérera comme entrée les sorties du tagger POS seul ou des segmenteurs en chunks. On reprend donc les formats de sortie définis pour ces modules.

5.2. Typologie des entités nommées dans le projet EPAC

Les entités nommées sont catégorisées suivant la typologique retenue pour la campagne ESTER2, qui est encore susceptible de légères modifications. La typologie ESTER2 repose sur une arborescence en deux niveaux, ce qui fait qu'une entitée pourra être étiquetée avec son super type (exemple : pers) ou son type précis (exemple : pers.hum). En résumé, voici la liste de types ESTER2 retenus au 20/06/2008 :

Les super types sont :

Super type	Format
Personne	pers
fonction	fonc
organisation	org
lieu	loc
production humaine	prod
date et heure	time
montant	amount

On peut ensuite préciser les super-types avec les types suivants :

Type	Format
personne _ humain réel ou fictif _ animal réel ou fictif	pers pers.hum pers.anim
Fonction _ politique _ militaire _ administrative _ religieuse _ aristocratique	fonc fonc.pol fonc.mil fonc.admi fonc.rel fonc.ari

organisation _ politique _ éducative _ commerciale _ non commerciale _ média & divertissement _ géo-socio-administrative	org org.pol org.edu org.com org.non-profit org.div org.gsp
Lieu _ géographique naturel _ région administrative _ axe de circulation _ adresse adresse postale téléphone et fax adresse électronique _ construction humaine	loc loc.geo loc.admi loc.line loc.addr loc.addr.post loc.addr.tel loc.addr.elec loc.fac
production humaine _ moyen de transport _ récompense _ oeuvre artistique _ production documentaire	prod prod.vehicule prod.award prod.art prod.doc
date et heure _ date date absolue date relative _ heure	time time.date time.date.abs time.date.rel time.hour
montant _ âge _ durée _ température _ longueur _ surface et aire _ volume _ poids _ vitesse _ autre _ valeur monétaire	amount amount.phy.age amount.phy.dur amount. phy.temp amount. phy.len amount.phy.area amount.phy.vol amount. phy.wei amount. phy.spd amount.phy.other amount.cur

Nous proposons actuellement des modifications des conventions ESTER 2. Les conventions d'annotation complètes se trouvent à l'adresse suivante :

http://www.afcp-parole.org/ester/docs/Conventions_EN_ESTER2_v01.pdf

5.3. Formats de sortie

Le balisage des entités nommées revient à encadrer la séquence de mots formant l'entité nommée, caractériser sa position dans le texte (par rapport à la référence commune en tokens) et à lui associer une catégorie. Chaque entité nommée est donc définie par une balise <EN> qui encadre la séquence de mots et à laquelle sont associés plusieurs attributs:

`token_deb` et `token_fin` font le lien avec la segmentation de référence en token. Ils pointent les tokens qui correspondent au début et à la fin de l'entité nommée considérée. Contrairement au format ESTER2, la sortie correspond à l'extraction des entitées nommées et non pas leur balisage dans le texte. Les parties du texte ne formant pas d'entité nommée n'apparaissant plus dans le format de sortie, il est nécessaire de préciser où s'arrête l'entité (position précisée par rapport à la référence en tokens).

• `word_deb` et `word_fin` font le lien à toutes fins utiles avec la segmentation en mots. Cette

information est redondante avec la précédente. Elle est seulement précisée afin de faciliter

d'éventuels traitements ultérieurs.

• id est l'identificateur de l'entité nommée. Il porte le numéro de la phrase où se trouve l'entité suivi du numéro de chunk (précédé de la suite de lettres en), suivant son ordre d'apparition, dans la phrase considérée. L'exemple suivant illustre la caractérisation d'une entité nommée simple :

```
<EN typ="pers.hum" token_deb="s0003_t0008" word_deb="s0003_w0003"
token_fin="s0003_t0011" word_fin="s0003_w0006" id="s0003_c0001" > Gonzague de Saint
Bris</EN>
```

Une entité nommée peut parfois se décomposer en deux entités imbriquées. Par exemple, l'entité complexe *Université François Rabelais Tours* encapsule le toponyme *Tours*. ESTER2 a choisi de représenter directement ces imbrications dans le balisage. Le format EPAC fait de même. Voici un exemple d'entités nommées imbriquées :

```
<EN type="org.edu" token_deb="s0003_t0008" word_deb="s0003_w0003" token_fin="s0003_t0011"
word_fin="s0003_w0006" id="s0003_c0001" > Universite Francois Rabelais <EN
type="loc.admi"
token_deb="s0003_t00011" word_deb="s0003_w0006" token_fin="s0003_t0011"
word_fin="s0003_w0006" id="s0003_c0002" > Tours </EN></EN>
```

Le fichier d'annotation en entités nommées reprend la racine du nom du fichier de tokens, avec extension .en.xml. Pour rappel, cette racine est également la même que celle du fichier audio. C'est pourquoi l'entête du fichier de segmentation précisera ces informations en reprenant l'entête définie par la DTD Transcriber. S'y ajouteront les informations concernant l'outil utilisé pour réaliser l'annotation et le type d'annotation (balise type) défini parmi les trois choix suivants :

• type= "AUTO" annotation automatique

• type= "REVISED" annotation automatique avec révision manuelle

• type= "MANUAL" annotation manuelle

Voici un exemple d'entête (partie additionnelle à l'entête Transcriber) :

```
<entites sys="LI_EN" type="AUTO" audio_filename="200409_0455_0535_INTER_ELDA"
version="10" date="070522">
```

6. Crédits

La version de ce texte a été définie par Jean-Yves ANTOINE (LI, U. Tours), Abdenour MOKRANE (LI, U. Tours), Nathalie FRIBURGER (LI, U. Tours) et Frédéric BECHET (LIA, U. Avignon).