Collaborative Annotation for Reliable Natural Language Processing: Technical and Sociological Aspects

Karën Fort

HAL Id: hal-01324322
https://hal.science/hal-01324322

Submitted on 19 Nov 2020

Collaborative Annotation
for Reliable Natural Language Processing

Karën Fort

February 12, 2016 14:47

Contents

Preface

This book represents a unique opportunity for me to construct what I hope to be a consistent image of collaborative manual annotation for Natural Language Processing (NLP). I partly rely on work that has already been published elsewhere, some of it only in French, most of it in reduced versions, and all of it available on my personal Website[1]. Whenever possible, the original article should be cited in preference to this book.

Also, a number of citations refer to publications in French. I kept those when there is no equivalent in English, hoping that at least some readers will be able to understand them.

This work owes a lot to my interactions with Adeline Nazarenko (LIPN / University of Paris 13), during my PhD thesis and after. Besides, it would not have been conducted to its end without (a lot of) support and help from Benoît Habert (ICAR / ENS of Lyon).

---

[1] Here: `http://karenfort.org/Publications.php`.

Chapter 1

Introduction

1.1 Natural Language Processing and Manual Annotation: Dr Jekyll and Mr Hy|ide?

1.1.1 Where Linguistics Hides

Natural Language Processing (NLP) has witnessed two major evolutions in the past 25 years: first, the extraordinary success of machine learning, which is now, for better or for worse (for an enlightening analysis of the phenomenon see [Church, 2011]), overwhelmingly dominant in the field, and second, the multiplication of evaluation campaigns or shared tasks. Both involve manually annotated corpora, for the training and evaluation of the systems (see Figure 1.1).

These corpora progressively became the hidden pillars of our domain, providing food for our hungry machine learning algorithms and reference for evaluation. Annotation is now the place where linguistics hides in NLP.
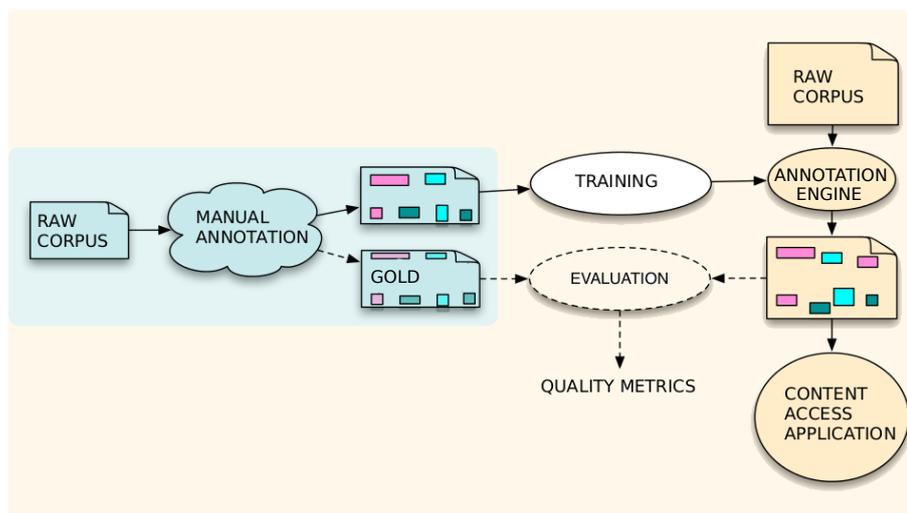
Figure 1.1: Manually annotated corpora and machine learning systems.

However, manual annotation has largely been ignored for quite a while and it took some time even for annotation guidelines to be recognized as essential [Nédellec et al., 2006]. When the performance of the systems began to stall, manual annotation finally started to generate some interest in the community, as a potential leverage for improving the obtained results [Hovy and Lavid, 2010, Pustejovsky and Stubbs, 2012].

This is all the more important as it was proven that systems trained on badly annotated corpora underperform. In particular, they tend to reproduce annotation errors when these errors follow a regular pattern and do not correspond to simple noise [Reidsma and Carletta, 2008]. Furthermore, the quality of manual annotation is crucial when it is used to evaluate NLP systems. For example, an inconsistently annotated reference corpus would undoubtedly favor machine learning systems, therefore prejudicing rule-based systems in evaluation campaigns. Finally, the quality of linguistic analyses would suffer from an annotated corpus that is unreliable.

Although some efforts have been done lately to address some of the issues presented by manual annotation, there is still little research done on the subject. This book aims at providing some (hopefully useful) insights on the subject. It is partly based on a PhD thesis [Fort, 2012] and on some published articles, most of them written in French.

1.1.2   What is Annotation?

The renowned British corpus linguist Geoffrey Leech [Leech, 1997] defines corpus annotation as: "[T]he practice of adding interpretative, linguistic information to an electronic corpus of spoken and/or written language data. "Annotation" can also refer to the end-product of this process." This definition highlights the interpretative dimension of annotation but limits it to "linguistic information" and to some specific sources, without mentioning its goal.

In [Habert, 2005], Benoît Habert extends Leech's definition, first by not restricting the type of added information: "annotation consists in adding information (a stabilized Interpretation) to language data: sounds, characters and gestures."[1] He adds that "it associates two or three steps: (i) segmentation to delimit fragments of data and/or add specific points; (ii) grouping of segments or points to assign them a category; (iii) (potentially) creating relations between fragments or points".[2]

We build on these and provide a wider definition of annotation:

Definition 1 (Annotation) *annotation covers both the process of adding a note on a source signal and the whole set of notes or each note that results from this process, without* a priori *presuming what would be the nature of the source (text, video, images, etc), the semantic content of the note (numbered note, value chosen in a reference list or free text), its position (global or local), or its objective (evaluation, characterization, simple comment).*

---

[1]In French, the original version is: "l'annotation consiste à ajouter de l'information (une interprétation stabilisée) aux données langagières : sons, caractères et gestes.".

[2]In French: "[e]lle associe deux ou trois volets : (i) segmentation pour délimiter des fragments de données et/ou ajout de points singuliers ; (ii) regroupement de segments ou de points pour leur affecter une catégorie ; (iii) (éventuellement) mise en relation de fragments ou de points".

Basically, annotating is adding a *note* to a *source signal*. The annotation is therefore the note, *anchored* in one point or in a *span* of the source signal (see Figure 1.2). In some cases, the span can be the whole document (for example, in indexing).
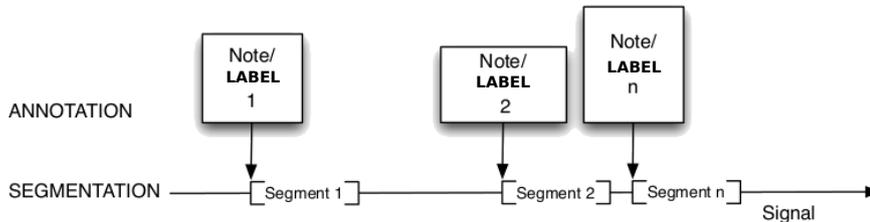
Figure 1.2: Anchoring of notes in the source signal.

In the case of relations, two or more spans of the source signal are connected and a note is added to the connection. Often, a note is added to the spans (or segments) too.

This definition of annotation includes many NLP applications, from transcription (the annotation of speech with its written interpretation) to machine translation (the annotation of one language with its translation in another language). However, the analysis we conduct here is mostly centered on categorization (adding a category taken from a list to a segment of signal, or between segments of signal). It does not mean that it does not apply to transcription, for example, but we have not yet covered this thoroughly enough to be able to say that the research detailed in this book can directly apply to such applications.

In NLP, annotations can either be added manually by a human interpreter or automatically by an analysis tool. In the first case, the interpretation can reflect parts of the subjectivity of its authors. In the second case, the interpretation is entirely determined by the knowledge and the algorithm integrated in the tool. We are focusing here on manual annotation as a task executed by human agents (annotators).

1.1.3 New Forms, Old Issues

Identifying the first evidence of annotation in history is impossible, but it seems likely that it appeared in the first writings on a physical support allowing for a text to be easily commented upon.

Annotations were used for private purpose (comments from readers) or public usage (explanations from professional readers). They were also used for communicating between writers (authors or copyists, i.e. professional readers) [Bakhouche et al., 2010]. In these latter senses, the annotations had a collaborative dimension.

Early manuscripts contained *glosses*, i.e., according to the on-line Merriam-Webster dictionary[3]: "a brief explanation (as in the margin or between the lines of a text) of a difficult or obscure word or expression ". Glosses were used to inform and train the reader. Other types of annotations were used, for example to update the text (*apostils*). The form of glosses could vary considerably and

---

[3]See 3.1 in `http://www.merriam-webster.com/dictionary/gloss`.

[Muzerelle, 1985, p 134] distinguishes between nine different types. *Interlinear glosses* appeared between the lines of a manuscript, *marginal glosses* in the margin, *surrounding glosses* in the circumference and *separating glosses* between the explained paragraphs. They are more or less merged in the text, from the *organic gloss*, which can be considered as part of the text, to the *formal gloss*, which constitutes a text in itself, transmitted from copy to copy (as today's standoff annotations).[4]

Physical marks, like *indention square brackets*, could also be added to the text (see an example in a text by Virgil[5] in Figure 1.3) indicating the first commented words. Interestingly, this primitive anchoring did not indicate the end of the commented part.
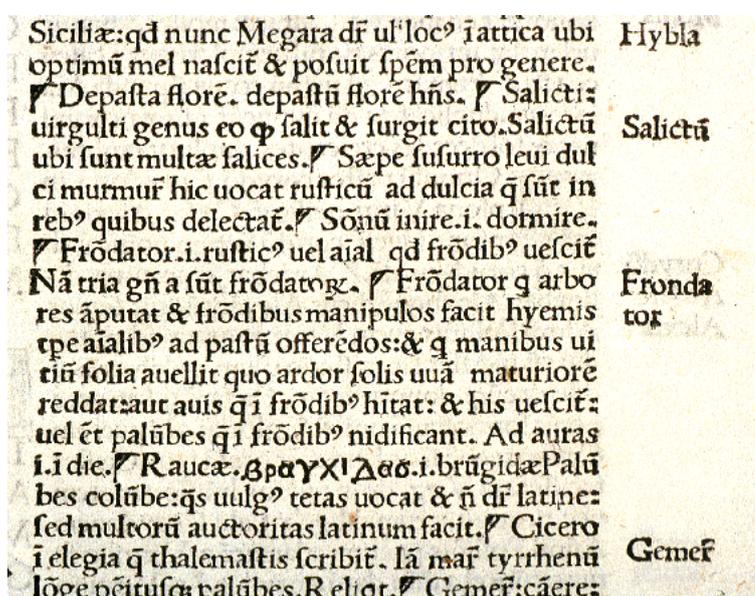
Figure 1.3: Indention square brackets in a text by Virgil. Bibliothèque municipale de Lyon (Courtesy of Town Library of Lyon), France, Res. 104 950.

The same limitation applies to the *auctoritates*, which appeared in the VIIIth century to cite the authors (considered as authorities) of a citation. The anchoring of the annotation is noted by two dots above the first word of the citation, without delimiting the end of it (see Figure 1.4).

This delimitation problem was accentuated by the errors made by the copyists, who moved the *auctoritates* and their anchors. To solve this issue, and without the possibility of using other markers like quotes (which will appear much later), introductory texts (pre and peri-annotation) were invented.

From the content point of view, the evolution went from the *explanatory gloss* (free text) to the citation of authors (name of the author, from a limited list of authorities), precisely identified in the text. As for the anchoring, it

---

[4]For more information and illustrations on the subject, see: `http://codicologia.irht.cnrs.fr`.

[5]Opera, com. de Servius. Milan: Leonardo Pachel, 1509, in-fol., source : `http://enssibal.enssib.fr/bibliotheque/documents/travaux/sordet/nav.liv.ancien.html`
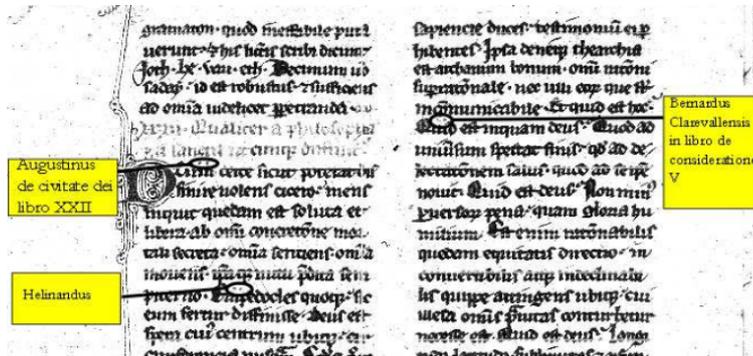
Figure 1.4: Anchoring of *auctoritates* in *De sancta Trinitate*, Basel, UB B.IX.5, extracted from [Frunzeanu and Pons, 2012], by courtesy of the authors.

improved progressively to look like a real text markup.

This rapid overview shows that many of today's preoccupations – frontiers to delimit, annotations chosen freely or from a limited reference, anchoring, metadata to transmit – have been around for a while. It also illustrates the fact that adding an annotation is not a spontaneous gesture, but one which is reflected upon, studied, a gesture which is learned.

1.2 Rediscovering Annotation

1.2.1 A Rise in Diversity and Complexity

A good indicator of the rise in annotation diversity and complexity is the annotation language that is used. The annotation language is the vocabulary used to annotate the flow of data. In a lot of annotation cases in NLP, this language is constrained[6]. It can be of various types.

The simplest is the boolean type. It covers annotation cases in which only one category is needed. A segment is annotated with this category (which can be only implicit), or not annotated at all. Experiences like the identification of obsolescence segments [Laignelet and Rioult, 2009] use this type of language.

Then come the first order languages. Type languages are for example used for morpho-syntactic annotation without features (part-of-speech) or with features (morpho-syntax). The first case is in fact rather rare, as even if the tagset seems little structured, as in the *Penn Treebank* [Santorini, 1990], features can almost always be deduced from it (for example, *NNP*, proper name singular and *NNPS*, proper name plural, could be translated into *NNP+Sg* and *NNP+Pl*).

As for relations, a large variety of them are annotated in NLP today, from binary oriented relations (for example, gene renaming relations [Jourde et al., 2011])

---

[6]Note that this is not the case for transcription or machine translation, which both need unconstrained annotation languages (the natural language itself).

to not oriented n-ary relations (for example, co-reference chains as presented in [Poesio and Artstein, 2005]).

Finally, second order languages could be used, for example to annotate relations on relations. In the soccer domain, for example, *intercept(pass(p1, p2), p3)* represents a pass (relation) between two players (p1 and p2), which is intercepted by another player (p3). In practice, we simplify the annotation by adapting it to a first order language by reifying the first relation [Fort and Claveau, 2012]. This is so commonly done that we are aware of no example of annotation using a second order language.

Jean Véronis concluded his state-of-the-art of the automatic annotation technology in 2000 with a figure summarizing the situation [Véronis, 2000]. On this figure, only the part-of-speech annotation and the multilingual alignment of sentences are considered "operational". Most applications are considered as prototypes (prosody, partial syntax, multilingual words alignment) and the rest was still not allowing for "applications which are useful in real situations" (full syntax, discourse semantics) or was close to prototype (phonetic transcription, lexical semantics). The domain has quickly evolved and today much more complex annotations can be performed, on different media and concerning a large variety of phenomena.

In the past few years, we have witnessed the multiplication of annotation projects involving video sources, in particular sign language videos. A workshop on the subject (DEGELS) took place within the French TALN conference in 2011 and 2012.[7] and a training concerning video corpus annotation was organized by the Association pour le Traitement Automatique des LAngues (ATALA) in 2011.[8]

Moreover, more and more complex semantic annotations are now carried out on a regular base, like opinions or sentiments. In the biomedical domain, proteins and gene names annotation is now completed by the annotation of relations like gene renaming [Jourde et al., 2011] or relations between entities, in particular within the framework of BioNLP shared tasks.[9] Semantic annotations are also performed using a formal model (i.e. an ontology) [Cimiano and Handschuh, 2003] and linked data are now used to annotate corpora, like during the Biomedical Linked Annotation Hackathon (BLAH).[10]

Finally, annotations which are now considered as traditional, like named entities or anaphora, are getting significantly more complex, for example with added structuring [Grouin et al., 2011].

However, there are still few corpora freely available with different levels of annotations, including with annotations from different linguistic theories. *MASC* (*Manually Annotated Sub-Corpus*) [Ide et al., 2008][11] is an interesting exception as it includes, among others, annotations of frames a la FrameNet [Baker et al., 1998] and senses a la WordNet [Fellbaum, 1998]. Besides, we are not aware of any freely available multimedia annotated corpus with each level of annotation aligned to the source, but it should not be long until it is developed.

The ever growing complexity of annotation is taken into account in new annotation formats, like GrAF [Ide and Suderman, 2007] but still has to be in-

---

[7]See: http://degels.limsi.fr/.

[8]See: http://tals.limsi.fr/jatala2011.html.

[9]See the 2016 shared tasks here: http://2016.bionlp-st.org/.

[10]See: http://2015.linkedannotation.org/.

[11]See here: http://www.anc.org/data/masc/corpus/.

tegrated in the methodology and in the preparation of an annotation campaign.

1.2.2 Redefining Manual Annotation Costs

The exact cost of an annotation campaign is rarely mentioned in research papers. One noteworthy exception is the *Prague Dependency TreeBank*, for which the authors of [Böhmová et al., 2001] announce a cost of US$600,000. Other articles detail the number of persons involved in the project they present: *GENIA*, for example, involved 5 part-time annotators, a senior coordinator and one junior coordinator for 1,5 year [Kim et al., 2008]. Anyone who participated in such a project knows it, manual annotation is very costly.

However, the resulting annotated corpora, when they are well-documented and available in a suitable format, as shown in [Cohen et al., 2005], are used well beyond and long after the training of the original model or the original research purpose. A typical example of this is the *Penn TreeBank* corpus, created in the beginning of the 90s [Marcus et al., 1993] and that is still used more than twenty years later (it is easy to find recent research like [Bohnet et al., 2013]). On the opposite, the tools trained on these corpora usually become quickly outdated as research is making progress. An interesting example is that of the once successful `PARTS` tagger, created using the *Brown* corpus [Church, 1988] and used to pre-annotate the *Penn TreeBank*. However, when the technology becomes mature and generates results that the users consider satisfactory, the lifespan of such tools gets longer. This is the case for example in part-of-speech tagging for the `TreeTagger` [Schmid, 1997], which, with nearly 96% of accuracy for French [Allauzen and Bonneau-Maynard, 2008] is still widely used, despite the fact that it is now less efficient then state-of-the-art results (`MElt` [Denis and Sagot, 2009] for example, obtains 98% accuracy on French). Such domains are still rare.

This trivial remark concerning the lifetime of corpora leads to important consequences with regards to the way we build manually annotated corpora.

First, it puts the cost of the manual work into perspective: a manual corpus costing US$600,000 like the *Prague Dependency TreeBank*, that is used for more than twenty years like the *Penn TreeBank*, is not that expensive (US$30,000 per year). It is even cheaper if you consider the whole number of projects which used it: a quick search in the Association for Computational Linguistics (ACL) anthology[12] with the keyword "Penn TreeBank" reveals that more than 30 research articles directly use the corpus (including the *Penn Discourse TreeBank*, but excluding the Penn treebanks created for other languages like Chinese), which corresponds to US$20,000 per project. If we consider that many research projects used it without putting its name in the title of the article, like the paper we wrote on the effects of pre-annotation [Fort and Sagot, 2010], we can assume that a lot more than 30 projects were based on the *Penn TreeBank*, lowering its cost to probably less than that of a long internship.

Second, it is a strong argument not for building manually annotated corpora according to the possibilities of the system(s) that will use it, as they will be long forgotten when the annotated corpus is still be used. If the corpus is too dependent on the systems' (limited) capabilities, it will not be useful anymore when the algorithms become more efficient.

---

[12]See: `http://aclanthology.info/`.

Third, this implies that manual annotation should be of high quality, i.e. well-prepared, well-documented and regularly evaluated with adequate metrics. Manual annotation campaign preparation is often rushed and overlooked, because people want to get it over with as quickly as possible.[13] This has been particularly emphasized in [Sampson, 2000], where the author notes (on page 7) that: "[...] it seems to me that natural language computing has yet to take on board the software-engineering lesson of the primacy of problem analysis and documentation over coding."

There is in fact a need for annotation engineering procedures and tools and this is what this book aims at providing, at least partly.

---

[13]Obviously, there are still people who consider it trivial to annotate manually. They usually change their mind quickly when they start doing it for real.

Chapter 2

Annotating Collaboratively

2.1 The Annotation Process (Re)visited

A simplified representation of the annotation process is presented in Figure 2.4. We will detail further in this section the different steps of this process, but we first introduce a theoretical view on the consensus and show how limited is the state-of-the-art on the subject.

2.1.1 Building Consensus

The central question when dealing with manual corpus annotation is how to obtain reliable annotations, that are both useful (i.e. meaningful) and consistent. In order to achieve this and to solve the "annotation conundrum" [Liberman, 2009], we have to understand the annotation process. As we saw in section 1.1.2, annotating consists in identifying the unit(s) to annotate and to add a note (or a label, a tag) to it or them. In some annotation tasks, units can be linked by a relation, oriented or not, and the note applies to this relation. In most cases, the note is in fact a category, taken from a list (the tagset).

Alain Desrosières, a famous French statistician, worked on the building of the French socio-professional categories [Desrosières and Thévenot, 2002] and wrote a number of books on categorization (among which, translated into English [Desrosières, 1998]). His work is especially relevant to our subject, as he precisely analyzed what categorizing means.

First, and this is fundamental for the annotation process, he makes a clear distinction between *measuring* and *quantifying* [Desrosières and Didier, 2014]. Measuring "implies that something already exists under a form that is measurable, according to a realistic metrology, like the height of the Mont Blanc."[1] Quantifying, on the other hand, consists of "expressing and transforming into a numerical form what used to be expressed with words and not numbers."[2] For this to be realized, a series of conventions of equivalence should be elaborated through collaboration.

---

[1] In French: "[...] l'idée de mesure [...] implique que quelque chose existe *déjà* sous une forme mesurable selon une métrologie réaliste, comme la hauteur du Mont Blanc".

[2] In French: "exprimer et faire exister sous une forme numérique ce qui auparavant, était exprimé par des mots et non par des nombres."

The categories are not measurable, they have to be agreed upon before they can be applied. There has to be a consensus on them and one evidence that the categories emerge from a consensus (and are not "natural") is that they can change in time. A typical example of this are named entities, which evolved from proper names only [Coates-Stephens, 1992] to the MUC (Message Understanding Conferences) classic categories (person, location, organization) [Grishman and Sundheim, 1996] and on to structured named entities, with subtypes and components [Grouin et al., 2011]. This evolution was initiated and validated by the named entity recognition community. This also happened, although in a less spectacular way, with parts-of-speech [Colombat, 1988].

The result of this consensus-building process is logged in the annotation guidelines, that are used by the annotators to decide what to annotate (which segment(s)) and how (with which category). However, even with very detailed guidelines, like the 80 pages long *Quæro* structured named-entities annotation guidelines,[3] the annotators will still disagree on some annotations. This is why we need constant evaluation (to see when they disagree) and consensus building (to improve the consistency of annotations).

Once this posited, there remain many practical issues: who should participate to the annotation guidelines? and how can we determine when they are ready, or at least ready-enough to start annotating? when do we start evaluating the agreement between annotators, and how? The following sections will hopefully provide answers to these questions.[4]

2.1.2 Existing Methodologies

Manual annotation has long been considered as straightforward in linguistics and NLP. Some researchers still consider that computing inter-annotator agreement is useless (since the annotators *have to* agree) and it took some time and demonstration [Nédellec et al., 2006] before the need for an annotation guide became obvious. It is therefore logical that the interest for the manual annotation process itself is growing slowly.

If speech processing inspired the evaluation trend and metrics like inter-annotator agreements, corpus linguistics provided good practices for manual annotation, in particular with Geoffrey Leech's seven maxims [Leech, 1993] and later work on annotation [Leech, 1997], and with the collective efforts like [Wynne, 2005]. However, it did not propose any in-depth analysis of the annotation process itself.

Some high level analyses of the work of the annotators were carried on, for example to create the *UniProt* Standard Operating Procedures[5] or the `GATE` manual.[6] However, very few studies are concerned with the manual annotation process as a whole.

According to Geoffrey Sampson [Sampson, 2000], the "problem analysis and documentation" of annotation should be taken much more seriously and be

---

[3]Available   here: `http://www.quaero.org/media/files/bibliographie/`
`quaero-guide-annotation-2011.pdf`.

[4]The sections about the annotation process, from preparation to finalization, are adapted from my PhD thesis (in French) [Fort, 2012].

[5]See: `http://www.uniprot.org/help/manual_curation` and `http://geneontology.org/`
`page/go-annotation-standard-operating-procedures`.

[6]See: `https://gate.ac.uk/teamware/man-ann-intro.pdf`.

considered primary over coding (annotating). His reflection is based on a parallel with software development and engineering. Interestingly, this parallel has been extended to the annotation methodology with "agile corpus creation" and "agile annotation" [Voormann and Gut, 2008], an analogy with agile development [Beck, 2011].

From our point of view, the methodology presented in [Bonneau-Maynard et al., 2005], even if it is generally not cited as reference for agile annotation, pioneered the field. The authors show that computing inter-annotator agreement very early in the campaign allows to identify problems rapidly and to update the annotation guide accordingly, in order to minimize their impact.

Agile annotation [Voormann and Gut, 2008] goes further as it reorganizes completely the traditional phases of manual annotation (see Figure 2.1) and advocating for a more lenient process, with several cycles annotation/guidelines update. To our knowledge, this methodology was used only once in a real annotation project [Alex et al., 2010]. Therefore, it is difficult to understand to which extend it really differs from the methodology presented in [Bonneau-Maynard et al., 2005] and if it allows to produce better results.
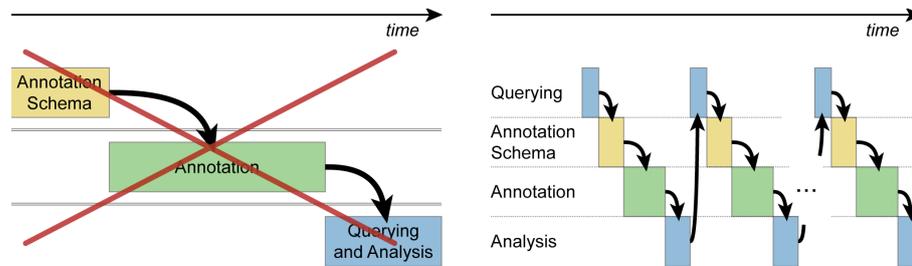
Figure 2.1: Traditional annotation phases (on the left) and cycles of agile annotation (on the right). Reproduction of Figure 2 from [Voormann and Gut, 2008], by courtesy of the authors.

Eduard Hovy presented a tutorial on manual annotation at ACL 2010 during which he gave interesting insights about methodology and process. This partial methodology is detailed in [Hovy and Lavid, 2010] and presented graphically in Figure 2.2. It includes the training and evaluation of the (NLP) system (*engine*), the results of which can lead to modify the manual annotation. Our point of view on this is quite different and we already expressed it in Section 1.2.2: manual annotation should be carried out with an application in mind, not in accordance with a tool, as (i) it would largely bias any evaluation performed with the annotated corpus and (ii) would limit the lifespan of the corpus. However, the manual annotation part of this methodology is the most complete we know of and includes six steps: (1) building the corpus, (2) elaborating the tagset and writing a first version of the guidelines, (3) annotating a sample of the corpus, (4) comparing the annotators' decisions, (5) measuring the inter-annotator agreement and determining which level of agreement would be satisfactory (if not, return to step 2), (6) annotating the corpus. Although it includes a pre-campaign (steps 2 to 5), post-campaign (delivery and maintenance) and consensus building elements (meetings), it does not define who does what (the precise roles), neither does it give indicators allowing to move up one

step, in particular concerning the training of the annotators.
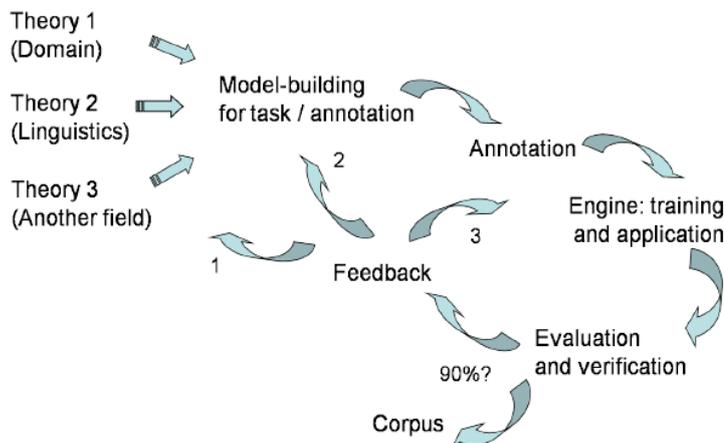
Figure 2.2: Generic annotation pipeline (Figure 1 from [Hovy and Lavid, 2010], by courtesy of the authors).

The book written by James Pustejovsky and Amber Stubbs [Pustejovsky and Stubbs, 2012] also presents a view on annotation where the training of systems and the manual process interpenetrate. This is the MATTER methodology (for *Model-Annotate-Train-Test-Evaluate-Revise*). Within MATTER lies the manual annotation cycle itself, MAMA (for *Model-Annotate-Model-Annotate*). In this, annotation is further decomposed into another cycle: Model and Guidelines - Annotate - Evaluate - Revise. According to MAMA, the corpus is entirely annotated by at least two annotators, several times, then completely adjudicated by an expert. This is ideal, but very costly if not done with crowdsourcing. Another weak point in this methodology is that it contains various cycles and do not indicate when they should be stopped.

We will focus here on the annotation process and will not detail the corpus creation, which is a very important step. Useful information on the subject can be found in [Pustejovsky and Stubbs, 2012], but we advice to read John Sinclair first, in particular the easily accessible [Sinclair, 2005].

We participated in a dozen annotation campaigns (in half of them, as campaign manager), most of them within the framework of the Quæro project. These campaigns cover a wide range of annotation types (and were carried out either in French or in English): POS, dependency syntax, named entities, gene renaming relations, protein and gene names, football, pharmacology. They allowed us to build and test the annotation process framework we propose in the following sections.

2.1.3   Preparatory Work

An annotation campaign does not start with the annotation itself, it requires some preparatory work to identify the actors, to get to know the corpus and to write a first version of the guidelines. This should not be neglected, as the

productivity and the quality of the annotation largely depend on it.

Identifying the Actors

The following reflection was started with Sophie Rosset (LIMSI-CNRS). It aims at identifying clearly the various actors in an annotation campaign and to show the tensions that can emerge from their often diverging visions of the campaign.

Our experience and the state-of-the-art bring us to distinguish between five main roles in an annotation campaign:

1. the final users: users of the potential application developed using the annotated corpus;

2. the financier(s): person(s) representing the organism(s) funding the campaign (often funding agencies);

3. the client(s): person(s) or team(s) who need the corpus to train, create or evaluate their system;

4. the campaign manager: person in charge of planning the campaign and guarantying its performance. In general, the manager is the contact person between the client, the evaluators and the experts (and in some rarer cases the financiers);

5. the expert annotators: annotators who are specialized in the domain of the corpus (sometimes in the annotation task), who select the annotators, train them, evaluate them, answer their questions and adjudicate the annotation when necessary;

6. the annotators: persons performing the biggest part of the annotation; in crowdsourcing annotation they are sometimes called "non-experts", but we will see in the next chapter that it is far from the reality;

7. the evaluator(s): person(s) in charge of evaluating the quality of the annotated corpus and/or of the systems trained or evaluated with this corpus.

All these roles are not always fulfilled in an annotation campaign. For example, evaluators can be absent from smaller internal campaigns, with no external evaluation. As for the number of actors per role, it can vary, but the annotation manager should be a unique person, in order to avoid incoherences in the campaign. In case where s/he is also an expert there can be only one expert, but there should be at least two experts to build a mini-reference. Finally, we will see in Section 2.4 that to evaluate the quality of the annotation, at least two annotators are needed. In the meta study presented in [Bayerl and Paul, 2011], the authors conclude by suggesting to use at least five annotators for the most difficult tasks, and at least three or four for other tasks. Before them it wad shown in [Klebanov and Beigman, 2009] that using more annotators allowed to lower the influence of chance on the inter-annotator agreement results. However, it was demonstrated in [Bhardwaj et al., 2010] that well-trained annotators produce better annotations than a "crowd of non-experts". As we will see further, the number of annotators is not the key, training is.

Obviously, all the roles we presented here should not necessarily be perfectly distinct and organized as a hierarchy in all campaigns, but when they are missing

or merged with another role, it should be taken into account, since it may generate biases. For example, the financier, being in the most powerful position, could bias the whole campaign. Therefore, s/he should not intervene directly on the annotators or the annotation guide. Besides, when the financier is not the client, the balance of the campaign is more sound. With no financier, the client has a lot of influence (maybe too much). As for the manager, s/he is accountable for the overall balance of the campaign and should not play any other role. If s/he is also an expert (which is often the case), s/he has to work with other experts to compensate this imbalance. Finally, the expert has not only to supervise the annotators, but also to be their representative. It should be noted that, although the annotator is at the bottom of this organization, s/he is at the center of the annotation, as the value added to the corpus is the interpretation provided by the annotators. It is therefore essential to take their remarks and suggestions into account. Annotators on microworking platforms like `Amazon Mechanical Turk` are paid by the task and have no interest in giving feedback to the *Requester* (and the platform does not encourage them to do so), their point of view is therefore seldom considered.

Finally, we would like here to nuance a common statement according to which the researchers, who are often both managers of the campaign and experts of the annotation task, are the best annotators. Our experience shows that (i) even if they are experts of the task, they are not necessarily experts of the domain and can experience difficulties understanding the context, like in the case of names entity annotation in old press (what was "Macé" in "krach Macé"? A person? An organization? A place?) and (ii) they too often question the annotation guide they wrote or they do not consult it enough. During the structured named entity annotation campaign in broadcast news, the four experts (researchers) who annotated the mini-reference obtained inter-agreement scores which were not better than that of the annotators.

Taking the Corpus into account

We managed a football annotation campaign in which the heterogeneity of the corpus affected all the aspects of the campaign: the selection of a sub-corpus for the training of the annotators, the length of the training, the complexity of the annotation scheme and the resulting annotation quality. We showed in [Fort et al., 2011b] how important it is to have an in-depth knowledge of the corpus to annotate.

The campaign manager does not necessarily choose the corpus on which the annotation will be performed, s/he therefore has to adapt the campaign to the specificities of the source. This means that the corpus should be analyzed and decomposed into its constituents: domains, sources, media, etc.

The best way to "dive into" the corpus is to annotate a small but representative part of it, even before starting the campaign. Obviously, this is possible only if the domain and the language are mastered by the manager. If not, s/he should use one or several experts of the domain (or the language) to help with this work.

This is what we did in several annotation campaigns, as campaign manager [Fort et al., 2011b] or as advisor [Rosset et al., 2012]. It allowed us not only to identify problems with the annotation guide even before the annotators started working, but also to create a pre-reference for evaluation. In some

cases, having the client annotate this pre-reference is a good way to validate the choices that were made and to check that the annotation is not diverging too much from the initial application.

Whether it is done *a priori*, during the corpus selection, or *a posteriori*, once the corpus is selected, a precise analysis of its contents and of the consequences of this on the campaign has to be performed as soon as possible.

Creating and Modifying the Annotation Guide

The annotation guide (also called annotation guidelines) is now recognized as essential to an annotation campaign. For the structured named entity annotation campaign, the design of the annotation guide took six months. This preparatory work was costly (especially as it involved several researchers), even if the resulting guide has been used in a second annotation campaign, as well as for another French project (ETAPE).

However, writing an annotation guide is not a one-shot task performed at the beginning of a campaign, with only a couple of modifications added afterwards. On the opposite, the guide evolves during a large part of the annotation campaign. It is the necessary condition for its usability as the accompanying documentation for the resulting annotated corpus.

However, a first version of the guide should be written rapidly, before the campaign starts, in collaboration with the client. It is then tested by annotating a mini-reference. Usually, this generates a first round of modifications. Then, the break in phase allows, thanks to feedback from the annotators, to continue improving the document. In turn, these modifications should allow for a better quality of the annotation and for a gain in time, since the ill-defined or ill-understood categories and rules generate a waste of time for the annotators. Several cycles annotation/revision of the guide can be necessary to obtain a certain stability, which is demonstrated through a constant and sufficient annotation quality.

If their underlying principles are very close, agile annotation [Voormann and Gut, 2008, Alex et al., 2010] differs from the methodology proposed in [Bonneau-Maynard et al., 2005] in that the cycles continue until the very end of the campaign (see Figure 2.1). However, it seems to us that when the annotation is stabilized in terms of annotation quality and speed, it is not necessary to go on with the process, even if other evaluations should be performed to ensure the non-regression.

Finally, ill-defined or ill-understood categories are a cause of stress and mistakes. In order to alleviate the stress and to keep a precise trace of the problems encountered during annotation, it is important to offer the annotators the possibility to add an uncertainty note when they have doubts about their decisions. This uncertainty note can take the form of typed features, which allow for an easier processing. These types of uncertainties should of course be described in the annotation guide.

We give a number of recommendations concerning the annotation guide in [Fort et al., 2009]. We briefly summarize them here:

- indicate *what* should be annotated rather than *how*;

- do not *a priori* exclude what would be doubtful or too difficult to reproduce with a NLP system;

- give the annotators a clear vision of the application in view;

- add precise definitions, justify the methodological choices and explain the underlying logics of the annotation (do not just provide examples).

Following these recommendations should empower and motivate the annotators, by giving them access to the underlying logics. This way, we allow them to evolve from a "father-son" relationship to a pair relationship [Akrich and Boullier, 1991], which influences the annotation quality and is all the more necessary if the annotators are (corpus) domain experts who have to be as autonomous as possible.

It is therefore essential not to describe everything and to let a sufficient interpretation margin to the annotators so that they can really add value to the corpus. Guidelines which are too detailed and long to consult are less useful than a condensed guide, presenting what is essential, with a few well-chosen examples and concrete tests to distinguish between the categories which are known to be ambiguous. From this point of view, the *Penn Treebank* guidelines for POS annotation are an example to follow.

In crowdsourcing, this principle is pushed to its maximum, as the annotation guide is reduced to a couple of lines on `Amazon Mechanical Turk`, or to a couple of pages for `Phrase Detectives`. In these cases, the annotation task should remain simple or the training should replace at least part of the guidelines.

The preparatory work allows to clearly define the application in view, to write a first version of the annotation guide, to explore the corpus and to identify the actors of the campaign. It can now start. It includes three main phases: the pre-campaign, during which a mini-reference is agreed upon and the annotators are trained, the annotation itself, which starts with a break in period and includes regular evaluations and updates and finalization, which consists in a correction (manual or not) of the annotated corpus, before its publication. The general organization of an annotation campaign is presented in Figure 2.4.

2.1.4   Pre-campaign

The consensus building phase is too often reduced to a couple of meetings, but it is an iterative process that involves various actors. If the pre-campaign is organized by the campaign manager, s/he is generally associated with (annotation) domain experts in building the corpus sample which will be annotated to be used as a mini-reference. S/he is also in charge of the training of the annotators, during which they will give the first feedback on the campaign (organization, tools, guidelines).

Building the Mini-reference

Building a mini-reference from the very beginning of the campaign (see Figure 2.4) presents numerous advantages. First, it allows to test in real conditions the first version of the annotation guide, written by the manager, sometimes in collaboration with the client. Building the mini-reference also allows to evaluate the reliability of the annotation very early in the campaign. The result of this evaluation will be compared to others, later in the campaign. Moreover, once it is finalized, the mini-reference contains all the information needed to compute the complexity dimensions of the campaign (see Section 2.2), that will give precise indications to select the most appropriate tools for the campaign, be they

annotation tools (see Section 2.3), pre-annotation tools or methodological solutions (for example adding elements to the guidelines). This step also allows to select the most appropriate inter-annotator agreement metric (see Section 2.4).

The reference sub-corpus (or mini-reference) is a sample from the original "raw" corpus, if possible representative. The preparatory work (see Section 2.1.3) allowed to establish a detailed typology of the corpus and the creation of a representative sub-corpus for the mini-reference can be done by selecting files (or parts of files) corresponding to each identified type, in a proportionate way. Our goal here is not to be perfectly representative (which is an illusion anyway), but to cover enough phenomena to deal with a maximum of issues during the annotation of the mini-reference.

The size of this sub-corpus mostly depends on the time available for this annotation, but a corpus that is too small or an insufficient representativeness can lead to important errors in the computation of the complexity dimensions of the campaign. For example, we noticed when we computed the complexity dimensions for the structured named entity annotation campaign, that the selected sample was too small. The theoretical ambiguity is relatively limited on the mini-reference (around 0.15) and much higher on the global corpus (around 0.4). These results are detailed in [Fort et al., 2012b].

This mini-reference is annotated by the campaign manager (or by an expert, if the domain of the corpus is unknown to the manager), with at least one or more experts. The annotation phase is punctuated by informal meetings during which modifications of the tagset and of the guidelines are decided upon. Collective solutions are found to disagreements (by consensus). We created mini-references for two annotation campaigns (football and structured named entities) and in both cases they were finalized late in the campaign, but were used for the evaluation.

In crowdsourcing annotation, such mini-references are quite common, and are used to validate the work of the participants. For example, in `Phrase Detectives` [Chamberlain et al., 2008] and `ZombiLingo` [Fort et al., 2014b], a reference corpus annotated by experts of the task is used for the training and evaluation of the players.

It has to be noted that building a mini-reference represents a "mini-campaign" inside the campaign. Consequently, the steps described in Sections 2.1.5 and 2.1.6 also apply to the mini-reference. However, in practice, the break in period and the publication are not needed.

Training the Annotators

The training of the annotators is now recognized as essential to the quality of the annotation (see, among others [Dandapat et al., 2009, Bayerl and Paul, 2011]) and should be taken into account in the annotation campaign.

Usually, the annotators are trained for the task, i.e. both on the annotation itself and on the tool used for it. However, the two trainings present different types of difficulties. For annotators who are very competent in their domain but not at ease with computers, it is important to find the most appropriate tool, even if it means being a little less efficient (for example, a point-and-click tool like `Glozz`). Note that getting familiar with the tool can take more time than expected for these annotators. The training phase can also be used to detect annotators who are unable to perform the task correctly and to exclude them.

The training is done on an extract from the mini-reference, which has to be annotated by the annotators using the annotation tool and according to the provided guidelines. If possible, a first collective training session, with all the annotators, is more profitable than distant training, as they can ask all the questions they want and get all the answers at once.

This first (collective) phase should be followed by another phase during which the annotators work in real conditions and in parallel, without consulting each other, on the same sub-corpus, tracking their time. This tracked time will be used to visualize the learning curve of the annotators, like we did with ours on the *Penn Treebank* (See Figure 2.3). This curve is the first indicator of the level of training of the annotators. The second indicator is the produced quality.
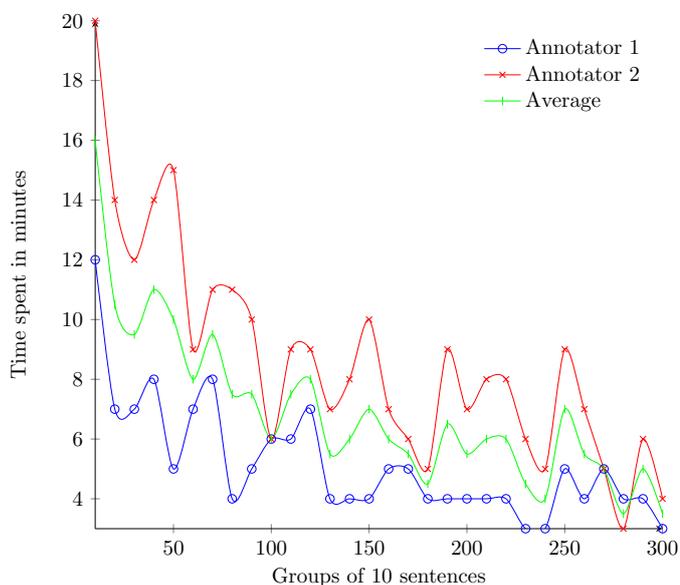
Figure 2.3: Learning curve for the POS annotation of the *Penn Treebank* [Fort and Sagot, 2010].

The evaluation of the training can be done on the mini-reference (accuracy or F-measure) or between annotators (inter-annotator agreement). A discussion should be organized with the annotators to explain the difficult points (the ones on which they disagree the most between themselves or with the reference).

The training phase can expose errors or imprecisions in the annotation guide and thus lead to modifications of the guidelines and of the mini-reference.

In games like `Phrase Detectives` or `ZombiLingo`, the training phase is automatized (indications are provided to the players to help them train themselves during the tutorial phase) and ends only when the player performs sufficiently well (less than 50% errors on `Phrase Detectives` for example).

On the opposite, in microworking platforms, the annotators can at best be submitted to a competency test before starting to work, but no training phase is planned in the system.

We will see in the next chapter that training and crowdsourcing are not contradictory, but to associate them questions what some consider as one of the

fundamental principles of the system: the participation of "non-experts". Is training "non-experts" not the same as transforming them into experts, at least of the task?

2.1.5 Annotation

Breaking-in

The end of the pre-campaign does not immediately correspond to a definitive stabilization of the campaign. First, the training of the annotators will continue, since they rarely reach the maximum of their possibilities at the end of the pre-campaign (for the POS annotation of the *Penn Treebank*, the learning period lasted one month). Second, the annotation guide will be modified again, according to the annotators' remarks. Therefore, their annotations will possibly have to be corrected.

A more or less long break in period thus succeed to the pre-campaign. Depending on the available means, the manager will continue to modify the guide more or less late in the campaign. The ideal would be to be able to review it until the very end of the campaign, in order to take into account all the elements discovered in the corpus. In practice, the guide needs to be stabilized so that the annotators can progress in the annotation and do not spend too much time correcting what they have already annotated. A good moment for that is probably when they reach their cruising speed (it can be detected easily like on Figure 2.3).

This break in phase also exists in crowdsourcing. The design of the game or of the microworking task requires several trials before the instructions (the minimal annotation guide), the interfaces (annotation tools) and the conditions of the annotation (for example with or without time limitation). An example of these iterations is presented in [Hong and Baker, 2011].

Annotating

The vast majority of the work will be carried out during the annotation phase, by the annotators. The preceding steps allowed to prepare it, but it is still important that the annotators be monitored on a regular basis, by the expert or the manager.

Inter-annotator agreement metrics should be computed regularly, to check that the annotation is reliable (see Figure 2.4). This implies that at least partial parallelization is planned. Depending on the available time, the annotation can be performed totally in parallel, by at least two annotators, but most of the time only parts of it will be. In crowdsourcing, however, it is quite common to have the participants annotate all the corpus in parallel.

The annotation phase itself can include an automatic pre-annotation step. In this case, the work of the annotators is limited to correcting this existing annotation and to complete it if necessary. We carried on a very systematic study on pre-annotation with Benoît Sagot in [Fort and Sagot, 2010], which showed that, at least on English POS annotation, there is a bias due to the pre-annotation (attention slips from the annotators, who rely too much on the pre-annotation). However, the observed gains are such (twice the speed of annotation, even with a low accuracy tool) that it is worth warning the annotators of the dangers in
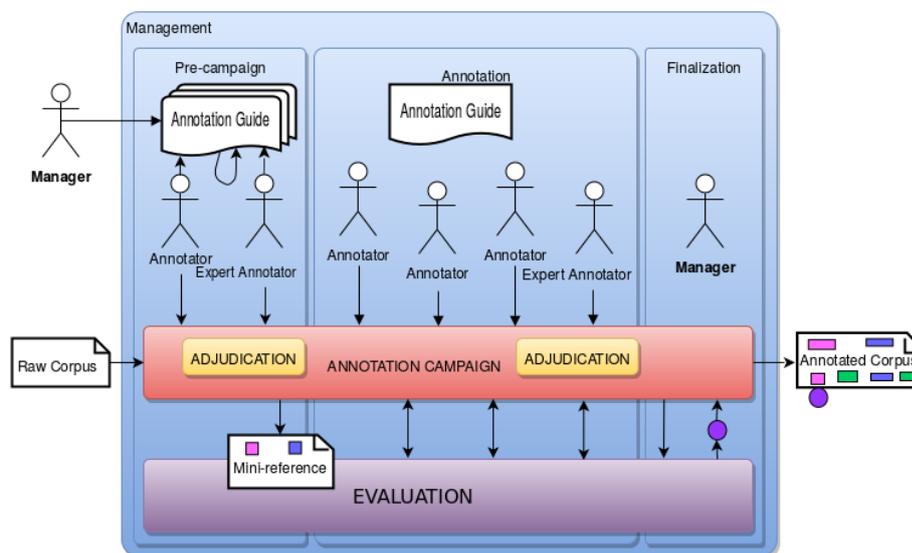
Figure 2.4: The annotation process, revisited (simplified representation).

the guidelines. The task itself is not fundamentally different, so we will use the term annotation in those cases too.

Each annotator in the campaign is assigned some files to annotate. They are provided with the up-to-date annotation guide, which is coherent with the used data model, and an appropriate annotation tool. They have been trained for the task and they have assimilated the principles explained in the guide. The break in period should have helped them refining their understanding of the task.

Ideally, the guidelines should be directly integrated in the annotation tool and the tool should be able to check the conformity of the annotation with regards to the guidelines, but if it was possible, human annotators would no longer be needed. However, intermediary features exist, which allow for a more efficient usage of the guidelines. A first one consists in providing an easy access to the guidelines, using for example a hypertext link from the tool. Another one would be to have the tool apply constraints which are defined in the guidelines (this was done with `EasyRef` in the EASy campaign, see Appendix A.2.2). The minimum is to ensure that the guidelines and the data model used in the tool are consistent (`Slate` includes for example a general versioning of the annotation, see Appendix A.4.1).

The annotation tool used by the annotators should help them not only in annotating, but also in monitoring their progression on the files which were assigned to them, in tracking the time spent on each file (or on each annotation level) and in notifying the expert or the manager of problems. It should also provide some advanced searching features (in the categories and in the text), so that the annotators can efficiently correct their annotations.

During the annotation phase, a regular evaluation of the conformity of the annotation with regards to the mini-reference should be done, associated with regular intra and inter-annotator agreement measurements.

Updating

Even if it has been decided to stabilize the guidelines at the end of the break in phase, updates are inevitable during the pre-campaign and the break in phase. These updates have to be passed on to the annotated corpus, in order for it to remain coherent with the guidelines.

During the pre-campaign, updates are decided informally, between experts. The mini-reference being small by definition, the corrections can be made immediately.

During the break in period, updates are either formally decided upon by the manager, following disappointing evaluations or less formally by the annotators, who ask the expert(s) or the manager to modify the guidelines. The manager can decide to give on some of them for reasons of cost.

2.1.6 Finalization

Once the corpus is annotated, the manager has to finalize the campaign. S/he has at her or his disposal the annotations added to the corpus and a series of indicators, including at least evaluation metrics (conformity and intra and inter-annotator agreement results), and sometimes uncertainty features added by the annotators. The manager can run a quick questionnaire among the annotators to try and catch their impressions concerning the campaign. Then s/he has to decide what to do next. Four options are available:

1. publish the corpus, which is considered to be in a sufficiently satisfactory state to be final,

2. review the corpus by adapting the annotation guide,

3. adjudicate the corpus,

4. give up on revision and publication (failure).

In most cases, a correction phase is necessary. If the annotation was carried out totally in parallel by at least two annotators, this correction can correspond to an adjudication by an expert, but it is most of the time performed more or less automatically, using the indicators provided during the campaign.[7]

In case there is a correction (adjudication and reviewing), the corpus has to be evaluated and be submitted, with its indicators, to the decision of the manager, who can either publish the corpus or have it corrected again.

---

[7]It has to be noted that corrections in the annotated corpus can also be done during the annotation phase itself. Therefore, it is important that the annotation tool provide a powerful search tool.

Failure

A complete failure, which would be noticed at the end of the campaign (during finalization) is a sign of an absence of management and remains rare. However, we witnessed such a case of failure in the campaign described in [Fort et al., 2009]. It was due to a series of causes, the main one being that there was no real manager in charge of the campaign.

Adjudication

The adjudication is the correction, by one or more expert(s) of the annotations added by the annotators. This correction is usually limited to the disagreements between annotators (hence the name), but we extend here its definition to the correction y an expert of all the annotations (a rare case in traditional annotation). In the first case, the expert validates (or not) one of the concurrent annotations. The annotations have therefore to be sorted prior to the adjudication, so that the expert only decides on disagreements. The expert can also be called for punctually, to decide on a case that is particularly difficult to annotate.

In all cases, the work of the expert can be facilitated using a tool, for example an adapted interface showing in parallel the conflicting annotations.

Interestingly, microworking a la `Amazon Mechanical Turk` does not exempt from manual correction. For example, in [Kaisser and Lowe, 2008], PhD students were hired to validate the questions/answers corpus. In `Phrase Detectives` the corrections are made by the players themselves, who judge annotations added by others.

Reviewing

In most annotation campaigns, the available resources are not sufficient to manually correct the entire annotated corpus. The correction is therefore more or less automatized, from the indicators gathered during the annotation phase. When errors are consistent, they can be corrected globally on the whole corpus, without the need for an expert.

The manager (associated with an expert, if s/he is not one) can decide to merge two categories that are too ambiguous. The annotation then needs to be modified. S/he can also remove one or two categories if their annotation was problematic. Finally, s/he can decide not to take into account the annotations from a specific annotator, if they diverge too much (in particular in crowdsourcing).

Semi-automatic correction procedures were used in the structured named entity campaign in old press. These corrections were identified thanks to a manual analysis of the errors carried on on a sample of the annotated corpus.

Publication

It is essential that the quality of the reviewed annotated corpus (or final corpus) is evaluated. In the case of a correction through adjudication of the disagreements, an evaluation performed by an expert of a random sample of uncorrected elements can be sufficient to evaluate the quality of the final corpus. In the (rare)

case of a total correction of the annotated corpus, such a final evaluation is not needed, but can be carried out by a different expert on a sample of the corpus.

This final evaluation can be used as a seal of approval of the annotated corpus and can be taken into account during the evaluation of systems trained with this corpus. The corpus is published with its up-to-date annotation guide, if possible with a version number.

In all cases, the indicators provided with the annotated corpus are crucial to the manager. However, they are often underused.

2.2 Annotation Complexity

What is complex? What should we automatize? In which case? An important step in the preparation of an annotation campaign is to identify the complexity dimensions of the annotation task at hand, as it allows to better plan the annotation work and to put the right tools at the right place. However, this is far from being trivial, as everything seems entangled like in a wood ball.

We worked on the subject with Adeline Nazarenko (LIPN/University of Paris 13) and Sophie Rosset (LIMSI-CNRS), using the various annotation projects in which we participated as a basis for our analysis. We identified and tested six complexity dimensions which we believe to be universal to all annotation tasks and we presented them in [Fort et al., 2012b]. We provide here what we hope to be a more pedagogical (and a bit simplified) view on these complexity dimensions, trying to improve their presentation by taking into account the feedback we got on the main article.

The six complexity dimensions we will describe here are all independent of each other, except for one (the context). Identifying them for a specific annotation campaign means un-entangling the wood ball. It may seem a little confusing at first, because we are not used to considering the complexity of a task as independent dimensions. However, this mental effort is essential to the deep understanding of an annotation task and we observed that the complexity grid we propose represents a very useful guide to changing perspectives on a campaign.

In order to be able to visualize the result globally without reforming another wood ball, we decided to associate metrics to each dimension, which, once computed, give results between 0 (null complexity) and 1 (maximum complexity). Some of these metrics can be computed *a priori* (without any annotation done yet) while others require an annotation sample or annotations from a similar campaign. Note that these metrics are independent from the volume to annotate and the number of annotators. An example of vizualization is shown in Figure 2.5 using a spiderweb diagram. The blue lines correspond to the dimensions linked to the identification of the segment to annotate, the three red lines to the dimensions related to the added note and the green one is the context. Instantiated examples will be given later on.

2.2.1 Examples Overview

First, let us have a look at examples of annotation in NLP. We take three, in which we participated either as annotators (the *Penn Treebank* part-of-speech
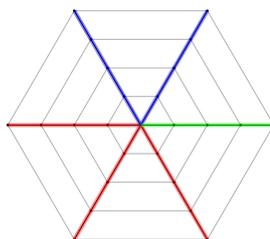
Figure 2.5: Visualization of the complexity dimensions of an annotation task.

annotation[8], and the structured named entities annotation) or as campaign manager (the gene renaming campaign). We believe they correspond to a large enough variety of situations to illustrate the complexity dimensions presentation we are going to make.

Example 1: POS

In the *Penn Treebank* part-of-speech (POS) annotation campaign, the corpus was pre-annotated and the annotators had to correct the provided annotations. As can be seen on Figure 2.6, the annotations were added in-line (inserted in the text itself),[9] separated from the original text by a simple marker (a slash), in a simple text editor. Like in any POS annotation campaign, all the lexical units[10] were annotated.

> I/PRP do/VBP n't/RB feel/VB very/RB ferocious/JJ ./.

Figure 2.6: POS annotation in the *Penn Treebank* [Marcus et al., 1993].

Example 2: Gene Renaming

Gene renaming annotation, on the other hand, implied to annotate very few segments in the whole corpus (in average one renaming per file). The annotators had to identify the gene names involved in a renaming relation and to annotate the former name of the gene and its new name (see Figure 2.7). Due to constraints imposed by the annotation tool, `Cadixe` [Alphonse et al., 2004], which was already in use when we join the project, the annotators could not annotate the relation as such. The annotations in XML therefore included an identifier (<Former id="1">, <New id="1">), were added in-line and rendered

---

[8]Obviously we did not participate in the original campaign, but we re-annotated part of the corpus for the experiments we led for [Fort and Sagot, 2010].
[9]We put them in blue here for easier reading.
[10]In the *Penn Treebank*, these were tokens.

in corresponding colors (one per renaming relation) by the tool. The corpus was not pre-annotated.[11]

> The yppB:cat and ypbC:cat null alleles rendered cells sensitive to DNA-damaging agents, impaired plasmid transformation (25- and 100-fold), and moderately affected chromosomal transformation when present in an otherwise Rec+ B. subtilis strain. The yppB gene complemented the defect of the recG40 strain. yppB and ypbC and their respective null alleles were termed recU and "recU1" (recU:cat) and recS and "recS1" (recS:cat), respectively. The recU and recS mutations were introduced into rec-deficient strains representative of the alpha (recF), beta (addA5 addB72), gamma (recH342), and epsilon (recG40) epistatic groups.

Figure 2.7: Gene renaming annotation [Jourde et al., 2011].

Example 3: Structured Names Entities

Finally, in the structured named entities annotation campaign, the work was done from scratch (no pre-annotation) on an advanced text editor (`XEmacs`) with a specific plugin allowing the annotators to select the appropriate tags step by step following the structure of the tagset. For example, if the annotator selected the segment "Lionel" and the tag *pers*, the tool then proposed him or her with the subtypes *ind* or *coll* (see Figure 2.8 for a theoretical illustration of the annotation). Obviously, not all the text was annotated, however, it represented a much larger proportion than in the case of gene renaming. Note that the corpus was this time in French and some of it was transcribed speech (broadcast news) [Rosset et al., 2012].
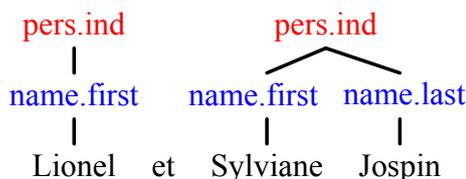
Figure 2.8: Structured named entities annotation [Grouin et al., 2011].

These three annotation campaigns are so different that it seems difficult to compare them in terms of complexity. We will see that the complexity dimensions allow for that too.

2.2.2 What to Annotate?

The first logical step in manual annotation is to identify the segment of signal to annotate. This "identification" consists in fact in two movements: (i) extracting,

---

[11]We made some tests and pre-annotation did not really help as most gene names were not in a renaming relation, thus generating a lot of noise, and some could not be found by our pre-annotation tool (silence).

in a rather gross way, a piece to annotate from the signal (discrimination) and (ii) delimiting the precise boundaries of this segment.

Discrimination

If the second step is easy to grasp for most people, as delimitation builds on existing metrics like the word error rate (a well-known metric of the performance of speech recognition systems for example), the first step, the discrimination phase, is usually more difficult to understand and is often overlooked. It captures the "needle in a haystack" effect, i.e. the fact that the segment to annotate is more or less easy to find in the source signal.

Let us consider examples 1 and 2. In POS annotation (example 1), all the tokens are to be annotated, there is nothing to search for (especially as the corpus was pre-annotated), so the discrimination will be null (0). On the opposite, in the gene renaming annotation case (example 2), the segments to annotate are scattered in the corpus and rare (one renaming per text in average), so the discrimination will be very high (close to 1).

When the segments to annotate are lost in the crowd of the text, i.e. when the proportion of what is to be annotated as compared to what could be annotated (resulting from the default segmentation, often token by token) is low, the complexity due to the discrimination effort is high. This is expressed in the following way:

Definition 2

$$Discrimination_a(F) = 1 - \frac{|A_a(F)|}{|D_i(F)|}$$

where F is the flow of data to annotate, a is an annotation task, $|D_i(F)|$ is the number of units obtained during the segmentation of F at level i and $|A_a(F)|$ is the number of units to be annotated in the relevant annotation task.

Applying this metric, we obtain a discrimination of 0 for POS annotation and 0.95 for gene renaming.

Delimitation

Once the units coarsely identified, they have to be finely delimited. This is the delimitation process.

The definition of the delimitation metric is inspired by the slot error rate (an adaptation of the word error rate) [Makhoul et al., 1999]:

Definition 3

$$Delimitation_a(F) = min(\frac{S + I + D}{|A_a(F)|}, 1)$$

where $|A_a(F)|$ is the final number of discriminated units, I the number of inserted units, obtained by initial units decomposition, D the number of units deleted when grouping some of the initial units, and S is the number of substitutions, i.e. *the number of discriminated units that underwent a change in their boundaries other than that of the previous decomposition and grouping cases.*

The delimitation complexity dimension is null in the case of gene renaming, as gene names were simple tokens. It reaches the maximum (1) for the structured named entities task, as many frontiers changes had to be performed by the annotators from a basic segmentation in tokens.

The computation of both the discrimination and the delimitation complexity dimensions requires at least a sample of annotation, either from the campaign being prepared or from a previous, similar campaign.

2.2.3 How to Annotate?

Once precisely identified, the units have to be characterized by the annotators. To do so, they rely on an annotation language with a certain expressiveness, instantiated in a tagset of a certain dimension.

Expressiveness of the Annotation Language

To evaluate the complexity due to the expressiveness of the annotation language, we decided to rely on an arbitrary (but logical) scale, graduated from 0.25 (type language) to 1 (higher order languages). Relational languages of arity 2 are attributed 0.5 in complexity and 0.75 is associated to relational languages with arity higher than 2.

In the simplest and most frequent case, the annotation language is a type language: annotating consists in associating a type to a segment of data. A lot of annotation tasks use this category of language: POS, speech turns, named entities, etc. The number of tags can vary, but this does not change the expressiveness of the language.

Establishing relations between units has become a relatively common task, but it is more complex. It requires to connect different segments of data, which are often typed. The relations are often typed too and they can be oriented. This is for example the case for dependency syntax relations or gene renaming annotation.

In general, the relations are binary, but sometimes relations of arity above two are necessary, for example in information extraction: who bought what? when? to whom? at which price? In such cases, the annotation task is much more complex: the annotators have to discriminate, delimit and categorize the arguments of the relation, then to identify the couples, triplets, n-uplets of segments to annotate and finally, to label the relation.

Higher order languages are used when annotations are added to annotations, for example to qualify an annotation as uncertain. However, the complexity of this type of language is such that, in most cases, the problem is avoided by increasing the dimension of the tagset (creating a new feature associated to the main types).

Most annotation tasks correspond to a complexity of 0.25 or 0.5 for this dimension. In our examples, the POS and structured named entities annotation tasks are performed using simple type languages, so they reach a complexity of 0.25. Interestingly, the gene renaming campaign, that should correspond to 0.5, as it is a relation, reaches only 0.25 in complexity, due to the fact that the annotation tool did not allow for the annotation of real relations. Although it simplified this complexity dimension, it made the tagset more complex to use.

Tagset Dimension

The size of the tagset is probably the most obvious complexity dimension. It relates to short-term memory limitations and is quite obvious when you annotate. However, a very large number of tags is not necessarily a synonym for maximum complexity: if they are well-structured, like in the structured named entities annotation task (31 types and sub-types), then the annotators have to make choices from a reasonable number of tags each time, at different levels. In the structured named entities case, they first have to choose between 7 main types (*Person, Function, Location, Production, Organization, Time, Amount*), which corresponds to a degree of freedom of 6. Then, in the worst case (if they selected *Production*), they have to choose between 9 sub-types (see Figure 2.9), i.e. a degree of freedom of 8. Finally, sub-subtypes are available in some cases like *Location* and *Time*, so there can be a choice to make from a maximum of 4 tags, which corresponds to a degree of freedom of 3. We propose to use these degrees of freedom to compute the tagset dimension complexity, in order to take into account the fact that tagset constraints relieve the annotators from part of the categorizing effort.

Person		Function		
pers.ind (individual person)	*pers.coll* (group of persons)	*func.ind* (individual function)	*func.coll* (collectivity of functions)	
Location			**Production**	
administrative (*loc.adm.town, loc.adm.reg, loc.adm.nat, loc.adm.sup*)	physical (*loc.phys.geo, loc.phys.hydro, loc.phys.astro*)	facilities (*loc.fac*), oronyms (*loc.oro*), address (*loc.add.phys, loc.add.elec*)	*prod.object* (manufactured object) *prod.serv* (transportation route) *prod.fin* (financial products)	
			prod.doctr (doctrine) *prod.rule* (law) *prod.soft* (software)	
			prod.art *prod.media* *prod.award*	
Organization		**Time**		
org.adm (administration)	*org.ent* (services)	*time.date.abs* (absolute date), *time.date.rel* (relative date)	*time.hour.abs* (absolute hour), *time.hour.rel* (relative hour)	
Amount				
amount (with unit or general object), including duration				

Figure 2.9: The tagset dimension: taking the structure into account in the structured named entities annotation task [Grouin et al., 2011].

The total degree of freedom ν for the choice of m labels is given by the following formula:

$$\nu \leq \nu_1 + \nu_2 + \ldots + \nu_m$$

where ν_i is the maximal degree of freedom the annotator has when choosing the ith tag ($\nu_i = n_i - 1$).

The tagset dimension can then be computed using the following formula:

$$Dimension_a(F) = min(\frac{\nu}{\tau}, 1)$$

where ν is the global degree of freedom the annotator has when choosing a tag for an annotation task a within a flow of data F, and τ is the threshold from which we consider the tagset as arbitrarily large. In the experiments detailed below, τ is worth 50, based on the feedback of the annotators.

Using these formulas, the tagset dimension complexity of the structured named entity annotation task reaches 0.34, which is quite low as compared to the 0.62 we would obtain without taking the structure into account.[12] As for the gene renaming annotation task, it involved only two tags, *Former* and *New*, but to bypass the annotation tool constraints, an identifier had to be added to disambiguate between the renaming couples that were present in the same text. As there were no more than ten renaming relations per text, this represents around 10 "subtypes", i.e. ν is close to 10 and the tagset dimension reaches 0.2, which is no so far from the result for structured named entities.

Both the expressiveness of the language and the tagset dimension can be computed *a priori*, without any annotation done yet.

Degree of Ambiguity

Disambiguating the units to annotate is the heart of the work of the annotators. This is is obviously a complexity dimension and this is where most of the interpretation lies, but it is very difficult to evaluate precisely. However, we propose two ways of approximating it.

Residual Ambiguity First, we can observe the traces left by the annotators when they are given the opportunity and the possibility to do so. For example, in a gene and protein names annotation campaign [Fort et al., 2009], we gave the annotators the possibility to add an uncertainty feature to the annotation (see Figure 2.10). Although only of them used this possibility, it is quite useful to evaluate the ambiguities they faced.

> [...] <EukVirus>3CDproM< /EukVirus> can process both structural and nonstructural precursors of the <EukVirus **uncertainty-type** = "too-generic"><taxon>poliovirus< /taxon> polyprotein< /EukVirus> [...].

Figure 2.10: Example of typed trace left by the annotator when annotating gene and protein names [Fort et al., 2009].

We call this the *residual ambiguity* and we define it in a very simple way:

Definition 4

$$Ambiguity_{Res,a}(F) = \frac{|Annot_A|}{|Annot|}$$

where a and F are the annotation task and the flow of data to be considered and where $|Annot_A|$ and $|Annot|$ are respectively the number of annotations bearing an ambiguity mark and the total number of annotations added to F.

By definition, the residual ambiguity can only be computed from an annotation sample, if the possibility to add traces was given to the annotators. In the gene renaming campaign, it was nearly null (0.02), probably due to the fact that, again, only one annotator added traces. Therefore, this metric is not completely reliable and it should be associated to another one whenever possible.

---

[12]These results differ from the ones presented in [Fort et al., 2012b] because we simplified the example (the annotation task also included components).

Theoretical Ambiguity   The second way to measure the complexity of the disambiguation process is to measure the degree of *theoretical ambiguity* for the tasks where several occurrences of the same vocable are annotated. This applies to POS annotation or semantic disambiguation, but not to gene renaming annotation.

This metric relies on the idea that ambiguous vocables are annotated with different tags in different places in the text (or flow of data). We then need to compute the proportion of the units to annotate which correspond to ambiguous vocables, taking into account their frequency. This can be done using the following formula:

Definition 5

$$Ambiguity_{Th,a}(F) = \frac{\sum_{i=1}^{|Voc(F)|} \left(Ambig_a(i) * freq(i, F) \right)}{|Units_a(F)|}$$

with

$$Ambig_a(i) = \left\{ \begin{array}{lll} 1 & if & |Labels_a(i)| > 1 \\ 0 & else \end{array} \right.$$

where Voc is the vocabulary of the units of the flow of data F, $|Voc(F)|$ the size of the vocabulary, $freq(i, F)$ the frequency of the vocable i in F, $|Units_a(F)|$ the number of units to annotate in F and $|Labels_a(i)|$ the number of tags available for the vocable i for the annotation task a.

Again, to compute this metric, we need an annotation sample or results from a similar task.

2.2.4   The Weight of the Context

The context to take into account during annotation is an obvious complexity factor. However, this dimension is not independent from all the above-mentioned dimensions. It influences directly the discrimination, delimitation and disambiguation processes, as the larger the context, the more difficult it gets to identify the units to annotate and to disambiguate them. Nonetheless, we decided not to include it as a modifying factor of these three dimensions, first to keep them simpler, and second because of its strong identity.

In NLP, the context is traditionally the co-text taken into account by the systems. Despite some evolutions (in particular in discourse annotation), the sentence is still the favored processing unit in our domain. However, for the annotators, the context is not only the text they have to read to be able to annotate (identify and characterize) properly, but also the knowledge sources they need to consult. These sources usually include the annotation guidelines, but they may also be external sources, either identified during the campaign preparation, like nomenclatures a la `SwissProt`[13], or to be found by the annotators themselves, on the Web or elsewhere.

Obviously, the more accessible and predictable the source, the less complex it is for the annotators to get the piece of information they need. As for the co-text, the larger the context to take into account, the more complex it is to annotate (see Figure 2.11).

---

[13]See: `http://www.uniprot.org/`.
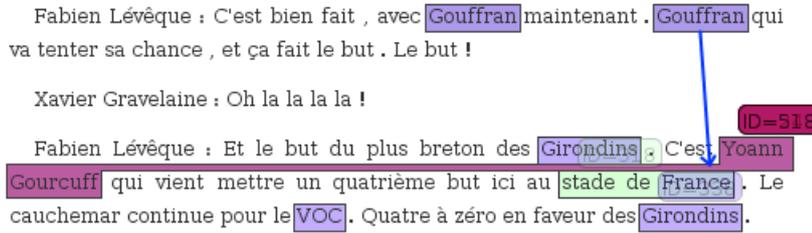
Figure 2.11: Example of annotation of a goal in football annotation [Fort and Claveau, 2012]: a context of more than the sentence is needed.

We therefore designed a common discreet scale including both these sub-dimensions. In this scale, 0 corresponds to an impossible case, where there is no need for an annotation guide and no co-text to take into account. This should never happen, as the consensus has to be somehow transmitted to the annotators. 0.25 corresponds to a case where an annotation guide is needed OR the immediate co-text is needed to annotate. Logically, the complexity reaches 0.5 when the OR of the previous description changes to an AND, i.e. when the annotators need guidelines AND a small context to annotate. Another case in which we reach 0.5 is when a larger part of the data (like the sentence) OR an identified external source of knowledge is needed. 0.75 corresponds to the case when the annotators need to read a larger co-text AND have to consult an identified external source to annotate. It also covers the cases in which the annotators have to access unpredicted sources of knowledge OR have to read the whole text to be able to annotate. Finally, 1 is for cases where the annotators both have to consult previously unidentified sources of knowledge AND the whole data flow (usually, text).

The gene renaming task is very complex from that point of view (1), as it required the annotators to read the whole text and they sometimes needed to consult new external sources. POS annotation would be close to 0.5, as most of the time only the guidelines and a small co-text are needed to annotate.

2.2.5 Visualization

Once the 6 complexity dimensions computed, it is rather easy to put them in a spiderweb diagram to visualize the complexity profile of the annotation task. This type of representation can prove useful to compare the complexity of different tasks. Figures 2.13 and 2.14 present examples of what can be obtained applying the complexity grid, even in a fuzzy way (for the POS annotation task).

In the *Penn Treebank* POS annotation task, the corpus was pre-segmented and pre-annotated, so the discrimination and delimitation are null. The annotation language is a type language. The tagset contains 36 tags [Santorini, 1990], so ν equals 35, but if we consider that there is an implicit structure in the tagset, with *JJR* and *JJS* being subtypes of *JJ*, then $\nu = 20 + 5 = 25$ and the
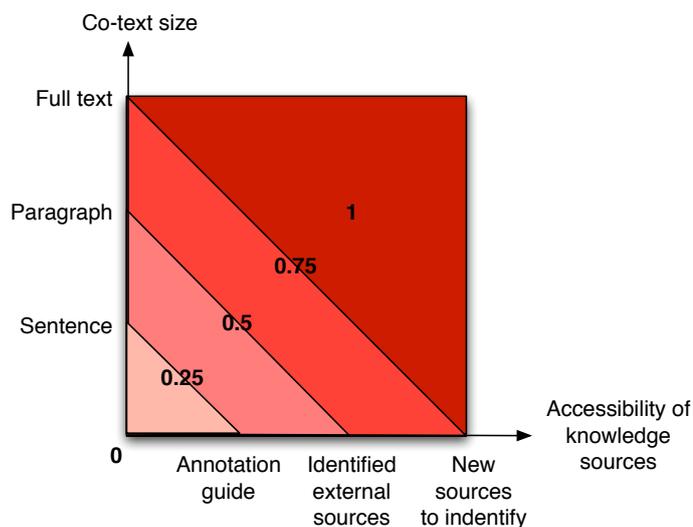
Figure 2.12: The context as a complexity dimension: two sub-dimensions to take into account.

complexity dimension of the tagset is 0.5. The annotation guidelines allowed for the usage of an ambiguity mark (a vertical slash, ”|”) in case of true ambiguities, so even if this is not exactly residual ambiguity, it can still be computed. However, for the *Wall Street Journal* part of the corpus, it represents only one case, so it probably can be considered as null over the whole corpus. As for the theoretical ambiguity, Dan Jurafsky and James H. Martin, in the new edition of their well-known book [Jurafsky and Martin, 2009] evaluate the ambiguity in
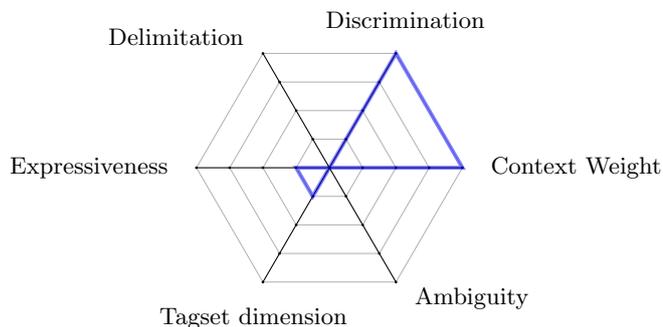
Figure 2.13: Instantiated visualization: the delimited surface represents the complexity profile of the annotation task, here, gene renaming.

POS for English saying that "[...] the ambiguous words, although accounting for only 14-15% of the vocabulary, are some of the most common words of English, and hence 55-67% of word tokens in running text are ambiguous".[14] This implies that the theoretical ambiguity is rather high and without even computing it precisely, we can evaluate it at 0.5. The context to take into account is restricted to an annotation guide et a limited co-text (0.5).
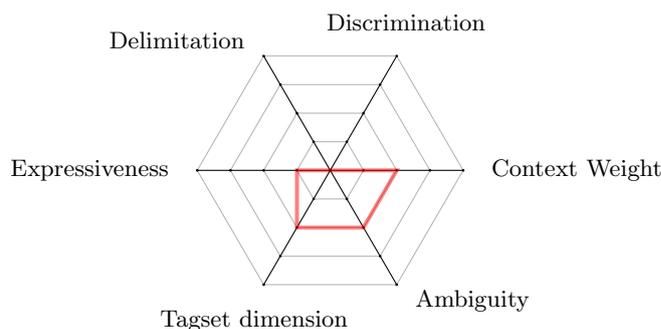
Figure 2.14: Instantiated visualization: POS annotation in the *Penn Treebank*.

The complexity profiles of these annotation tasks are very different, thus reflecting the need for very different solutions to limit the complexity of the tasks. For POS annotation, even without pre-annotation, the discrimination and delimitation would have been low, due to the fact that, in this campaign, only tokens were annotated. However, the tagset dimension complexity could have been reduced by structuring the tagset more (and taking this structure into account in the annotation tool). As for the gene renaming campaign, it could have benefited from an "intelligent" pre-annotation (taking into account keywords like "renamed") to reduce the discrimination effort. It could also have been easier from the context point of view if a precise list of the sources to consult were provided in the guidelines.

2.2.6 Elementary Annotation Tasks

We saw that the gene renaming annotation task can be analyzed with the complexity grid as it was performed in the campaign, with identifiers as features in the XML tags. However, it should probably have been annotated differently, with a more suitable tool and with real relations. In this case, it would have been difficult to analyze it as a whole.

We propose to decompose such tasks into Elementary Annotation Tasks (EATs) and to compute the complexity of the various EATs independently, the global complexity of the task being a combination of the local EATs' complexity. Note that EATs do not necessarily correspond to annotation levels or layers [Goecke et al., 2010] or to the practical organization of the work.

Definition 6 *An Elementary Annotation Task (EAT) is a task that cannot be decomposed. We consider that an annotation task can be decomposed into at least two EATs if its tagset can be decomposed into independent reduced tagsets.*

---

[14]See the draft here: `https://web.stanford.edu/~jurafsky/slp3/9.pdf`.

Tagsets are independent when their tags are globally compatible (even if some combinations are not allowed), whereas the tags from a unique tagset are mutually exclusive (apart from the need to encode ambiguity).

In the gene renaming campaign, for example, the annotation of the relations can be analyzed as a combination of two EATs: (i) identifying gene names in the source signal and (ii) indicating which of these gene names participate into a renaming relation. The two tagsets are independent and the global task is easier to analyze this way:

(i) Identifying gene names   Only few words are gene names, so the discrimination is high (0.9). Gene names, in our case, are only tokens, so delimitation is null. The tagset dimension is null too, as there are only one tag (*gene name*). We use a type language (expressiveness=0.25). The ambiguity is very low, as only few gene names are ambiguous and the annotators left little trace of uncertainty on this. On the opposite, the necessary context is relatively high (between 0.5 and 0.75) as although only a few words are needed to identify a gene name, the annotators had sometimes to consult external resources.

(ii) Annotating gene renaming relations   This EAT consists in identifying, among all the gene name couples appearing in the same text (*PubMed* abstracts), the ones that are connected by a renaming relation, i.e. the ones that are the former and the new names of one gene. As we already said, renaming relations are rare, so the discrimination for this EAT is high (0.95). Gene names are already annotated (EAT 1), so delimitation is null. The relation is oriented, but there is only one type of relation, so the tagset is close to null. The annotation language is relational (0.5) and ambiguity is very low according to the traces left by the annotators (0.02). The context is maximum (1), as the annotators had to read the whole text to be able to identify renaming relations and they at least had to consult identified external sources.

The two EATs are then combined to provide a global view on the campaign with a scale that is twice the scale for one EAT (see Figure 2.15). In this particular case the result if very close to that of the single EAT analysis (see Figure 2.13).

Note that the decomposition into EATs does not imply a simplification of the original task, as it is often the case for Human Intelligence Tasks (HITs) performed by *Turkers* (workers) on `Amazon Mechanical Turk` (see, for example [Cook and Stevenson, 2010]).

2.3   Annotation Tools

Annotation tools make manual annotation much easier, in particular when using markup languages like XML. These interfaces allow to avoid the tedious writing of tags and the associated typing errors, but their contribution reaches far beyond that.

If there are many articles detailing a specific annotation tool, only few provide a high level view on the subject. To our knowledge, only [Dipper et al., 2004], [Reidsma et al., 2004] and [Burghardt, 2012] present an in-depth comparison of
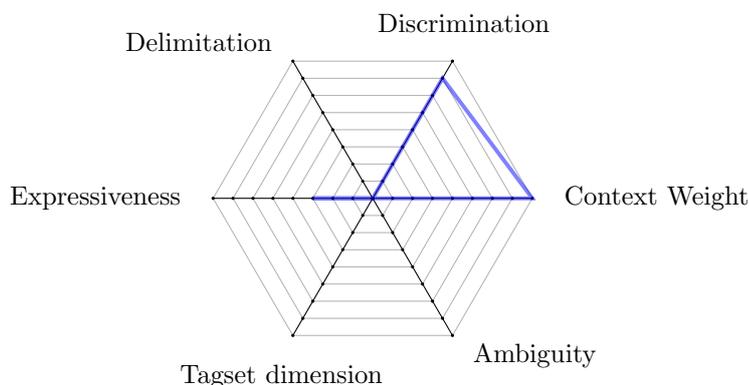
Figure 2.15: Synthesis of the complexity of the gene names renaming campaign (new scale x2).

the tools in order to allow for their evaluation. However, these articles only consider a limited number of annotation tools (five in [Dipper et al., 2004], two in [Reidsma et al., 2004] and three in [Burghardt, 2012]) and the analysis carried out in the first two is focused on a specific annotation task (purely linguistic annotation in the first one and video annotation for the second one). The present state-of-the-art uses some information from these articles, but it is mainly drawn from our own experience and analysis of the existing tools (a non exhaustive list of these tools is presented in Appendix A).

2.3.1   To be or not to be an Annotation Tool

Before going into more details about annotation tools, we need to clarify what they are:

Definition 7 *A system supporting manual annotation, or annotation tool, is an interface facilitating the manual annotation of a signal.*

Some tools support the manual annotation of non-textual corpora, like video (`Anvil`[15] or `Advene`[16]), speech (`Praat`[17]) or music (`wavesurfer`[18]), but such an inventory would take us too far. We therefore restrain our analysis to manual text annotation.

We do not present Web annotation interfaces either, but their features are usually close to that of the tools we present here, without being as complex and rich. Finally, we do not consider XML or text editors as annotation tools as such. As this can seem surprising for some, we will explain why.

A number of manual annotation campaigns use XML editors to help the annotators in their work. This was the case for example for the manual annotation of Stendhal's manuscripts [Lebarbé, 2008], which was performed using

---

[15]See: `http://www.anvil-software.de/`.
[16]See: `http://liris.cnrs.fr/advene/`.
[17]See: `http://www.fon.hum.uva.nl/praat/`.
[18]See: `http://sourceforge.net/projects/wavesurfer/`.

`Morphon`[19]. Another example is the *Definiens* project of annotation of the definitions of the French dictionary *Trésor de la Langue Française* [Barque et al., 2010], in which the annotators used `oXygen`.[20] Because we had to use XML tags, a partner imposed on us to use `Epic`[21] to annotate patents in pharmacology.

An XML editor is designed to edit XML files, not to annotate. If the features seem similar, the underlying logic is quite different: an XML editor supports the modification of an XML file, not the annotation of a corpus of texts. The first difference concerns the notion of corpus, which does not exist in XML editors. This prevents to have a global vision of the annotation. Moreover, these tools usually do not support standoff annotation, which prevents the annotation of overlaps, discontinuous groups and relations. Obviously, the management of annotation campaigns is not supported by such tools. Finally, some annotation tools (like `Knowtator`, `Glozz`, `Slate` or `GATE`) allow to visualize the disagreements between annotators and for some of them to compute the inter-annotator agreement. This is never possible with an XML editor.

Text editors present the same limitations. Besides, they usually do not provide any means to validate the XML and annotators may therefore put the XML tags in the wrong order. However, simple tools can prove very useful for limited experiments (for example in prototyping) or when they are completed by normalization scripts.

Given the multiplication of annotation campaigns, it would take months to install and test all the annotation tools which are in use today. They are more or less available, more or less maintained, some are open-source, some not. From our point of view, the lifespan of a tool depends on the same criteria as that of corpora as described in [Cohen et al., 2005]: an annotation tool, to be used on the long-term, should be freely available, maintained and well-documented. This means that to survive on the long run, annotation tools should be supported, either by a community of developers or by an institution. This kind of tool should also be easy to install and use. Ergonomics is important, as features which are difficult to access are not used by the annotators. We witnessed this in an annotation campaign in microbiology, in which the annotators often failed to report their uncertainties (which are needed to compute the residual ambiguity, see Section 2.2.3) because the corresponding feature was not easy to add.

2.3.2   Much more than Prototypes

Annotation tools are generally designed for one or several annotation tasks, rather than around the needs of the annotators. However, the tendency seems to evolve towards taking them more into account, through more user-friendly and more efficient interfaces. Besides, there is a growing consensus about the use of XML and standoff annotation, which seems to correspond to a form of standardization of the formalisms.

---

[19]This tool is now deprecated, see: `https://collab.itc.virginia.edu/wiki/toolbox/Morphon's\%20Xml-editor.html`.
[20]See: `https://www.oxygenxml.com/`.
[21]Now PTC Arbortext Editor: `http://www.ptc.com/service-lifecycle-management/arbortext/editor`.

Taking the Annotators into account

Even if there is still room for improvement (there are no keyboard shortcuts in
Glozz [Widlöcher and Mathet, 2009] and too many windows in MMAX2 [Müller and Strube, 2006],
etc.), the interfaces of the annotation tools become more and more user-friendly.
For example, they offer editing possibilities that allow the annotators to auto-
mate some tasks (*annotate all* in GATE [Cunningham et al., 2002], Glozz and
Djangology [Apostolova et al., 2010], automatically generated regular expres-
sions in SYNC3 [Petasis, 2012], rapid selection with one selection in Knowtator
[Ogren, 2006]). They also often allow to hide some annotations (by levels, or
according to other criteria, like in Glozz) to ease the visualization and almost
all of them allow for a certain level of customization (at least the colors of the
tags).

Moreover, searching and editing annotations is sometimes made easier thanks
to powerful search engines, which allow to search both the text and the annota-
tions (in GlozzQL for Glozz, using regular expressions in UAM CorpusTool [O'Donnell, 2008]).
Once the annotators trained for the task and with the annotation tool, they an-
notate more rapidly and more comfortably [Dandapat et al., 2009].

It has to be noticed that, even if it is not an equally shared preoccupation for
all the annotation tools developers, the vast majority of the available interfaces
are written in Java (characters are encoded in Unicode) and therefore support
most natural languages.

Some features, even if they are not yet widespread, are quite useful. For
example, brat [Stenetorp et al., 2011] associates to each annotation a unique
URL, which allows not only to link the data over the Web, but also to un-
ambiguously reference an annotation in the documentation. Glozz, which was
designed for discourse annotation, offers a global visualization of the text that
helps annotating macro level structures.

However, some useful features are still missing. For example, in a campaign
where old press was annotated with named entities [Rosset et al., 2012], we
noted that the digitalized text was often erroneous and that the annotators
needed to see the original image to be able to annotate correctly. For such
cases, it would be interesting to include the original scanned source as an image
into the tool. Another issue is that of meta annotations. If some tools like
Glozz allow to add commentaries or uncertainties, these have to be planned
beforehand in the data model and are not proposed by default, even though
they are essential.

Finally, most annotation tools are not robust enough and are not suitable for
the annotation of large files (this is in particular the case for Glozz and GATE).

Standardizing the Formalisms

XML has become the most widely used export and storage format for annota-
tions, associated in most tools to standoff annotation. Annotations are standoff
when they are presented separately from the source signal, often in another file.
This evolution was advocated in [Leech, 1997, Ide and Romary, 2006]. Standoff
annotation presents many advantages: it preserves the source corpus (the rights
on this corpus are respected and it cannot be polluted) and it allows to anno-
tate discontinuous groups, overlaps, inclusions and relations (oriented or not).
It is also more flexible than inline annotation since new annotation levels can be

added, without modifying the existing ones. Finally, each annotation level can be manipulated separately, including in different files, in particular to compare annotations.

However, some annotation campaign managers prefer to give the annotators the possibility to access the source data. It was in particular the case in the structured named entity campaign we participated in [Rosset et al., 2012]. In this case, the inconvenient is that an annotator may introduce errors in the file, for example by inserting XML tags in the wrong order, thus transforming it into an invalid XML file. `GATE` is the only tool we know of that offers the possibility to modify the source data as an option, leaving to the manager the choice to allow or not for such modifications.

The possibility to annotate relations (oriented or not) or sets (in the case of anaphora), is more and more commonly proposed. However, `Callisto` [Day et al., 2004] and `GATE` offer limited capabilities and `UAM CorpusTool`, `Cadixe` [Alphonse et al., 2004] and `Eulia` [Artola et al., 2004] do not support this.

Some tools allow for the definition and usage of different annotation layers (`MMAX2`, `Glozz`, `UAM CorpusTool`), corresponding to linguistic levels (POS, syntax, etc.) or to "groups", which are defined by the designer of the data model, like in `Glozz`. The flexibility of the definition of these groups allows for the grouping of semantically close elements, without them having any specific linguistic meaning (like *Player*, *Team*, *Referee* and *Coach* in football matches annotation [Fort and Claveau, 2012]). These groups can then be used to annotate (one group being annotated before the other), to customize the display (hiding or not a group) and for the inter-annotator agreement computation.

2.3.3   Addressing the new Annotation Challenges

We observe that annotation tools are progressively reaching maturity and are evolving in three main directions: genericity, collaboration and campaign management.

Towards more Flexible and more Generic Tools

We have witnessed, in the last decade, an evolution towards task-oriented annotation tools towards more generic and more flexible tools, often using plug-ins (this is the case for example in `GATE`) or a common Application Programming Interface (API) (like in the `LDC tools` [Maeda and Strassel, 2004]).

This genericity, when it results from an evolution of a task-oriented tool or of a tool with a different objective, often generates complexity issues and the final tool is often difficult to install and to parameterize. This is the case, in particular, for `GATE` and `Callisto`, which underlying logics is not so easy to understand (`GATE`, for example, was not originally designed for manual annotation). This results in a long learning curve for the campaign manager. When they are designed to be generic manual annotation tools from the very start, they are usually easier to get into and to parameterize. This is the case, for example for `Glozz` and `CCASH` [Felt et al., 2010].

Moreover, the generalization of XML allows to adapt more easily to *de facto* standards like TEI (Text Encoding Initiative, a format that is well-known in the humanities), and thus to share annotated data.

Finally, some annotation tools designed for biocuration, like `brat`, include linked data annotation capabilities, which allow to normalize the annotations using a unique identifier and to share them. This feature is very powerful and the development of linked data will accelerate its generalization. For the moment, linguistic linked data are still limited, but the development of new formats like NIF (NLP Interchange Format) [Hellmann et al., 2013] will certainly help.

Towards more Collaborative Annotation

As we saw in Section 2.1, annotating implies to reach a consensus on the definition and perimeter of the used categories. This consensus is at the heart of the annotation process and it cannot be built without collaborating. Annotation is therefore by essence collaborative. However, we hope that we show in this book that collaboration in annotation can take various forms.

Georgios Petasis uses the adjective *collaborative/distributed* in [Petasis, 2012] to distinguish between collaborative annotation tools as Web applications and his tool, which is not a thin client. By doing so, he is trying to unravel two terms that are often mixed up today. The term *collaborative annotation* is ambiguous and for some means crowdsourcing annotation, for others annotation by a community of experts, if not both (the call for paper for the Linguistic Annotation Workshop VI is a good example of that[22]), and for others, including Petasis, it means the participation to a common annotation project.

Collaboration in annotation is defined along two axes: its visibility to the annotators (do they know they are collaborating?) and the potential means used for its implementation, as collaboration can be direct or indirect. For example, the wiki-like annotation mode, in which each annotator sees what the others are doing and can modify their annotations, like in `brat` and as an option in `SYNC3`, is fully collaborative, as it is both direct and visible to the annotators.

On the opposite, the adjudication by an expert (of the field of the annotation) of annotations added by others is a form of indirect collaboration, since the expert benefits from the work previously done and is inspired by it. Beyond that, annotating in parallel parts of the corpus and using the resulting inter-annotator agreement to improve the annotation guidelines, which in turn will be used by the annotators, is another form of collaboration since the work carried out by some upstream influences the annotation to be performed by all. This type of indirect collaboration is rather invisible to the annotators, as they only see its negative manifestation (when they are told about it): their disagreements.

A more obvious form of collaboration is the possibility to interact with other annotators and to keep track of these interactions. If in `EasyRef` this interaction is indirect through bug reports [de la Clergerie, 2008], it is direct and clearly visible in `AnT&CoW` [Lortal et al., 2006][23] as a forum. Another benefit of this type of interaction is that it fosters the motivation of the annotators. Unfortunately, it is not yet offered in existing annotation tools for NLP. We plan to add such a feature to the Game With A Purpose (GWAP) `ZombiLingo` [Fort et al., 2014c], so that the annotators can both socialize and correct themselves.

Collaboration has always been part of annotation. However, we have been witnessing, since the advent of the Web 2.0, the development of new forms of

---

[22]*"The special theme for LAW VI is Collaborative Annotation (both community-based and crowd-sourced)"*: `http://faculty.washington.edu/fxia/LAWVI/cfp.html`

[23]`AnT&CoW` is not annotation tool for NLP, which is why it does not appear in the appendix A.

collaboration. We present in details, in the second part of this book, the various forms of crowdsourcing annotation, but we can summarize here the main forms of collaboration it implies. Games with a purpose like `Phrase Detectives` [Chamberlain et al., 2008] or `ZombiLingo` usually provide for an indirect and visible collaboration (through the obtained scores and the leaderboards).[24] As for the microworking platforms like `Amazon Mechanical Turk`, they only allow for a very indirect collaboration through the agreement (or disagreement) among the workers, which is invisible to them (they have very little feedback on their work).

This evolution is accompanied by a raising awareness of the importance of the annotators' training and of the evaluation of the annotation. Both `Phrase Detectives` and `ZombiLingo` put emphasis on these two points, with mandatory training and regular evaluations of the performance of the annotators. One of the objectives of collaboration is to facilitate the training of the annotators thanks to co-evaluation.

Towards the Annotation Campaign Management

To our knowledge, the first research paper explicitly mentioning the annotation campaign management is [Kaplan et al., 2010], which presents `SLATE`, a tool offering features not only to support the annotation process, but also, and this is what makes it original, a more macro vision of the annotation process, including a clear definition of its actors (*administrator* and *annotators*, considered as completely distinct). Thanks to `SLATE`, the administrator can distribute and monitor the texts to annotate and therefore manage the corpus. The corpus itself is versioned throughout the project and each annotation is identified with the version number of the project, which also corresponds to that of the tagset at the time of the annotation. `SLATE` also includes comparing and merging features.

A more formal and explicit definition of the roles can be found in `GATE Teamware` [Bontcheva et al., 2010], which identifies three of them (*campaign manager*, *editor* or *curator* and *annotator*) and in `WebAnno` [de Castilho et al., 2014] (*Users*, *curator* and *Administrators*). `Egas` distinguishes only between *Managers* and *Curators*. As for the annotation management features, they are similar in `Djangology` and `GATE Teamware` and were to be developed in `CCASH` and `Callisto` (but it does not seem to be done yet).

The evolution towards annotation management is now obvious. However, it started long before 2010. Interfaces allowing to compare annotations and to compute inter-annotator agreements were added in many tools (`Knowtator`, `MMAX2`, `Glozz`, `SYNC3`). Besides, if NLP platforms like `GATE` propose automatic processing to optimize manual annotation, most of the other tools support that such processing is applied beforehand (provided the result is adapted to the format of the tool, like in `Glozz`) and some even provide some pre-annotation, like *tag dictionary* (a unit is pre-annotated with the tags that were associated to it earlier in the corpus), in `Djangology` and `CCASH`.

Given the potential biases generated by pre-annotation [Fort and Sagot, 2010], we consider that automatic processing should be decided upon and applied by the campaign manager. It therefore falls under campaign management and not

---

[24]Some future features of `ZombiLingo` will allow for a more direct collaboration.

annotation as such. The same goes for the possibility to modify the annotation schema during the campaign (this is proposed in `UAM CorpusTool`, `GATE` and `ANALEC` [Landragin et al., 2012]).

Finally, tools like `Slate` or `EasyRef`[25] propose to define constraints on the annotation (for example, in `EasyRef`, pop up menus allowing only for the actions authorized in this context), which, again, should be defined by the manager.

The monitoring of the annotation campaign is another feature offered by many "simple" annotation tools that is directly linked to campaign management, even if it can be useful to annotators too. For example, `brat` can be configured to monitor the time spent by an annotator on a document and on each editing and typing action (a similar feature is proposed in `CCASH`). `EasyRef` keeps track of the activities on the system using logs. This monitoring, which is done locally in annotation tools, is enriched by a more global management in annotation management tools like `WebAnno`, `SLATE` or `GATE Teamware`, which allows to visualize the progress of the campaign and of the annotators. However, this feature requires that the notion of corpus is taken into account, which is not the case in all annotation tools (it is for example absent in the annotation part of `Glozz`).

This evolution towards annotation management goes hand in hand with the multiplication of Web-based tools (`WebAnno`, `Slate`, `GATE Teamware`, etc.). This presents many advantages, in particular it offers the possibility to work from distance, but it can also be troublesome, for example for under-resourced languages annotation, as the annotators may have a limited Internet access.

2.3.4 The Impossible Dream Tool

The rising diversity in annotations (see Section 1.2.1) implies a variety of annotation tools. From text to video or speech, from the micro level (POS annotation) to the macro level (discourse), a unique, universal annotation tool, which would satisfy the needs and constraints (for example, the preservation of the original format) of each and everyone seems inconceivable.

Besides, many annotation campaigns managers would rather develop a new tool, adapted to the constraints of their campaign and which can be as simple as an old school `Emacs` plugin, than try and adapt to an existing tool, which would be time-consuming, could bias the annotation due to intrinsic limitations, and might in the end be disappointing.

If some tools are more used than others, often because they are well-featured and maintained (this is for example the case for `GATE` and `WebAnno`, and to a lesser extend, for `Glozz` and `brat`), there is yet, as of today, no annotation tool wining unanimous support. Developing a generic, reliable and well-documented annotation tool is a long-term endeavor. For example, it took two persons six months only to conceive `Glozz` and the same time to develop it.[26]

Besides, if there are many annotation tools available today, only few of them provide features allowing to manage an annotation campaign. To our knowledge there are only a couple of them: `Slate`, `GATE Teamware`, `Djangology`, `WebAnno` and `Egas` [Campos et al., 2014]. Moreover, two of them present important limitations: `Djangology` is not maintained anymore and `Egas` is solely provided

---

[25]This tool is mentioned here because it offered interesting original features, but it was used only in one annotation project.
[26]Yann Mathet, personal communication, January 12th, 2011.

as an online service, specializing in biocuration. Finally, none of them propose any feature to prepare the campaign (see Section 2.1). They provide no means to anticipate the complexities of the annotation campaign (see Section 2.2) and to select the appropriate automation or inter-annotator metric to apply (see Section 2.4).

The analyses of the annotation process and complexity dimensions presented in this chapter are therefore useful complements to your favorite annotation tool when preparing an annotation campaign.

2.4 Evaluating the Annotation Quality

2.4.1 What is Annotation Quality?

To be considered as "good", an annotation has to be valid, i.e. the notes added to the source have to be of the correct type and to be associated to the right segment in the flow of data. However, manually annotating is by definition interpreting, therefore there is no such thing as a "(ground) truth". We cannot directly measure the validity of manual annotation, we can only measure its reliability, i.e. how consistent the annotators were in annotating. This reveals how well they assimilated the guidelines and how coherent these guidelines are.

This *reliability* can only be evaluated by computing the agreement between annotators, or inter-annotator agreement, which is obtained by comparing the annotations of the same text made by different annotators. In addition to the inter-annotator agreement, which allows to measure the *stability* of the annotation, the agreement of the annotator with him or herself later in the campaign (the intra-annotator agreement) also needs to be computed, in order to capture the *reproducibility* of the annotation [Gut and Bayerl, 2004].

If computing the intra and inter-annotator agreements is essential, it does not have to be done on the whole corpus, for obvious reasons of cost-effectiveness. However, we strongly advice to do it very early in the campaign, so as to identify and address the problems rapidly, as was done in [Bonneau-Maynard et al., 2005].

Finally, to complete the quality evaluation, it is essential to randomly check the annotations on which the annotators agree. In *Składnica*, a Polish treebank, 20% of the agreed annotations were in fact wrong [Woliński et al., 2011].

The inter-annotator agreement research field has been very active in the past decade and is still evolving rapidly. We present here the main issues and metrics and refer the reader who would like to go further to more detailed articles, in particular [Artstein and Poesio, 2008] and [Mathet et al., 2015].

2.4.2 Understanding the Basics

How Lucky can you Get?

The most obvious and simplest metric measuring the inter-annotator agreement is the *observed agreement* (A_o). It corresponds to the percentage of times the annotators agreed, i.e. the number of agreeing annotations times 100 over the whole number of annotations. This metric is very simple and easy to compute, but it should not be used as such as it does not take chance into account. Why is that important?

In order to demonstrate the influence of chance on the inter-annotator agreement results, let us take a very simple example.[27] In an annotation task involving two categories and no segmentation (like the two sides of a coin), two annotators who would pick any of the two categories randomly (like tossing the coin) would statistically agree half of the time ($A_o = 0.5$). Therefore, in this case, an observed agreement below this baseline would be very bad (worse than by chance). The main issue with this kind of metrics is that their real scale depends on the context of the campaign: the minimum that can be obtained by chance differs according to the number of categories and annotators. This makes the results very difficult to interpret.

But there is worse. In the same case (two categories, A and B, and predefined segments) but with three annotators, it is impossible for them to completely disagree ($A_o \neq 0$): if Annotator 1 says A and Annotator 2 says B, Annotator 3 will necessarily agree with one of the first two annotators. So the observed agreement will at least be 0.33, even before taking chance into account (see Table 2.16).

Pairs	Annotations	Agreement?
Annotators 1&2	A B	No
Annotators 1&3	A A	Yes
Annotators 2&3	B A	No

Figure 2.16: Case of impossible disagreement, with 3 annotators and 2 categories.

Now, let us make a detour and consider the "truth". If the right answer was A, then they succeed in 4 out of 6 times, so they are right 66% of the time. But if the right answer was B, then they succeed in 2 out of 6 times, so they are right 33% of the time. Finally, there can be a perfect inter-annotator agreement ($A_o = 1$), for example if the three annotators say A, and 0% of truth (if the right answer was B). On the opposite, 100% of success in finding the truth implies a perfect agreement.

The same campaign with only two annotators allows for a total disagreement. In one case (3 annotators) the scale begins at 0.33 and in the other (2 annotators), it starts at 0, without even taking chance into account.

The Kappa Family

As of today, the reference article on the subject of this family of inter-annotator agreement metrics is the one written by Ron Artstein and Massimo Poesio in 2008 [Artstein and Poesio, 2008]. Its presents in details and very clearly these coefficients. We will focus here on the two most well-known, Scott's pi [Scott, 1955] and Cohen's kappa [Cohen, 1960]. These coefficients are applicable to two annotators only, but generalizations to more than two annotators are available, like Fleiss' kappa [Fleiss, 1971], a generalization of Scott's pi, or multi-κ, a generalization of Cohen's kappa [Davies and Fleiss, 1982].

---

[27]This example was suggested to us by Yann Mathet, from GREYC-CNRS (personal communication, fall 2015).

Pi and kappa are computed from the observed agreement (A_o), but they take chance into account, which is represented in the *expected agreement* (A_e). Hence, the metrics are defined using the same formula:

$$\kappa, \pi = \frac{A_o - A_e}{1 - A_e}$$

The only element that differs is the way they evaluate chance, i.e. the expected agreement (A_e). In one case (pi), the categories are affected to units by chance mimicking the way they were actually affected by the annotators, but the annotators themselves behave in the same way (their behaviors are averaged). In the other case (kappa), both the categories and the annotators can by chance behave according to the way the behaved in reality.

Scott's pi   This coefficient is also called K in [Siegel and Castellan, 1988] or Kappa in [Carletta, 1996] (or Carletta's kappa). In pi, the distributions realized by chance by the annotators are equivalent, but the chance distribution of the units (u) between categories (k) is not homogeneous and it can be estimated by the *average* distribution generated during their annotation by the annotators. The expected agreement for pi (A_e^π) is therefore defined as follows, with n_k being the number of units annotated with k by the two annotators.

$$A_e^\pi = \sum_{k \in K} \left(\frac{n_k}{2u}\right)^2$$

Cohen's kappa   This coefficient models chance by hypothesizing that the distribution of units between categories can differ from one annotator to the other. In this case, the probability for a unit (u) to be affected to a category (k) is the product of the probability that each annotator assigns it in this category. The expected agreement (A_e^κ) is therefore defined as follows n_{c1k} being the number of assignments to k for annotator 1:

$$A_e^\kappa = \sum_{k \in K} \frac{n_{c1k}}{u} \cdot \frac{n_{c2k}}{u}$$

Note that, by definition, $\pi \leqslant \kappa$. Usually, κ and π give very close results [Di Eugenio and Glass, 2004], which means that there is little bias between the annotators. It is therefore useful to compute both coefficients to check that.

The Dark Side of Kappas

The coefficients of the kappa family are very efficient, they take chance into account and are not so difficult to compute. For these reasons, they have been widely used in NLP. The problem is that they are not always appropriate. In particular, they require the number of markables (segments that could be annotated) for their computation. If it is obvious for certain tasks like POS annotation, in which all the tokens are markables, it is less easy to determine in tasks in which the discrimination is not straightforward, like in gene renaming annotation.

To illustrate this, we introduce here the most widely used representation of data for inter-annotator agreement, the contingency table. This type of representation allows not only to immediately visualize the agreement between annotators (the diagonal of the table), but also to rapidly identify the specifics of a campaign, like the Prevalence of a category, i.e. the fact that a category is used (much) more often than the others. For these reasons, we strongly advocate for the presentation of the contingency table of an annotation campaign in the accompanying articles, whenever possible (two annotators and not too many categories). We completely agree with what is said in [Hripcsak and Heitjan, 2002]:

> "showing the two-by-two contingency table with its marginal totals is probably as informative as any measure".

We present in Table 2.1 a contingency table for a toy POS annotation task with 5 categories and 100 segments, imagined from the following *Penn Treebank* example:

I/PRP do/VBP n't/RB feel/VB very/RB ferocious/JJ ./.

		Annot. 1					
		PRP	VBP	RB	JJ	Punct	Total
	PRP	15	0	0	0	0	15
	VBP	2	17	1	2	0	22
Annot. 2	RB	0	2	22	3	0	27
	JJ	0	1	2	13	0	16
	Punct	0	0	0	0	20	20
	Total	17	20	25	18	20	100

Table 2.1: (Imaginary) contingency table for a toy example of POS annotation.

In POS annotation, all the segments get an annotation, so there is no "hole" in the pavement. In this case, $A_o = 0.87$, $A_e^\kappa = 0.2058$, $A_e^\pi = 0.2062$, $\kappa = 0.8363$ and $\pi = 0.8362$.

On the contrary, in the gene renaming campaign, very few elements from the source are annotated and the empty category (no annotation) corresponding to the markables, is overwhelmingly prevalent, with 18,878 tokens (see Table 2.2).

		Annot. 1			
		Former	New	No annotation	Total
	Former	71	13	23	107
Annot. 2	New	8	69	15	92
	No annotation	7	8	18,840	18,855
	Total	86	90	18,878	19,054

Table 2.2: Contingency table for the gene renaming annotation campaign [Fort et al., 2012a].

		Annot. 1			
		Former	New	No annotation	Total gene names
	Former	71	13	23	107
A2	New	8	69	15	92
	No annotation	7	8	951	966
	Total gene names	86	90	989	1,165

Table 2.3: Contingency table for the gene renaming annotation campaign with the gene names as markables.

Considering all the tokens as markables, we obtain $\kappa \approx \pi = 0.98$.

Obviously, we could have chosen to consider as markables the gene names instead of the tokens (see Table 2.3). In this case, we obtain $\kappa \approx \pi = 0.77$.

We detailed in [Grouin et al., 2011] experiments that we led in the structured named entities annotation campaign on the inter-annotator agreement results, and we showed that the results vary quite significantly depending on the way the markables are computed.

The conclusion we draw from these various experiments is that coefficients from the kappa family should be avoided in cases in which there are "holes in the pavement", i.e. when not all of the signal is annotated, as in such cases, the necessarily arbitrary decisions in the definition of the markables may generate a prevalence bias.

The F-measure: Proceed with Caution

In some annotation campaigns, metrics usually used for the evaluation of the performance of the systems, like the F-measure, are used to evaluate the produced manual annotation. Often, this type of metric is chosen just because it is provided by default in the annotation tool, like in `GATE` (which also provides Cohen's kappa and Scott's pi). Sometimes, this choice is made to avoid the problem of the definition of the markables for the computation of kappa, for example in the case of named entities annotation [Alex et al., 2010, Grouin et al., 2011]. In fact, it was demonstrated in [Hripcsak and Rothschild, 2005] that when the number of markables is very high, the coefficients from the kappa family tend towards the F-measure.

The F-measure was designed for information retrieval and is now widely used in NLP. It corresponds to the weighted average of recall and precision:

$$\text{F-measure} = 2.\frac{\text{precision.rappel}}{\text{precision} + \text{recall}}$$

With recall and precision defined as follows:

$$\text{Recall} = \frac{\text{Nb of correct found annotations}}{\text{Nb of correct expected annotations}}$$

$$\text{Precision} = \frac{\text{Nb of correct found annotations}}{\text{Total nb of annotations}}$$

It is therefore easy to compute.

By definition, precision and recall require a reference annotation. In the case of manual annotation, we are (most of the time) building this reference, so it does not exist yet. However, one may consider that the work of one annotator can be used as a reference for the other(s). The F-measure is then computed for each category and the global metric is the average of the local ones. Moreover, it does not have to be computed both ways, as the recall of one annotator is the precision of the other [Hripcsak and Rothschild, 2005].

However, the F-measure does not take chance into account, and we saw that chance has a sometimes significant impact on the results. This limitation makes it less suitable for manual annotation evaluation than other, more adapted measures, like γ.

2.4.3 Beyond Kappas

A lot of metrics were proposed or revived, especially in the past few years, most of them to overcome the default of the kappa family metrics. We present here only a couple of them, from the weighted coefficients family, in order to introduce the final one, γ, which is very promising.

Weighted Coefficients

Weighted coefficients allow to give more importance to some disagreements than to others. The coefficients we briefly present here are more detailed in [Artstein and Poesio, 2008]: the weighted version of Cohen's kappa (κ_ω) [Cohen, 1968] and Krippendorff's Alpha (α) [Krippendorff, 2004].

Both coefficients are based on the disagreement between annotators and use a distance between categories allowing to describing how distinct two categories are. The idea behind this is that all disagreements are not equal, than some should have more weight than others. For example, a disagreement between two main categories (*Noun* and *Verb*), is more important than a disagreement in sub-types (*VerbPres* and *VerbPast*).

κ_ω and α are defined as follows:

$$\kappa_\omega, \alpha = 1 - \frac{D_0}{D_e}$$

where D_0 is the observed disagreement between the annotators and D_e the expected disagreement, i.e. the chance agreement. The expected disagreements in κ_ω and α are computed in a similar way as κ and π respectively and include the notion of distance between categories.

We will not detail the calculus of D_e, which is presented for both metrics in [Artstein and Poesio, 2008]. These metrics suffer from a major bias: distances are defined manually, based on intuition or knowledge of the campaign, and do not depend on the reality of the annotation. Another limitation is that they are dedicated to categorization tasks and do not take what Krippendorff calls unitizing into account.

Krippendorff then proposed a series of coefficients to go beyond α: αU [Krippendorff, 2004], which covers unitizing, $_u\alpha$ [Krippendorff, 2013], which focuses on positioning and $_{c|u}\alpha$ [Krippendorff, 2013], which deals with categories. "For now, it has to
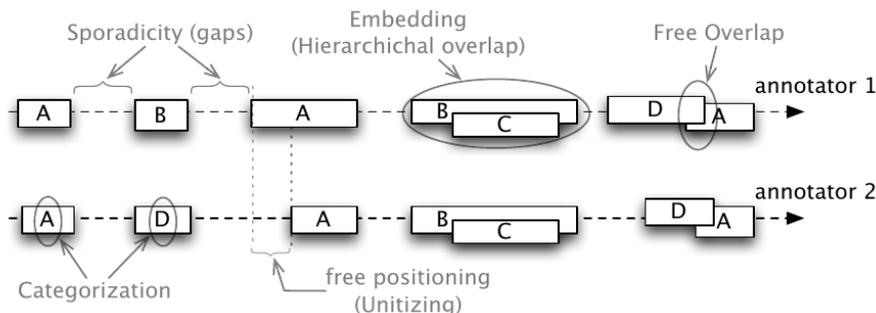
Figure 2.17: Phenomena to take into account when computing inter-annotator agreements (Figure 1 from [Mathet et al., 2015], by courtesy of the authors).

be noted that $_u\alpha$ and $_{c|u}\alpha$ are not currently designed to cope with embedding or free overlapping between the units of the same annotator." [Mathet et al., 2015]

γ: the (nearly) Universal Metrics

Yann Mathet and Antoine Widlöcher (GREYC-CNRS), who designed the annotation tool `Glozz`, also created a new metric for the computation of the inter-annotator agreement named γ[28] and is presented in details in [Mathet et al., 2015]. This metric is close to α in that it takes chance into account and does not require to identify the markables. However, γ takes the nature of the units into account and, for example, two appended entities and a unique entity spanning through two entities are considered differently in γ. The main advantage of γ is that it does not alter the annotations to compare them.

γ is holistic, it the sense that it takes the annotations from the whole corpus into account, rather than only local comparisons between units. It is also unified, as it does not dissociate between the alignment of the identified segments (discrimination and delimitation) and the agreement on the categories, both being performed simultaneously. Among the various possible alignments, γ keeps the one which minimizes the disagreement.

To do so, the metric relies on the notion of disorder of the system constituted by the set of annotations of a text. This disorder can be related to two types of dissimilarities, positional and categorial. The computation of the categorial dissimilarity requires, like in α and κ_ω, a distance between categories. This is the main weakness of the metric.

γ is defined as follows, for each annotation set j on a corpus c:

$$agreement(j) = \frac{e_{random}(c) - e(j)}{e_{random}(c)}$$

The entropy (the disorder) of an alignment of units corresponds to the average dissimilarities of its constituting units. The random entropy, $e_{random}(c)$, can be

---

[28]An earlier version of it was called the "Glozz metrics" [Mathet and Widlöcher, 2011].

computed using different methods, including the one presented in [Mathet and Widlöcher, 2011], which is implemented in `Glozz`. This method consists in observing the annotations produced by the annotators on the whole corpus and to generate, independently from the text content, multi-annotations that respect the statistical distribution of the corpus, both in terms of positions and of categories.

The main issues with the metrics taking chance into account is that computing them is not always straightforward. It is especially the case for γ. The solution is to integrate the metric into an annotation tool, so that it can be computed directly using an interface. This is what has been done for γ, which is now rather easily computable in `Glozz`.

Interestingly, a technique is underused if it is not encapsulated in a tool, and if it is, it becomes a "black box" as defined by Bruno Latour [Latour, 1987], i.e. something that is no more open, therefore no more questioned. This is exactly what happened with `GATE` and the F-measure. However, it is quite different with `Glozz` and γ, as the tool and the metric were created first for manual annotation and the metric has been well-tested before being integrated into `Glozz`.

For the moment γ is still little used, so it does not really allow for a comparison with older annotation campaigns. We therefore suggest to compute it as a complement to kappa, whenever possible.

Contrary to the annotation tools, we think that with γ the domain is now close to a rather universal solution, even if using a distance that is determined *a priori* is a bias. However, the inter-annotator agreement for relations is still a major open issue, with no appropriate solution in sight.

2.4.4 Giving Meaning to the Metrics

Ron Artstein and Massimo Poesio detail in [Artstein and Poesio, 2008] the different scales of interpretation of the Kappas that were proposed over the years (see Figure 2.18) and emphasize the fact that it is very difficult to define a meaningful threshold. What is a "good" agreement, as measured by kappa or another metric?

They conclude with caution, proposing a threshold of "reasonable quality" of 0.8 for the kappas, while adding that they "doubt that a single cutoff point is appropriate for all purposes". Other works, in particular [Gwet, 2012] that presents various inter-annotator agreement metrics, insist on the problem of their interpretation. Another, related issue is how to compare two different results obtained with different metrics.

Some studies concerning the evaluation of the quality of manual annotation allowed to identify factors influencing the inter-annotator agreement, thus giving clues on the behavior of the metrics that were used. For example, it was demonstrated in [Gut and Bayerl, 2004] that the inter-annotator agreement and the complexity of the task are correlated (which is not surprising), in particular, the larger the tagset, the weaker the agreement. In the same article it is shown that there are only a limited number of categories generating disagreement. The meta study presented in [Bayerl and Paul, 2011] extends this research and identifies eight factors influencing the inter-annotator agreement: the "domain" (we would rather talk about the annotation type, as they compare word-sense disambiguation, prosodic transcriptions and phonetic transcriptions), the number of annotators, the training of the annotators, the annotation purpose, the knowledge of the domain, the language, the number of categories and the cal-

[Landis and Koch, 1977]
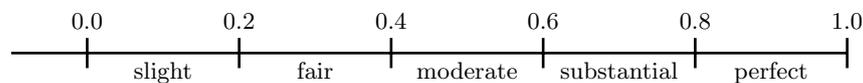

[Krippendorff, 1980]
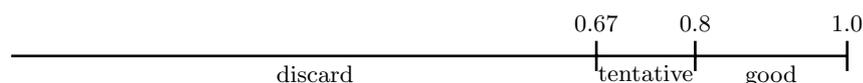
[Green, 1997]
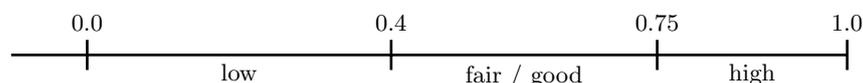
Figure 2.18: Scales of interpretation of kappas (from the ESSLI 2009 course given by Gemma Boleda and Stefan Evert on inter-annotator agreement, by courtesy of the authors) .

culation method. The authors deduce from this recommendations to improve the quality of manual annotation. However, none of these analyses give a clear view on the behavior of agreement metrics or their meaning.

We created a French working group on the subject, with people from LIMSI-CNRS (Sophie Rosset, Pierre Zweigenbaum, Cyril Grouin), LNE (Olivier Galibert and Juliette Kahn), INIST-CNRS (Claire François) and GREYC-CNRS (Yann Mathet and Antoine Widlöcher). Together, we discussed and reflected on the meaning of inter-annotation agreements and the interpretation of the results. However, the original idea and the implementation of what is presented here come from Yann Mathet and Antoine Widlöcher. This work is detailed in [Mathet et al., 2012]. We will present it here rapidly and complete it with real experiments led on the *TCOF-POS* corpus [Benzitoun et al., 2012].

The idea proposed by Yann Mathet and Antoine Widlöcher is to reverse the problem and to analyze the results obtained with the various metrics on reference annotations (or artificial annotations), which are degraded in a controlled way.

The Corpus Shuffling Tool

This idea of applying controlled degradations to a reference is derived from research in thematic segmentation described in [Pevzner and Hearst, 2002] and in [Bestgen, 2009]. It was applied for the first time to inter-annotator agree-

ment metrics in [Mathet and Widlöcher, 2011]. The principle is to generate degraded annotations in a statistically controlled way from a reference corpus. Several corpora are generated, corresponding to the different values of a deteriorated parameter, then the metrics are applied to the degraded corpora and their behavior can be observed.

The annotators produce errors that can be of various types and concern different dimensions. Each annotated unit can diverge from what it should be (a reference, imperfect by definition) in one or several ways:

- the delimitation of the unit is not correct (the frontiers do not correspond to the reference);

- the categorization of the unit is not correct (wrong category or wrong feature value);

- the discrimination of the unit is not correct: the annotation is not present in the reference (false positive);

- or, on the contrary, a unit from the reference is missing (false negative).

All these causes of errors in the annotation have to be taken into account in the inter-annotator agreement metrics. Mathet and Widlöcher developed a tool that generates "tremors" (i.e. degradations) along several dimensions.[29] These tremors are of various controlled magnitudes: the higher the magnitude, the more serious the errors. The obtained corpora with degraded annotations are then used to observe the behavior of the metrics according to different types of errors (dimensions) and a whole range of magnitudes. This allows not only to compare the metrics (for a given magnitude, it is easy to compare the results obtained by the different metrics), but also to interpret the scores in a tangible manner (a given score for a given metric corresponds to a certain magnitude, of which we know the effects on the corpus). This tool takes as input the magnitude of error, from 0 (the perfect annotator) to 1 (the worst annotator, who annotates without even reading the text).

Experimental Results

The experiments presented in [Mathet et al., 2012] implied artificial annotations, i.e. annotations which were generated automatically from a statistical model describing the positional and categorial distribution of the markables. We will focus here on the obtained results rather than on the protocol, which is detailed in the article, and will present an experiment carried out on a real corpus.

At the time of the experiments, the `Corpus Shuffling Tool` did not allow for a combination of paradigms to be taken into account. We therefore had to process segmentation and categorization separately. A first experiment was carried out on segmentation alone, rerunning the one described in [Bestgen, 2009], comparing the generalized Hamming distance and WindowDiff, and adding γ. We will not present this experiment here, as it concerns metrics that cannot be considered as inter-annotator agreement metrics, since they require a reference. We simulated three annotators.

---

[29]This tool is freely available under a GPL license and will soon reappear here: `http://www.glozz.org/corpusshufflingtool`.

Artificial annotations   We present here results concerning the categorization process. The simulated situation is that of an annotation task in which the units to annotate are already localized, like in the *Penn Treebank* POS annotation. We created four annotation sets, including or not prevalence cases and a structured tagset, for which we consider that an error between a sub-category and a category should be considered as less serious than one between categories.

The `Corpus Shuffling Tool` was applied on this annotations to compare the following metrics: Cohen's kappa [Cohen, 1960], weighted kappa [Cohen, 1968], with two different weight matrices (the first one being much more lenient than the other) and γ, with or without the ability to deal with a structured tagset (taking the proximity between categories into account). An observed agreement (percentage of strict agreement between the three annotators) is also computed as baseline. The results of these experiments are presented in Figure 2.19.
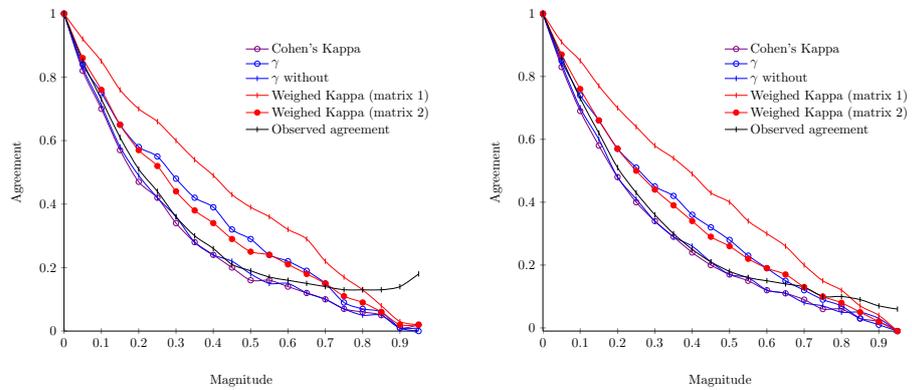
(a) Annotations with a structured tagset and prevalence

(b) Annotations with structured tagset
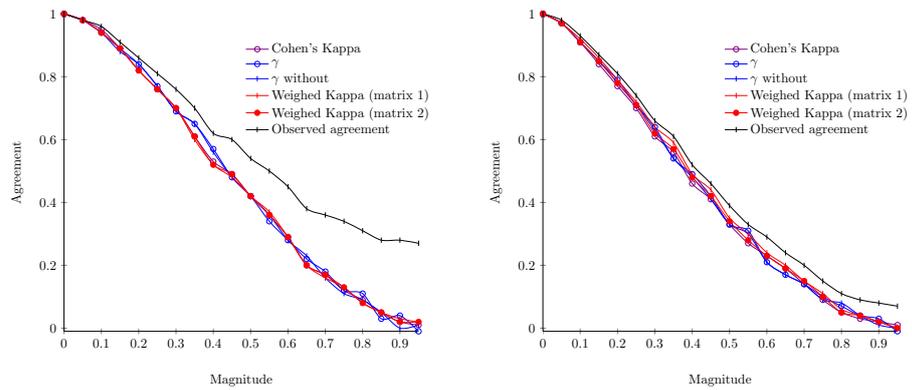
(c) Annotations with prevalence

(d) Annotations without any structure in the tagset or prevalence

Figure 2.19: Comparison of the behaviors of the metrics on categorization.

These results show first that when there are no prevalence and no structured tagset (with different types of proximity between categories), all the compared metrics behave similarly (see Figure 2.19d), including the observed agreement

(even if it slightly overestimates the agreement when the magnitude gets higher, because it does not take chance into account).

In cases of prevalence of one category over the others (see Figure 2.19c), all the metrics continue to behave similarly, apart from the observed agreement, which tends to more and more overestimate the agreement, by nearly 0.25 at most. Chance has a significant impact here.

In case of a structured tagset, the weighted kappa and γ behave very differently than the other metrics. When taken into account, the more or less important proximity between categories, whether it is associated to prevalence or not (see Figures 2.19a and 2.19b, respectively), generates noticeable differences, of 0.15 for γ and 0.25 for the weighted kappa. This can be easily explained by the fact that these metrics use a matrix actually describing the proximity between categories, whether this matrix is created directly from the data (γ) or provided by the user (weighted kappa). Moreover, it is interesting to note that when applying these two metrics to data without a structured tagset (or at least without taking it into account), they behave almost exactly the same way as the simpler metrics which do not take the proximity into account (bottom figures). These metrics (γ and weighted kappa) are not biased, whatever the corpus.

As for the observed agreement, it is closer to the other metrics in cases where the tagset is structured, probably due to the fact that in cases of average magnitudes, proximity is more influential than prevalence. However, with a magnitude above 0.6, the observed agreement overestimates the agreement again.

Annotations from a real corpus Since we were limited by the fact that we could not combine various dimensions, we chose to focus on categories only. *TCOF-POS* [Benzitoun et al., 2012] is a freely available corpus[30] of French spontaneous speech annotated in POS. It was perfect for our experiment, as the corpus was pre-segmented and the annotators did not have to review this, but only to categorize the units. Besides, this annotation task did not generate significant prevalence. However, the used tagset contained a hierarchy of types (*PRO:ind*, *PRO:dem*, *PRO:cls*, etc.), which has to be taken into account (an error between two types should be considered as more serious than one between a type and a subtype).

The results we obtained are presented in Figure 2.20. They confirm the ones we got from artificial annotations. The observed agreement, which does not take chance into account, this time under-estimates the agreement. The weighed kappa seems to be the metrics that underestimates the least the agreement in this case. However, this metrics was computed from a user-defined matrix of weights, deduced from the annotation guide. These weights take into account the fact that an error between two categories (two types) is more serious than one between a category and its sub-categories. For example, the weight associated to an error between the following two sub-categories *Verb-PPRES* and *Verb-FUTUR* of the same category (*Verb*) is 0.5, whereas the weight associated to an error between two categories, like *Verb-PPRES* and *Noun* would be 1.

Originally, the inter-annotator agreement on this corpus had been computed using Cohen's kappa and reached 0.96. On the Richter scale obtained using the

---

[30]The corpus is available here: `http://www.cnrtl.fr/corpus/perceo/`.

`Corpus Shuffling Tool` and presented in Figure 2.20, this corresponds to a magnitude of 0.1, i.e. to a very limited deterioration. We can therefore say now without any doubt that the corpus is annotated in a very consistent way.
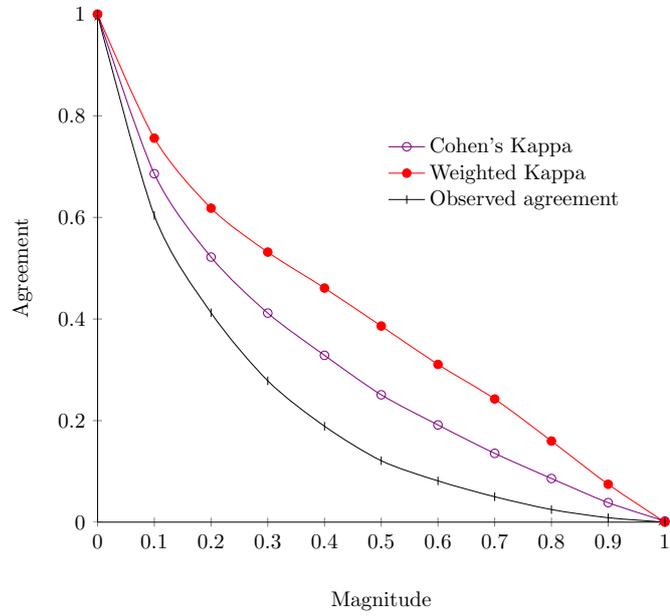

Figure 2.20: Comparison of the behaviors of the metrics on categorization on the *TCOF-POS* corpus (no prevalence, but structure of the tagset taken into account).

Chapter 3

Crowdsourcing Annotation

Crowdsourcing represents a dramatic evolution for language resources development and data production in general. Not only does the change in scale push to their limits the annotation methodologies and tools, but it modifies our relationship, as researchers, to the citizens, who become our employees or our partners.

These specificities makes it the perfect testbed for manual annotation "engineering".

3.1 What is Crowdsourcing and Why Should we be Interested in it?

3.1.1 A Moving Target

Defining the term "Crowdsourcing" appears to be a research subject as such [Estellés-Arolas and González-Ladrón-de Guevara, 2012]. As new applications appear, the definition evolves and revolves around the chosen focus. Rather than adding ours to the pile, we will comment the mostly used definitions and give examples, that we will go into in depth to deconstruct the myths.

The term "crowdsourcing" was first coined by Jeff Howe of Wired Magazine as "the act of a company or institution taking a function once performed by employees and outsourcing it to an undefined (and generally large) network of people in the form of an open call." [Howe, 2006b]. The crowdsourced definition of crowdsourcing in `Wikipedia` is now that of the Merriam Webster's dictionary: "the practice of obtaining needed services, ideas, or content by soliciting contributions from a large group of people and especially from the online community rather than from traditional employees or suppliers"[1]. But it used to be[2] closer to the original definition:

Definition 8 (Crowdsourcing) *"the act of outsourcing tasks, traditionally performed by an employee or contractor, to an undefined, large group of people or community (a crowd), through an open call".*

---

[1]See: `http://www.merriam-webster.com/dictionary/crowdsourcing`.
[2]Consulted on December 2, 2010.

Anyway, two important differences are to be noted between Howe's definition and the other ones. First, Howe does not emphasize as much the number of participants. Second, using the term "network", he insists on the relationship between participants. We will soon see that his definition is closer to reality than `Wikipedia`'s and Merriam Webster's from those points of view.

Besides, Howe explicitly defines the source of the task to perform as "a company or institution", thus differentiating between "crowdsourcing" and "commons-based peer production" (also known as "social production") [Howe, 2006a], where the task can be initiated by anyone, like `Wikipedia`[3] or the `Distributed Proofreaders`[4] of the `Gutenberg project`. The importance given to the distinction for Howe is linked to his economics standpoint and seems somewhat artificial for our purpose. We will therefore consider "social production" as part of crowdsourcing rather than a distinct production mode.

Due to the instability in the definition, there are today many crowdsourcing taxonomies, a lot of which are presented in [Geiger et al., 2011]. We will use the following two criteria to categorize crowdsourcing applications: (i) is the crowdsourced task remunerated (remuneration)? (ii) is the purpose of the task obvious to the participant (transparency)? Using this simplified grid of categorization, and focusing on crowdsourcing for data production, a superset including annotation, we obtain three main types of crowdsourcing:

1. volunteer and transparent crowdsourcing, such as `Wikipedia` and `Distributed Proofreaders`

2. remunerated and transparent crowdsourcing, which includes microworking crowdsourcing platforms like `Amazon Mechanical Turk`

3. volunteer and non transparent crowdsourcing, including GWAPs like `JeuxDeMots` and `ZombiLingo`

The purpose of GWAPs can be to solve a problem or to create data, in our case it is to generate language data.

Obviously, as in any categorization process, there is a continuum here and a well-known GWAP for NLP like `Phrase Detectives` can prove difficult to categorize. First, it is more a gamified interface than a game, so the task is rather transparent, but it is presented in an indirect way. Second, players can win gift cards, which is not a remuneration, but still represents money. However, this is not the main motivation for playing and it is not emphasized like, for example, on `Amazon Mechanical Turk`. Despite its specifics, `Phrase Detectives` can therefore be included in the third category.

We will use examples from these three types of applications throughout this chapter to illustrate what can be expected from crowdsourcing from the annotation standpoint.

3.1.2 A Massive Success

Crowdsourcing is undoubtedly a massive success and it represents more than just a low-cost solution, a shift in power.

---

[3]See: `https://en.wikipedia.org`.
[4]See: `http://www.pgdp.net/`.

Started in 2001, the `Wikipedia` crowdsourced encyclopedia contains more than 30 million articles in more than 241 languages[5]. Its English version is accessed 8 million times an hour (800,000 for the French version).

The `Distributed Proofreaders` was created in 2000 to assist the `Project Gutenberg`[6] in providing and correcting digitalized Public Domain books. The number of active proofreaders is around 1,000 (1,372 in February 2015, 1,209 in August) and they proofread nearly 30,000 books [7] in 7 languages (English, German, French, Italian, Portuguese, Spanish, Dutch).

The first well-known and advertised Game With A Purpose (GWAP, see Section 3.3), the `ESP game`, proposed the players to tag images. The success was such, with 13,500 participants tagging 1.3 million images in 3 months [von Ahn, 2006], that its creator, Luis von Ahn, sold it to *Google*.

In a very different domain, the French law imposes representatives to declare their conflicts of interest, who handed out unusable hand-written forms. The association `Regards citoyens` (citizen watch) created an online platform to digitize the scanned pdf files[8]. With a little advertisement in the press, 11,095 declaration extracts were digitalized in less than a week by nearly 8,000 participants. The declarations are now searchable and were used by journalists to unveil misconducts.

3.2 Deconstructing the Myths

3.2.1 "Crowdsourcing is a recent phenomenon"

The advent of the Web 2.0, in the 2000s has witnessed the multiplication of crowdsourcing activities, from the benevolent `Wikipedia` (2001), to the Microworking platform `Amazon Mechanical Turk` (2005). However, the phenomenon is far from being recent. In particular, the scientific community did not wait for the Internet to become "social" to start using the power of the crowd through "citizen science".

A good example of this type of open call to perform science is the `Longitude Prize`, a £20,000 prize[9] awarded in 1714 by the British government to reward a simple and handy method to determine the longitude of a ship. The challenge still exists and in 2014 was focused on "Global antibiotics resistance". Another interesting example comes from the French National Museum (*Museum National d'Histoire Naturelle*), in Paris, which published in 1824 "Instructions for the travelers and the employees of the colonies" (*Instructions pour les voyageurs et les employés des colonies*, see Figure 3.1), a guide for non-academic scientific travelers, explaining how to publicize their own experiments and observations in order for the scientific community to benefit from this work.

Nevertheless, the Web represents a dramatic change in scale, allowing to reach a potential of more than three billion people[10] with a click of a mouse. The evolution towards an interactive, "social" Web, that we witnessed in the

---

[5]`http://stats.wikimedia.org/EN/Sitemap.htm`
[6]See: `http://www.gutenberg.org/`.
[7]`http://www.pgdp.net/c/stats/stats_central.php`
[8]See: `http://regardscitoyens.org/interets-des-elus/`.
[9]See `https://longitudeprize.org/`.
[10]See `http://www.internetlivestats.com/internet-users/`.
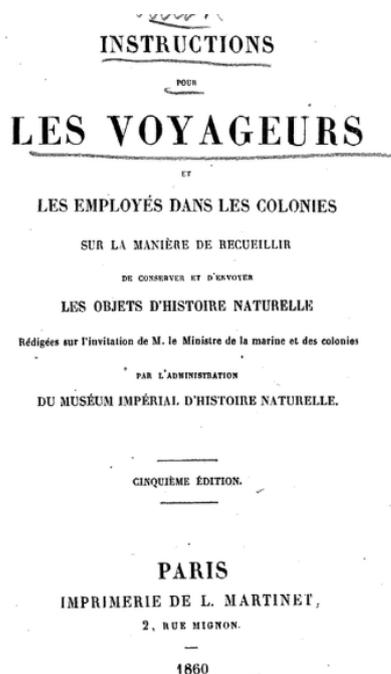
Figure 3.1: Instructions for the travelers and the employees of the colonies (French National Museum, 1860). The first edition dates from 1824.

2000s, offers even more possibilities for crowdsourcing, as the users can now directly interact and participate to the project on the Web site.

3.2.2 "Crowdsourcing involves a crowd (of non-experts)"

It reflects in the name itself: crowdsourcing implies a crowd (of participants). But what exactly is a crowd? and how many persons does it take to call it so? hundreds? dozens? more than ten? Although we can probably agree on the fact that less than a dozen is definitely not a crowd, the limit is hard to set.

According to the Merriam-Webster online dictionary[11], a crowd is first "a large number of persons especially when collected together", which corresponds to the idea of a multitude, without defining what "a large number" is. The dozen of most active participants in `Phrase Detectives` can hardly be considered as a crowd. However, the third sense of the word is "a group of people having something (as a habit, interest, or occupation) in common", a definition that focuses more on a common activity and less on the number of participants. This vision of the crowd corresponds precisely to what crowdsourcing is.

The "(out)sourcing" part of the term reflects the fact that the call to participants is open: anyone, with or without expertize in the domain, can participate.

---

[11]See `http://www.merriam-webster.com/dictionary/crowd`.

Obviously, this does not mean that everyone will. First, they have to know about the call and second, they have to be motivated to be part of it.

The reality of crowdsourcing practice is therefore far from what is being advertised. In `Phrase Detectives`, for example, although 2,000 persons registered as players between 2011 and 2012, only 13 created most of the annotations (see Figure 3.2) [Chamberlain et al., 2013].
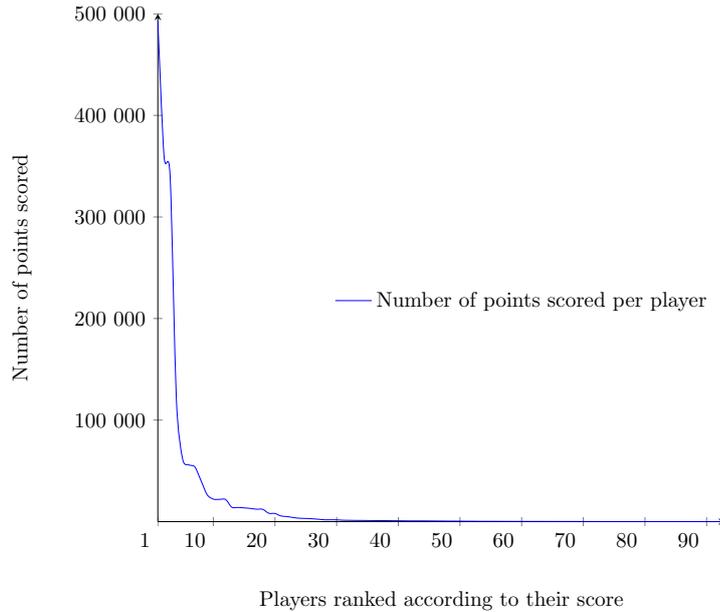
Figure 3.2: Number of players on `Phrase Detectives` according to their scores (Feb. 2011 - Feb. 2012).

The same phenomenon appears on `JeuxDeMots`, with more than 1,200 players as a whole (see Figure 3.3[12]) and `ZombiLingo`, with a total of 370 players (see Figure 3.4). The power law curve is typical of human activities. It also reflects the difference between advertising and reality. For example, `Amazon Mechanical Turk` announces a workforce of 500,000 Turkers (workers)[13], while we computed in [Fort et al., 2011a] that 80% of `Amazon Mechanical Turk` Human Intelligence Tasks(HITs) are completed by 3,011 to 8,582 Turkers, and that there are 15,059 to 42,912 Turkers in total. These numbers are confirmed by [Stewart et al., 2015], which states (in its own title) that "The average laboratory samples a population of 7,300 Amazon Mechanical Turk workers".

The "crowd" in `Amazon Mechanical Turk` corresponds to the first sense of the word: a multitude of persons. However, the players on `Phrase Detectives` `JeuxDeMots` and `ZombiLingo` can certainly be considered as a crowd, because they engage in a common activity.

Participants in `Wikipedia` correspond to the same type of "common activity" crowd. The crowdsourced encyclopedia is, to our knowledge, the eldest online

---

[12]The numbers are regularly updated and available at: `http://www.jeuxdemots.org/generateRanking-4.php?cat=scores&ts=1000&td=all&rd=10000`.

[13]See `https://requester.mturk.com/tour`.
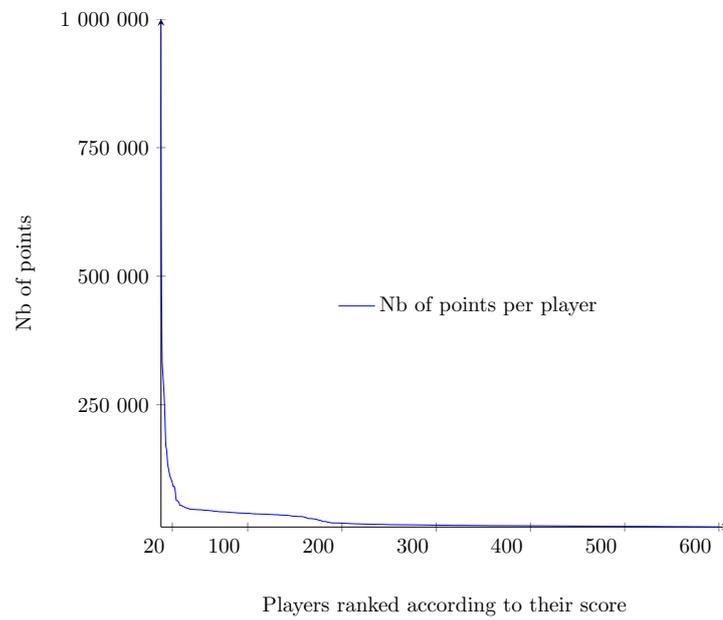
Figure 3.3: Players on `JeuxDeMots`.
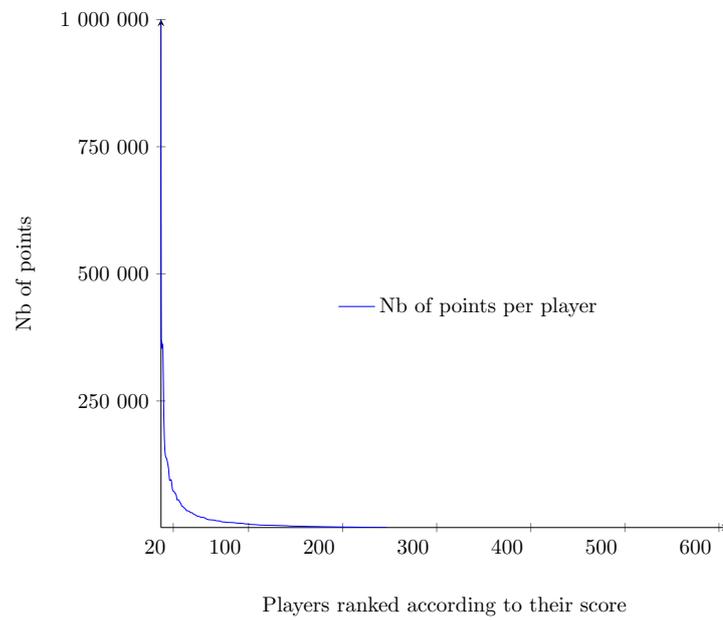
Figure 3.4: Players on `ZombiLingo`.

crowdsourcing application. As such, it offers a perspective on the functioning of such a system that no other does. It is no surprise that the study of its participants shows more complex patterns.

The rise in the number of editors began slowly, with an acceleration from 2004 to 2007 (see Figure 3.5) to reach a peak at 56,400 active editors in 2007. It then began to decline steadily. This decline was attributed to an excess in rejection of contributions, leading to a drastic change "from 'the encyclopedia that anyone can edit' to 'the encyclopedia that anyone who understands the norms, socializes him or herself, dodges the impersonal wall of semi-automated rejection and still wants to voluntarily contribute his or her time and energy can edit'"[Halfaker et al., 2013]. As of today, and for the English `Wikipedia`, the most important edits are from almost once in a lifetime editors[14] and the most active editors mainly perform minor (but numerous) edits.
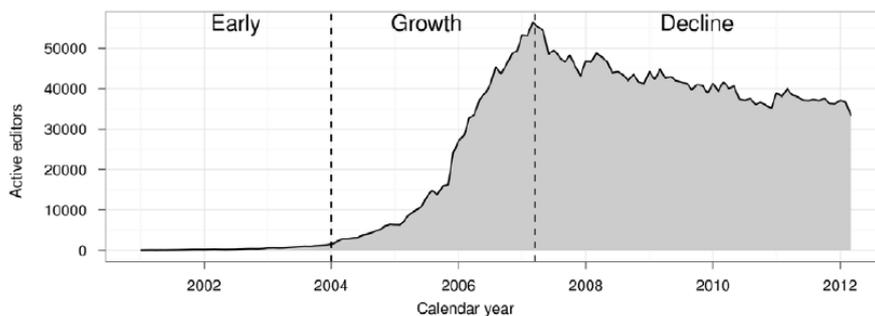
Figure 3.5: Number of active, registered editors ($>=$ 5 edits/month) in `Wikipedia` (Figure 1 from [Halfaker et al., 2013], by courtesy of the author (CC-BY-SA)).

One of the main issues with crowdsourcing systems is to manage to keep the momentum going: at least some of the newcomers should become the new "leaders", the new experts.

3.2.3 "Crowdsourcing involves (a crowd of) non-experts"

People participating in crowdsourcing activities are often referred to as "amateurs" or "non-experts", as opposed, for example in citizen science, to researchers or to traditionally employed people in agencies like the LDC[15] in the US or ELRA-ELDA[16] in France.

The definition of an expert, as given by the Merriam-Webster dictionary[17] is the following: "one with the special skill or knowledge representing mastery of a particular subject ". The definition of the adjective bears more informative power: "having, involving, or displaying special skill or knowledge derived from training or experience".

---

[14]See: `http://www.aaronsw.com/weblog/whowriteswikipedia`.
[15]See `https://www.ldc.upenn.edu/`.
[16]See `http://www.elra.info/en/`.
[17]See `http://www.merriam-webster.com/dictionary/expert`.

The question of the expertise of the annotators is one that regularly appears in research articles dealing with manual annotation, but is seldom considered as a subject as such. Authors write about "expert" annotators, sometimes opposing them to "naive" or "non-expert" annotators (see for example [Tellier, 2014]), without defining those terms, either because they seem obvious or because that would open the Pandora box of acceptability judgments. Sometimes, more details are given concerning the annotators' background [Péry-Woodley et al., 2011] or they are defined as "domain experts" [Candito et al., 2014]. In rare cases, a very detailed description is given, like in [Geertzen et al., 2008]:

> Naive annotators can be characterized as subjects that have not been linguistically trained but that have participated in an introductory session explaining the dialogue data, the dialogue act tagset, and the use of an annotation tool. Expert annotators can be characterized as linguistically trained subjects that have experience in annotating dialogue and are thoroughly familiar with the tagset.

However, this description is not linked to any general typology.

This ambiguity in the definition of the annotators' expertise is obvious in domains like biomedicine. A long discussion was held in October 2012 on the BioNLP mailing list[18] concerning the type of expert that would be the most efficient to annotate "linguistic" elements in a biomedical corpus, like protein or gene names: a biomedical specialist of a linguist? To illustrate the issue, let us take the following (real) example from the French *Sequoia* corpus [Candito and Seddah, 2012], which is manually annotated with constituents (syntax):

> Pour les SCA, la durée de la perfusion dépend de la manière dont le SCA doit être traité : elle peut durer jusqu'à 72 heures au maximum chez les patients devant recevoir des médicaments. [For the ACS [Acute Coronary Syndroms], the duration of the IV [intravenous drip] depends on the way the ACS should be treated: it can last a maximum of 72 hours for patients who need to take drugs.]

Who would be an expert in this case? The sub-corpus (*EMEA*) is from the pharmacology domain and the annotation is of a certain type of linguistics (a certain type of syntax). Would you need a linguist? A pharmacist? Would a French-speaking person, without any specific knowledge in syntax or pharmacology, but trained for the task, be able to annotate (correctly) this type of sentence? If yes, would s/he be an expert or a non-expert (a naive annotator)?

It seems to us that it is crucial to distinguish between (i) the expertise of the domain of the corpus (here, pharmacology), (ii) the expertise of the annotation domain (here, a certain type of syntax) and (iii) the expertise of the task (here, annotating constituents with a certain tool, according to certain guidelines). In all cases, at the heart of expertise lies training.

The crowdsourcing participants who produce a lot of data do have experience and in many cases they are well-trained. We know that because we know that they produce quality data. In `Phrase Detectives` and in `ZombiLingo`,

---

[18]The discussion title was: "Trends in Clinical NLP (Jon Patrick)".

for example, the games include reference annotations which help evaluate the quality of the players' production. In `Phrase Detectives`, the accuracy of the players ranges from 0.7 to 0.8 when compared to experts of the annotation domain [Chamberlain et al., 2009a]. However, the identification of "properties" resulted in a massive failure with an accuracy close to 0. In the following example, postman is a property of Jon:

(3.1) Jon, the **postman**, delivered the letter.

If we can consider that at least some players became experts in the anaphora annotation task as presented in `Phrase Detectives`, none of them managed to master the annotation of properties and become experts of the task. This is probably due to a lack of training, with the tutorial focusing on anaphora rather than properties.

In `ZombiLingo`, early results show an average of 0.86 accuracy for the 10 best players (removing the results from the best one and the worst one, see Table 3.1). The accuracy gets even better if we include all the players (around 0.9), but then the majority of the participants played only the easiest phenomena. It has to be noted, however, that these results underestimate the produced noise (identification of a phenomenon where there is none) in non-reference cases, due to a (still) badly designed interface. To attract the attention of the players on this, the tutorial has to include sentences without the phenomenon at hand, which we added in the new version.[19]

Rank	Correct	Total	%
1	1, 702	1, 785	95.35%
2	549	985	55.74%
3	807	909	88.78%
4	566	659	85.89%
5	477	517	92.26%
6	318	444	71.62%
7	296	382	77.49%
8	201	214	93.93%
9	195	213	91.55%
10	131	150	87.33%

Table 3.1: Accuracy of the most productive players in `ZombiLingo`.

The players do manage to annotate dependency relations in domains like pharmacology, as they became experts of the annotation task (see Figure 3.6).

If the training of the players is at the heart of GWAPs, especially for manual corpus annotation, it is not planned in microworking. In particular, platforms like `Amazon Mechanical Turk` do not provide for any specific means to train the workers (*Turkers*). The latter are at the mercy of wrongdoings from *Requesters* (persons proposing tasks) who can decide to exclude them without even having to justify it [Fort et al., 2011a]. *Turkers* therefore have to train by themselves on the task, without being remunerated. This generates a large amount of hidden work, as shown in [Gupta et al., 2014].

[19]Version 2.0 was released mid-January 2016.
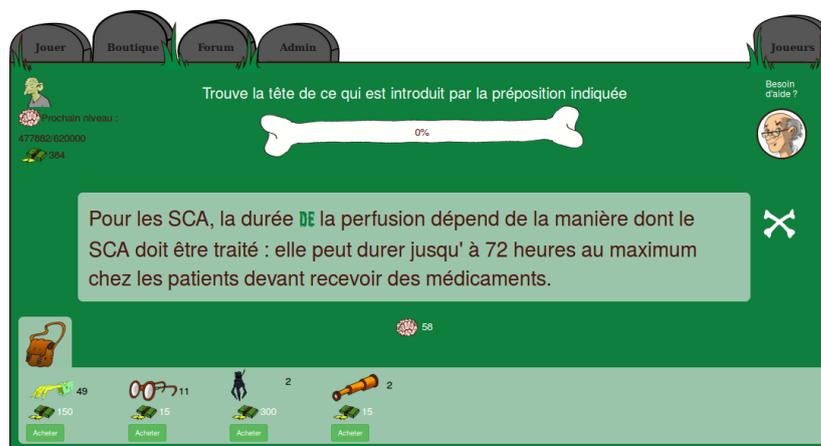
Figure 3.6: Annotation of a dependency relation with `ZombiLingo` in the pharmacology domain: "For the ACS [Acute Coronary Syndroms], the duration of the IV depends on the way the ACS should be treated: it can last a maximum of 72 hours for patients who need to take drugs".

On `Amazon Mechanical Turk`, *Turkers* can be awarded the distinction of *Masters*: "Workers achieve a Masters distinction by consistently completing HITs [tasks] of a certain type with a high degree of accuracy across a variety of Requesters [people proposing tasks]"[20]. However, this distinction is not awarded by *Requesters*[21], who seem to favor the identification of their own experts, mainly because they do not have to pay them as much as Masters. Thus, some *Turkers* can probably be considered as experts, whether they be Masters[22] or not, as they specialized in one or several types of HITs.

As we already saw, the English `Wikipedia`, after a rise in edits from administrators, saw from 2004 an evolution towards more newcomers' edits until 2007, when the number of active editors began to slowly decrease (see Figure 3.5) [Kittur et al., 2007]. These active editors now mainly perform the maintenance, cleanup and encyclopaedian police work. They know their Five Pillars[23] by heart and enforce them with more or less patience. They are considered by some as `Wikipedia`'s bourgeoisie [Kittur et al., 2007].

The quality of `Wikipedia` is a field of research as such, of which an interesting state-of-the-art is presented in [Calzada and Dekhtyar, 2010], but also in `Wikipedia` itself.[24] More or less scientific studies were carried out, trying to compare the crowdsourced encyclopedia with, for example, the venerable `Encyclopedia Britannica`. Although it is much like comparing apples and

---

[20]See `https://www.mturk.com/mturk/help?helpPage=worker#what_is_master_worker`

[21]See the point of view of *Requesters* on this: `http://turkrequesters.blogspot.fr/2012/11/thought-masters-was-just-bad-for-non.html`.

[22]The real number of Masters is difficult to evaluate (it is probably less than advertised by Amazon).

[23]See: `https://en.wikipedia.org/wiki/Wikipedia:Five_pillars`.

[24]See: `https://en.wikipedia.org/wiki/Reliability_of_Wikipedia`.

oranges, as `Wikipedia` has a much wider coverage than the `Britannica`, the outcome is interesting: articles in `Wikipedia` show a great variety in length and readability, but information they provide is overall right. When comparable to the `Britannica`, the real issue lies in omission errors. Direct assessment methods were also developed (see, for example [Voss, 2005, Hu et al., 2007, Calzada and Dekhtyar, 2010]), including within `Wikipedia` itself.[25] This is a never ending issue, as `Wikipedia` evolves constantly and hopefully, thanks to the administrators' efforts, it gets better. However, the best way to ensure its quality is to (stop complaining and) participate!

3.3 Playing With A Purpose

Mathieu Lafourcade defines GWAPs as games that are useful for a community (of researchers, for example) and real games for the players [Lafourcade et al., 2015b]. This definition suggests that the players are only participating to play. This underestimates the importance given by some players in their participation in a scientific project (citizen science). We therefore prefer to give it a more open definition:

Definition 9 *A Game With a Purpose is an entertaining setting (a gamified interface, or a full-fledged game) in which voluntary participants produce data that require human knowledge and learning capabilities.*

The purpose of these games can vary, from problem solving (`FoldIt`) to language resources creation (`JeuxDeMots`, `ZombiLingo`).

A detailed list of games can be found in French in [Lafourcade et al., 2015b]. This section is intended to show the possibilities offered by GWAPs, rather than to list the existing ones. It therefore focuses on a limited number of examples.

Also, for the sake of analysis, games are classified into three groups, but there is obviously a continuum here, as, for example, all games imply some sort of training period. However, in some of them, the training concerns mainly the interface and the gameplay, while in some others, it concerns the task itself.

3.3.1 Using the Players' Innate Capabilities and World Knowledge

Using the innate capabilities and world knowledge of the players is the obvious way to go.

In the NLP field, `JeuxDeMots`[26] (see Figure 3.7) uses the players' ability to associate ideas to create a lexical network [Lafourcade and Joubert, 2008]. One player's associations of ideas are compared to another random player and points are gained only when they found identical terms, especially if the terms are new to the network. In this sense, it is a directly collaborative game, in which the participant knows with whom s/he has been compared to.The game allowed to create more than 48 million relations in a network containing more than 800,000 terms and named entities.[27] More importantly, these relations are constantly

---

[25]See: `https://en.wikipedia.org/wiki/Wikipedia:WikiProject_Wikipedia/Assessment`.
[26]See (and play, in French): `http://www.jeuxdemots.org`.
[27]These numbers are provided on the homepage of the game, as of January 10th, 2016.

updated and the resulting network can be freely downloaded at anytime[28]. This makes the created language resource completely dynamic, a unique feature in the language resources world.

Another interesting characteristic of `JeuxDeMots` is that it is not just a gamified interface, but a full-fledged game, with a rich long-lasting gameplay. Apart from the usual leader board and scored points, `JeuxDeMots` offers the player a whole range of interactions with others: possibility to challenge other players, to bring them to "justice" (lawsuits) when their games seem too badly played, to send "hot potatoes" or gifts, etc. Most players in `JeuxDeMots` do not seem to care about the created language resource or the fact that they play to help researchers: they play simply because they like the thrill of the game.
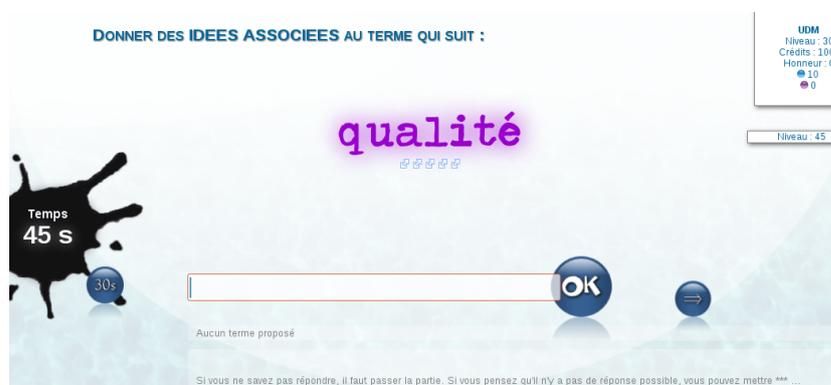
Figure 3.7: Interface of the game `JeuxDeMots`: give ideas associated with the following term: "Quality".

Moreover, `JeuxDeMots` gave birth to a galaxy of smaller GWAPs, often voting games, that offer a useful distraction from the main game, while allowing to enrich the produced language resource. As an example, `LikeIt` lets you vote on whether you like a term or not and see what the other players chose [Lafourcade et al., 2015a].[29] The game allowed to polarize 25,000 terms in three months with 150,000 votes. As a comparison, `Polarimots` [Gala and Brun, 2012], a polarized lexicon created manually by three linguists contains 7,473 words. One could think that the quality would differ, but a comparison shows that `LikeIt` agrees with `Polarimots` in almost 93% of the cases covered by the latter[30]. Besides, the resource created by `LikeIt` is dynamic and will therefore keep track of changes in polarity over time (for example "terrific" used to be negative, as well as "trop" in French).

3.3.2 Using the Players' School Knowledge

Another possibility is to rely on the basic knowledge learned at school by the players.

---

[28]Here, under a Public Domain license: `http://www.jeuxdemots.org/JDM-LEXICALNET-FR/?C=M;O=D`.

[29]See (and play, in French): `http://www.jeuxdemots.org/likeit.php`.

[30]See: `http://www.jeuxdemots.org/data/eval_polarimot.php`.

This is what the designers of `Phrase Detectives`[31] did concerning grammar, in particular anaphora (or co-reference) resolution [Chamberlain et al., 2008]. They built a gamified interface allowing the players to annotate the referent of a term (see Figure 3.8). The players are given detailed instructions and are briefly trained on the task, before they can actually annotate. This interface allowed to produce a 200,000 words annotated corpus, with a relatively high level of quality (see Section 3.2.3).
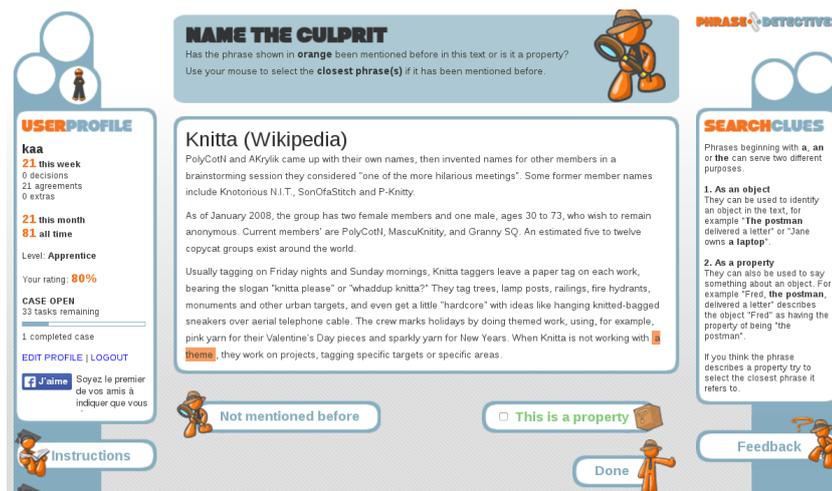
Figure 3.8: `Phrase Detectives`: annotation interface.

`Phrase Detectives` alternates the annotation mode with a second playing mode, in which participants are asked to judge other players' annotations (see Figure 3.9). This peer validation adds a collaborative layer to the interface: an annotation is considered correct if enough participants agree on it. This collaboration is indirect, as the participant do not know whose annotations they are correcting or agreeing with.

3.3.3 Using the Players' Learning Capacities

As GWAPs proved their efficiency, people started looking beyond the already acquired capacities of the participants and considered training them on complex tasks. This is how another type of GWAPs has emerged in the past few years, using the players' virtually endless learning capacities.

`FoldIt`[32] [Khatib et al., 2011] is a perfect example of this type of GWAP: it helps players, without any prior knowledge in biochemistry, learn how to fold proteins [Cooper et al., 2010]. Building upon the extraordinary innate 3D capabilities of human beings, the game includes a tutorial, broken down into concepts, with puzzles for each concepts, giving access to the following puzzles only if the player has reached a sufficient (skill) level (see Figure 3.10).

By playing `FoldIt`, a team of players found in a couple of weeks the solution to the crystal structure of a monomeric retroviral protease (simian AIDS-causing

---

[31]See `https://anawiki.essex.ac.uk/phrasedetectives/`.
[32]See `http://fold.it/portal/`.

Figure 3.9: `Phrase Detectives`: peer validation interface.
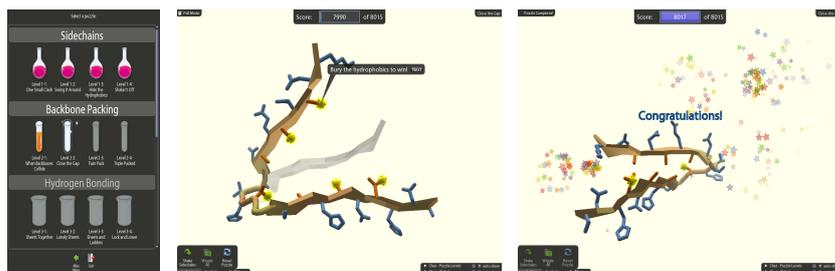
Figure 3.10: `FoldIt`: game interface.

monkey virus), an issue which had remained unsolved for over a decade (see the progress of the team on Figure 3.11). This result could allow for the creation of antiretroviral drugs, an amazing outcome for a game!

More recently, we developed such a GWAP for dependency syntax annotation, a notoriously complex annotation task. The resulting game, `ZombiLingo`[33] [Fort et al., 2014b], allows players to learn to annotate complex syntactic phenomena while eating heads to become a more degraded zombie (see Figure 3.12). To do so, we first identified the complexity dimensions of the annotation task, following the complexity grid proposed in [Fort et al., 2012b] and presented in Chapter 2.2. This led us to decompose the task into phenomena (dependency relations) instead of sentences and to propose a tutorial for each of them (see Figure 3.13). As in `Phrase Detectives`, the players are not allowed to play a phenomenon until they reach a certain level of success during the tutorial.

---

[33]See `http://zombilingo.org/`.
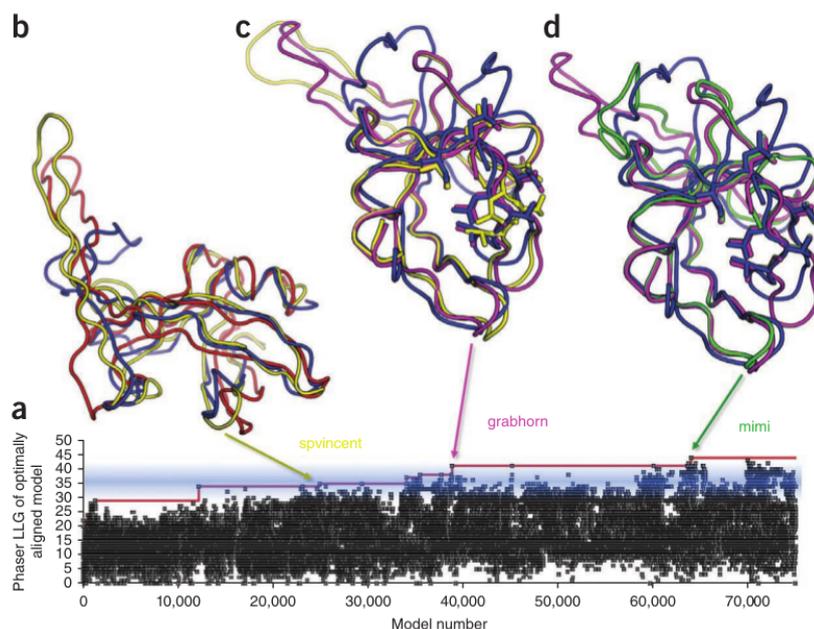
Figure 3.11: `FoldIt`: progress of the team that found the solution (pseudos of the players are shown in colors).

However, the training does not stop at the tutorial stage. Players are regularly proposed reference sentences and if they do not annotate them properly, a correction is given, the level of confidence we place in the players is lowered and their chance to play another reference sentence is raised. Besides, they are warned that after three faulty annotations they will have to redo the tutorial on this phenomenon. We also offer the participants the possibility to retrain using the tutorial whenever they need (for example, after a long period without playing).

The input corpus is pre-annotated using a parser (for French, we used `Talismane` [Urieli, 2013]) and the pre-annotated relations are given a score of 5. The players confirm or correct the pre-annotations. Each time a player selects a relation, the score of this relation is increased by the level of the player and if the relation does not exist yet, it is initiated with the level of the player. For the moment, there is no real collaboration between players in the game as the participants do not know what the others do. However, they can question the answers which are given to them (in case of reference sentences) and we (the managers of the game) try to answer all of them (or to correct the reference, when the players are right). We plan to add more interactions in future versions of `ZombiLingo`, including a forum (attached to annotations, so that players can interact and correct themselves), a chat room and a challenge mode (where one player can challenge another on some sentences).

The quality of the obtained annotation, as evaluated on the reference sen-
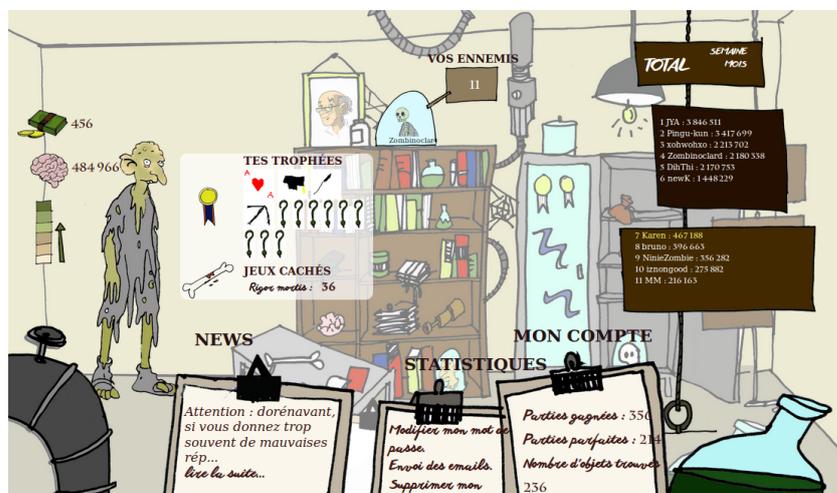
Figure 3.12: `ZombiLingo`: a full-fledged game for dependency syntax annotation (player's page).

tences, is very high (see Section 3.2.3) and even if we think it is a bit overrated, as reference sentences do contain the phenomenon, whereas the others do not, we think that this first version of `ZombiLingo` shows that it is possible to produce high quality complex annotations with a game. We are aware that there is room for improvement, but the first results are very promising from the quality point of view.

As for the produced quantity, it is simply astonishing! We obtained nearly 50,000 annotations in one month, produced by a total of approximately 300 players, with little publicity and a still Beta version of the game. As of end January 2016, we reached more than 90,000 annotations by 580 players.

Those examples show that there is virtually no end in sight in what we can train people to do if we manage to make it funny, while keeping the virtuous circle of quality in mind. The limits are mainly that of our imagination and of our capacity to design games and to obtain the necessary technical means.

3.4   Acknowledging Crowdsourcing Specifics

3.4.1   Motivating the Participants

A vital issue in crowdsourcing in general is to get a sufficient number of people to try and play or work, in order to gather a subgroup of highly productive participants. Motivation is therefore a key question, which is about convincing people not only to participate (motivation), but also to come back and participate again (volition [Fenouillet et al., 2009]).

There exists several theories of motivation in the field of gamification, including the *GameFlow* theory [Sweetser and Wyeth, 2005], derived from the
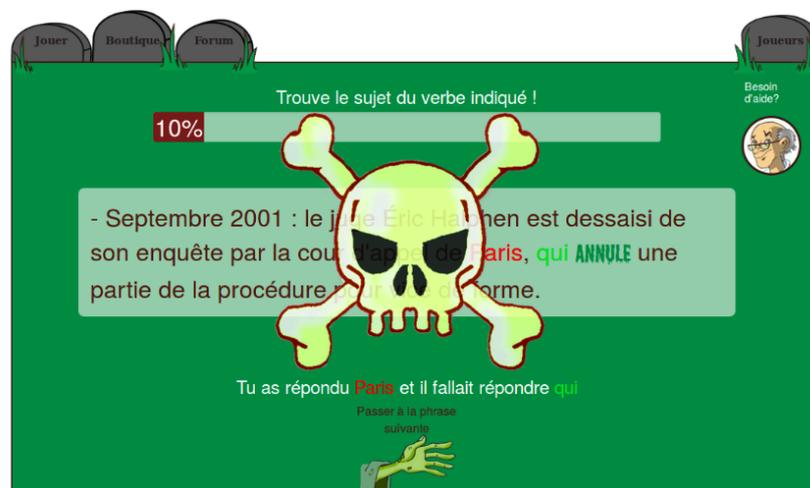
Figure 3.13: `ZombiLingo`: tutorial interface (correction of an error).

more general *Flow* theory, which dates from 1975 and is presented in details in [Nakamura and Csikszentmihalyi, 2002]. Mathieu Lafourcade used a much simpler and general grid for `JeuxDeMots`[34], the CIA *MICE* framework for agents recruitment[35]: *Money*, *Ideology*, *Coercion*, *Ego*. Obviously, in crowdsourcing, the terms have to be adapted, and *Money* can also be reward (in games), *Ideology* can be considered in a wider, less political sense of interest (participating to science, for example), *Coercion* should obviously remain light (like a loss of points if the player does not come back when s/he is challenged by another player, or being forced to retrain after a number of errors), only *Ego* will not need to be adapted. This "grid" of analysis is a bit simplistic (as expressed in the CIA document, which presents a more complex model, *RASCALS*), but it represents an interesting first attempt at understanding the potential motivations of the players in the crowd.

Although *Ideology* is probably too strong a word to characterize it, people participating in citizen science do have an interest in helping researchers. Beyond that, participants in platforms like `Wikipedia` are encyclopaedians and as such they participate to the transmission of knowledge [Kittur et al., 2007], which can be considered as an ideal (if not ideology). Needless to say that there is not only one motivating factor for the participation to such a task and that *Ego* is probably a powerful trigger too.

Contrary to what is written in some papers, in particular in [Snow et al., 2008], social science studies have shown that workers in microworking platforms like `Amazon Mechanical Turk` are mainly motivated by the money they earn and

---

[34]Personal communication, November 6th, 2013.
[35]See "Burkett-MICE to RASCALS.pdf" in `https://www.cia.gov/library/center-for-the-study-of-intelligence/csi-publications/csi-studies/studies/vol.-57-no.-1-a/vol.-57-no.-1-a-pdfs`.

that it is not a hobby for them [Ipeirotis, 2010a]. However, some difficulties
encountered by researchers on the platform show that there are not enough
Turkers. For example, the usual very low remuneration has proven not to be
enough reward in some cases where the number of potential *Turkers* for a spe-
cific task is low: for the transcription of Korean, wages had to be increased from
$5 to $35 per hour to get the task done [Novotney and Callison-Burch, 2010].
Also, very large tasks can prove difficult to perform [Ipeirotis, 2010b]. Besides,
the task payment logic induces a strong bias in favor of quantity rather than
quality.

   While most GWAPs do not propose any financial remuneration, some of
them, including `Phrase Detectives`, offer an indirect financial incentive through
regular lotteries (where each annotation has a chance to win) and prizes for
high scoring players [Chamberlain et al., 2013]. This mechanism is quite dif-
ferent from the microworking logic: the players are not remunerated for each
task they perform. Instead, the randomness of the lotteries generates motiva-
tion even for players who do not play a lot. More importantly, remuneration
is not placed at the heart of the system like in `Amazon Mechanical Turk` (see
Figure 3.14), and enjoyment (the *Flow* [Nakamura and Csikszentmihalyi, 2002])
remains the main incentive for players to come and play.
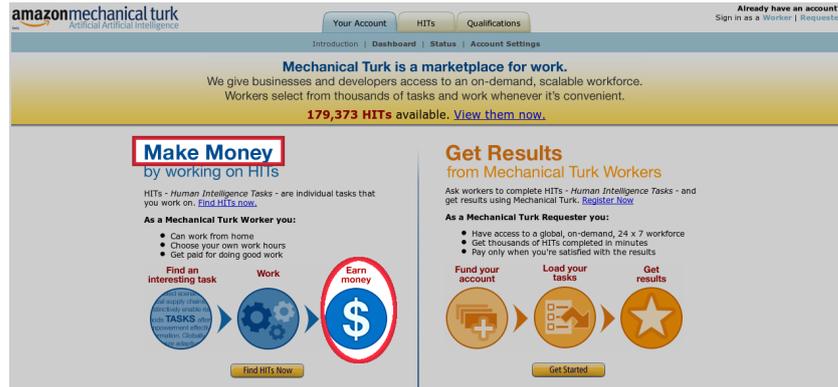
Figure 3.14: Homepage of `Amazon Mechanical Turk`: remuneration is at the
heart of the platform.

   However, the substance of enjoyment is difficult to capture with the *MICE*
grid of analysis, which seems a bit too coarse for that. A first step into under-
standing it is to try to classify the different types of participants. A well-known
typology in the field of game design is that of Richard Bartle [Bartle, 1996]. Bar-
tle defined four main types of players (*Achievers*, *Killers*, *Socialisers*, *Explorers*),
according to their interest in the game: *acting on* or *acting with* (interacting),
with an emphasis put on the game itself (world) or the players (see Figure 3.15).

   Achievers are interested in acting on the game, mastering it and its rules so
well that they gain points more quickly than others. They represent the type
of players everybody usually thinks of. The three other types are less obvious.
For example, *Killers* are players who want to act on the other players, usually
by attacking them. *Socialisers*, on the other hand, wish to interact with others,
getting to know them and sharing tips with them. Finally, *Explorers* want to
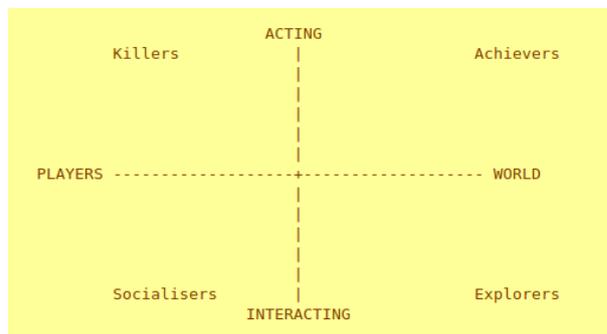
Figure 3.15: Bartle's Interest Graph [Marczewski, 2015].

discover what is hidden in the game.

This inspired a more complex typology, which is presented on the `Gamified UK` Website, the Marczewski's Player and User Types Hexad[36]. This model proposes four main types, *Achievers*, *Socialisers*, *Philanthropists* and *Free Spirits*, motivated by *Relatedness*, *Autonomy*, *Mastery* and *Purpose* (see Figure 3.16). The first two types correspond more or less to Bartle's equivalent, while the *Free Spirits* category covers a bit more than the *Explorers* as *Free Spirits* want to explore or create, and *Philanthropists* are close to *Socialisers* while being more interested in the purpose of the game. The typology also contains two less positive types: *Players* (motivated by rewards) which are close to *Achievers* (motivated by mastery) and *Disruptors* (motivated by change), which are close both to *Killers* and *Explorers*.
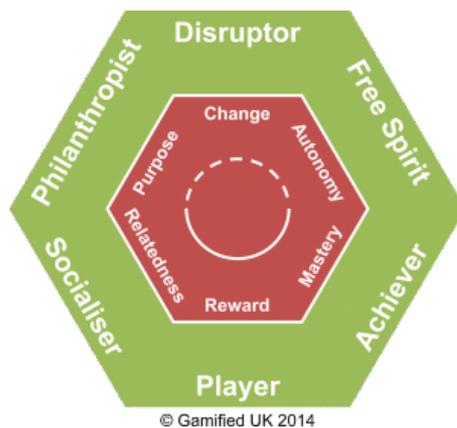
Figure 3.16: Marczewski's Player and User Types and Motivations Hexad, by courtesy of the author [Marczewski, 2015].

---

[36]See `http://www.gamified.uk/user-types/`.

Elements of gamification are added to the typology (see Figure 3.17). To attract a lot of participants, a "good" game or gamified interface should satisfy each player type. Obviously, these typologies are not exhaustive and some players may be attracted by parts of the game that were not associated to motivating features. A good example is the phenomena page in `ZombiLingo`, in which the player can see, for each phenomenon, the number of played sentences and the number of remaining sentences: for some players (whom we could call *Collectors*), "finishing" a phenomenon is so satisfying that it makes them play more. We decided to take it into account and to find ways to improve the interface accordingly (changing the color of a finished phenomenon and putting it in a reserved place).

The best way to improve a GWAP is to play it and to take the players' remarks into account. Maintenance is therefore a key issue.
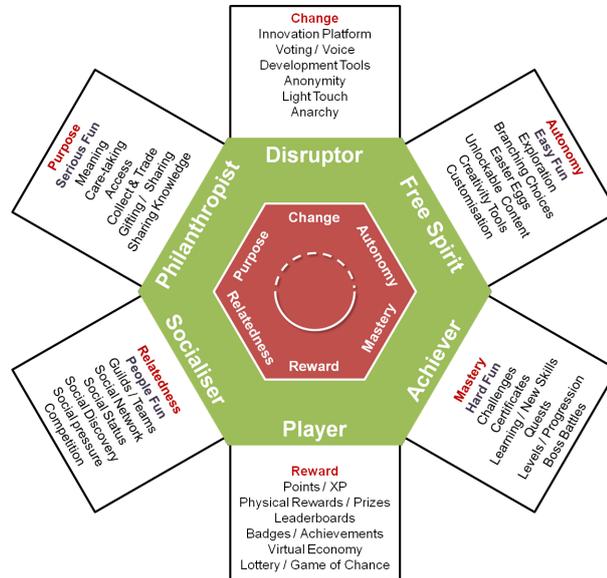
Figure 3.17: Gamification elements according to player types, by courtesy of the author [Marczewski, 2015].

Attracting participants is the most important issue in crowdsourcing, as only a small proportion will contribute a lot to the data creation (see Section 3.2.2). Advertising requires time and access to media, which is not always easy to obtain for researchers.

3.4.2 Producing Quality Data

Adding motivating game features should not impact the produced quality, on the contrary, it should have a positive effect on the quality. A player should score points only when s/he creates quality data (see Figure 3.18). However, preserving this virtuous circle is not always straightforward. Let us illustrate this with two examples from real games experiments.

Figure 3.18: GWAP virtuous circle: the players should gain points only when they produce quality data.

In order to add fun to the game, we decided to add a feature to `ZombiLingo`: sentences disappearing randomly while playing. The players were surprised and this really added fun to the game, but when the sentence began disappearing, they tended to click at random, therefore creating a bad quality annotation. We quickly fixed this, deactivating the mouse click when the sentence started disappearing.

Another interesting anecdote concerns `JeuxDeMots`[37]. A player found a hack in the game to get more time to play (the game includes a timer). This player thus managed to create more data, quality data, which was good for the language resource creation perspective. However, this generated envy and anger in the community of players and was therefore bad for the game.

More importantly, to be able to identify the degradation in quality we have to be able to evaluate this quality. Like in any other form of (non-crowdsourced) annotation, the evaluation means will depend on whether there is a reference or not.

If a reference exists, like in the case of `Phrase Detectives` or `ZombiLingo`, it can be used to evaluate the annotations produced by the participants, simply by comparing the produced result with the reference. This can (and should) be done on several occasions, not only during the training phase, to check that the players are still annotating correctly. In these GWAPs, players have to go through a training phase, during which all of their annotations are compared to the reference and feedback is given to them so that they understand and correct their errors (see for example Figure 3.13). They can progress in the tutorial and get to annotate non reference sentences only if the results are correct enough, that is if they gave ten correct answers per phenomenon on `ZombiLingo` (indicated by a progress bar) or if 50% of their answers are correct in `Phrase Detectives` [Chamberlain et al., 2013]. While playing, they will occasionally run into more reference sentences (without them knowing it) and receive feedback if their annotations are inconsistent with the reference ones. In `ZombiLingo`, the more wrong answers are given, the more likely the player will be exposed to reference sentences. This way, a very bad player will only play reference sentences and will either get better at it, or quit the game. The results of this evaluation can also be used to assign a confidence ranking to the players

---

[37]Mathieu Lafourcade, personal communication, July 11th, 2015.

and to give them more "power" in the game, for example by trusting their annotations more or by giving them access to specific features, like challenges or hidden games.

If there is no reference available, the evaluation of the produced data should rely on other procedures. In the case of `JeuxDeMots` (which is strictly speaking not about annotating), M. Lafourcade designed a "counter" game, `TOTAKI`[38], to help validating the created lexical network [Lafourcade and Joubert, 2015]. The production of a GWAP can therefore be evaluated, at least partly, with another GWAP.

In the case of annotation, there is almost always a reference of some kind, at least a mini-reference (see Section 2.1). However, this reference cannot help us evaluating new annotations. We therefore need to use any evidence at our disposal. One of them is the level of confidence we have in the player, computed using the reference sentences (either during the training phase or after). Another one is the agreement between participants (vote), computed in a more or less complex way. In `ZombiLingo`, we decided to use a mix of the two and to give more importance to a vote from a player we trust more (see Section 3.3.3).

3.5   Ethical Issues

3.5.1   Game Ethics

A well-known issue with games in general is that of addiction. It is however difficult to identify with on-line games, as, even if we track the time spent by each player on the game, it only corresponds to the time between the player's connection and logout, which is unrevealing, as first most players will not logout, and second we cannot know for sure they have been playing the whole time. Some games record the player's activity (directly or indirectly, like in `JeuxDeMots`, where the game is timed), but if there is no activity (mouse click or keyboard hit) it does not mean the player is not reflecting about an action in the game.

In addition, the limit between the temporary obsession with a game that most players have experienced and real addiction is difficult to identify. Although there is no real consensus over the definition, video game addiction seems to be revealed by three factors: (i) withdrawal symptoms, (ii) loss of control over gaming, and (iii) conflict [King et al., 2013]. As GWAPs creators, we should be aware of the dangers of creating a game that is so motivating that people can get unhealthily addicted to it. The role we can play in preventing players from falling into addiction is limited, but we can at least give information about video game addiction and provide contact information for a national hotline like *Joueurs Info Service* in France[39].

GWAPs for research purpose are citizen science platforms. As such, transparency should be at the heart of the process and players should be associated to the project: information about it should be given, with links to publications and contact information of the researchers. If relevant, they can (should) be associated to publications. Also, what comes from the crowd should go to the crowd: linguistic resources created using GWAPs should be

---

[38]See `http://www.jeuxdemots.org/AKI2.php?thema=-1`.
[39]See: `http://www.joueurs-info-service.fr/`.

freely and directly (on the game website) available, at least for research purposes [Lafourcade and Lebrun, 2014]. This means that the source corpora should be freely available too.

Finally, the personal data gathered from the players, if any, should be treated with respect. They should never be given away and the collect should obviously be limited to what is necessary for the game. If need be, the players who gave their email address can be contacted for further research questions, for example gender-related questions.

Information about all this should be presented in the game charter[40], in which game-related rules can be added (for example the prohibition to sell assets gathered in the game).

3.5.2 What's Wrong With `Amazon Mechanical Turk`?

`Amazon Mechanical Turk` is a microworking platform, massively used in NLP, which clearly puts the emphasis on remuneration on its homepage (see Figure 3.14). We already saw in Section 3.4.1 that the most productive workers on this platform consider it as an important source of revenue. However, the proposed (micro)remunerations on `Amazon Mechanical Turk` are very low (we explain why in [Fort et al., 2011a]), US$1.25 an hour in average according to [Ross et al., 2010]. In most developed countries, this would be considered as exploitation.

This is even more so as the Requesters (people proposing tasks) can simply refuse to pay the *Turkers* if they are not happy with the work performed, without even having to justify it (see Figure 3.19). This is a source of stress on the *Turkers'* side, which leads them to train on some tasks before officially (being registered) working on them, thus generating hidden unpaid work. This process is detailed in [Gupta et al., 2014], which also shows the constraints the *Turkers* impose themselves in order to get interesting HITs, in particular sleep deprivation due to the time difference (most of the Requesters are from the US, while the *Turkers* studied in this article are located in India).

Besides, `Amazon Mechanical Turk` does not provide any means to equivalently ban *Requesters* responsible for wrongdoings[41]. The work relationship is therefore completely unbalanced. In addition, there is no direct link between *Turkers* and *Requesters*, as only `Amazon` has information about who is behind which pseudo. Moreover, *Turkers* are considered as independent workers, not employees. As such, they are supposed to pay taxes on their remunerations, which is doubtful given the very low amounts they earn. Due to the dilution of work, microworking crowdsourcing deprives states of income.

Some researchers in the NLP community reacted in favor of the *Turkers*. Apart from our efforts in explaining what AMT really is [Fort et al., 2011a, Fort et al., 2014a], others tried to formalized good practices [Adda et al., 2013]. Finally, Chris Callison-Burch created a tool for them to be able to find higher paying HITs based on the hourly rate [Callison-Burch, 2014].

---

[40]See `ZombiLingo`'s charter (in French) here (login first): `http://zombilingo.org/compte/accueilCoIns#charte`.

[41]The *Turkers* created tools like `Turkopticon` to try and counter this: `https://turkopticon.ucsd.edu/login`.
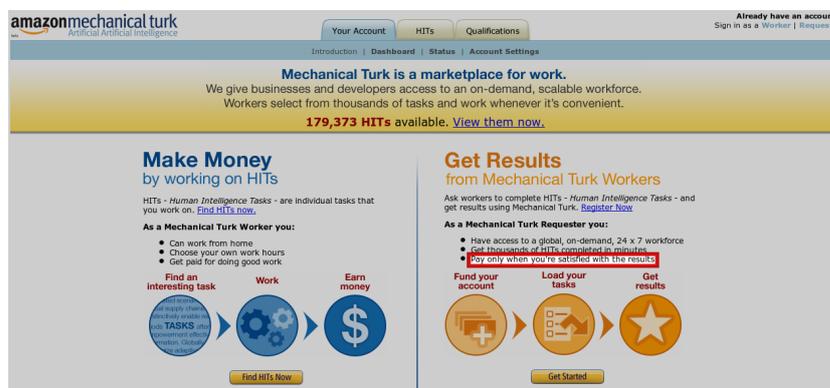
Figure 3.19: `Amazon Mechanical Turk`: "Pay only when you're satisfied with the results".

3.5.3   A Charter to Rule Them All

The Ethics and Big Data Charter was imagined as a tool to provide funding agencies with the necessary information they need to be able to make ethical choices when selecting projects. Another goal of this charter is to provide this information to all the potential users of a (language) resource. It was designed in collaboration with representatives from interest groups, private companies and academic organizations, including $ELDA$[42], the French $CNRS$[43], $ATALA$[44], $AFCP$[45] and $APROGED$[46].

    We demonstrated the need for such a document in [Couillault et al., 2014]. We showed, using the LRE map [Calzolari et al., 2012] as our main source, that if the providers of language resources usually give details on the nature of the data and the persons to contact, they are not always explicit about the license (it is provided in only half of the cases) and even less so (a little more than 30%) about the status of the persons who created the resource (and whether it was through crowdsourcing or not) or on the quality assurance of the production.

    The Ethics and Big Data Charter aims at addressing all these points in a single document, that, for practical reasons, is self-administered. It is split into three major sections dealing with: (i) traceability, (ii) intellectual property and (iii) specific legal issues. The first part gives details on the participants of the project, their status and remuneration and the selection criteria. It also describes if and how the data was de-identified as well as the quality assurance

---

[42]Evaluations and Language resources Distribution Agency, `http://www.elda.org/`.

[43]Centre National de la Recherche Scientifique/National agency for scientific research `http://www.cnrs.fr/`.

[44]Association pour le Traitement Automatique des Langues/Natural Language Processing Association `http://www.atala.org`.

[45]Association Française de Communication Parlée/French spoken communication association, `http://www.afcp-parole.org`.

[46]Association de la Maîtrise et de la Valorisation des contenus/Association for mastering and empowering content, `http://www.aproged.org`.

procedures that were used, if any. The second part deals with license and copyright issues, in particular detailing cases where the language resource is created from one or several existing resource: for example in the case of the *TCOF-POS* corpus [Benzitoun et al., 2012], which was built upon the *TCOF* corpus. The third part is intended for country-specific legal requirements to be explained.

The charter is available as a pdf document[47] and will soon be provided as a Web form to be filled online. Our hope is that it would help favor projects implying participants who are treated fairly. For the moment, only the French funding agency *Cap Digital* included the Ethics and Big Data Charter in its grants proposal.

---

[47]See, for English: `http://wiki.ethique-big-data.org/chartes/charteethiqueenV2.pdf`.

Chapter 4

Conclusion

Manual corpus annotation is now at the heart of natural language processing. It is where linguistics hides, in a still ill-explored way. There is a blatant need for manual annotation engineering (in the sense of a precisely formalized process) and this book aims at providing a first step towards a holistic methodology, with a global view on annotation.

Annotating is interpreting and this interpretation has to come from a consensus, built collaboratively. From this point of view, annotation is by definition collaborative and represents a shift from a vision according to which one specialist (a linguist), can make irrevocable judgments. There are many ways in which this collaboration can be fostered during the annotation process and we detailed some of them in this book.

However, the annotation methodology is still little explored as such and many issues remain open. In particular, the annotation of relations and sets has to be analyzed more closely, as they represent a wide variety of situations and complexities. Also, better solutions still need to be found to evaluate them.

As we showed in the second part of the book, crowdsourcing is one of the most effective solutions to lower the cost of manual annotation. Games with a purpose generate astonishing results, especially for language data production, both in terms of quantity and quality. However, designing and developing games require some specific gamification skills and technical expertise that are not accessible to everybody. Moreover, the necessary communication and the management of the community of players are time consuming and a lot of researchers would rather avoid spending time on that.

Therefore, this solution can only be effective if a common platform is created, allowing researchers to generate small (for example voting) games using simple online forms, or more ambitious ones using the skills of specialized engineers. A community manager would take care of the community of players and of the communication concerning the games. Two platforms propose at least some of the necessary services: METADATAGAMES[1], which provides a gamified interface for collecting metadata and `Zooniverse`, a freely accessible citizen science platform[2], which is not gamified.

---

[1]See: `http://www.metadatagames.org/`.

[2]See: `https://www.zooniverse.org/`.

Crowdsourcing brings us closer to the people we work for as researchers. This highlights our responsibilities, as employers or partners, and should encourage us to be more ethical and transparent in our work.

Appendix A

(Some) Annotation Tools

Keeping track of all the annotation tools is a job for Sisyphus. The list presented here is therefore far from exhaustive. It also contains deprecated tools or tools which were used only once, because they present interesting features which we want to expose. Finally, we did our best to keep this list up-to-date, but some tools evolve so rapidly that it is difficult to keep track of changes, so if one of them seems interesting to you, do not hesitate to check by yourself on the provided Web site.[1]

We chose not to use the traditional comparative tabular presentation, as it does not allow for the representation the different underlying principles behind each tool. We prefer to detail each of them in its context of creation and usage.

The organization of the inventory we propose here relies on a typology that we deduced from our tests.[2] Thus, Section A.1 presents generic tools, some of which were born so whereas others evolved towards from a specific annotation task. The following section logically introduces task-oriented tools. The annotation platforms, which aim more directly at applying NLP tools on corpora, are presented in Section A.3. Finally, Section A.4 details annotation campaign management tools. Within each section the tools are presented in historical order, from the eldest to the newest.

A.1 Generic Tools

The majority of the tools presented here were not initially designed to be generic, but were created for a specific task and were extended to other annotation tasks. They do not provide annotation management capabilities as such, but some of them include features like inter-annotator agreement computation.

A.1.1 `Cadixe`

`Cadixe` [Alphonse et al., 2004] is a deprecated annotation tool (the last version dates from March 2005), that is no longer available and that we only list here because it was used in the gene renaming campaign we mention in this book.

---

[1] Do not hesitate to contact us to bring a tool, a new feature or an error to our attention.
[2] Note that most of these tests were done between 2010 and 2012. Since then, we have tried to keep track of the new systems but could not test all of them.

It provided a Java interface to add in-line XML tags to a corpus, from a DTD, with customized colors and styles, defined in a stylesheet (CSS, Cascading Style Sheet)

It was reasonably easy to use and documented, but with too limited features. In particular, the in-line format did not allow for the annotation of relations.

A.1.2   Callisto

Callisto [Day et al., 2004] is freeware written in Java. It is "no longer actively supported"[3], but its source code is still available. The last version (1.8) dates from 2013. It was initially developed by the *MITRE Corporation*, with funding from the US government.[4]

Callisto supports XML and generates standoff annotations following the ATLAS data model [Bird et al., 2000]. Interestingly, it allows importation and exportation of inline annotations.

Another interesting feature is that it relies on a modular and configurable architecture. It provides annotation services which are available to annotation modules. Some modules are already available, others can be developed. A module for temporal relations annotation was created (TANGO), but is no longer available.

Callisto is quite user-friendly if it is already configured for the planned annotation task (like anaphora resolution, for example), but it is not easy to adapt to a new task.

New and interesting management features were planned (time tracking, inter-annotator agreement computation, etc.), but the last version is not documented and it is unclear whether they were actually added.

A.1.3   Amazon Mechanical Turk

Amazon Mechanical Turk[5] is now a very well-known microworking platform. It was first created by *Amazon.com* for its internal needs. The firm opened the platform to other requesters in 2005.

Although it is not unique, this microworking platform is the most well-known and the most used, in particular in NLP. It allows *Requesters* to propose groups of simple tasks (*Human Intelligence Tasks*, or HITs) to be carried out by a large group of workers (*Turkers*). The real tasks are simplified so that they can be done by non-experts—for example, a "real" task related to recognizing textual entailment across several sentences that reflect entailment, neutrality, or contradiction might be simplified to the presentation of one pair of sentences and a binary response to a single question, such as "would most people say that if the first sentence is true, then the second sentence must be true?" [Bowman et al., 2015]. Execution of these simple tasks is paid for by the *Requesters*, provided that they are satisfied with the result. This is known as *artificial artificial intelligence*.

Amazon Mechanical Turk provides Web components that the *Requester* can use to present tasks to the Turkers in forms. A form contains one or more questions (explicit or implicit) to which the *Turker* must respond [Snow et al., 2008].

---

[3]https://github.com/mitre/callisto
[4]http://www.mitre.org/
[5]https://www.mturk.com

The "art" of the *Requester* resides in their ability to decompose complex tasks, in particular annotation tasks (see, among many others, [Snow et al., 2008]), into simple (even simplified [Bowman et al., 2015]) tasks, and to adjust the remuneration, which is usually very low (for example, US$0.15 to identify a relation between two entities in a sentence[6]).

This system allows the production of very low-cost language resources (1/10th of the usual cost, at least according to [Callison-Burch and Dredze, 2010]), which explains, at least in part, its huge success in NLP over the course of the past ten years.[7]

We showed the technical limits of this system in different position papers [Fort et al., 2011a, Adda et al., 2011], as well as in Section 3.5.2, so we will not elaborate on them here. Further details concerning `Amazon Mechanical Turk` can be found in Section 3.2.

Note that in some respects, `Amazon Mechanical Turk` is close to an annotation management tool: it differentiates between two roles (*Requester* and *Turker*); it provides some monitoring concerning the progression of the HITs, of the time spent on the tasks, and of the effective hourly pay rate; it distributes tasks to workers; and it manages the flow of the results. However, the scope of annotation management is much wider than the task management and includes versioning, evaluation tools, guidelines, corpus management, etc. (see Section A.4).

A.1.4 Knowtator

`Knowtator` [Ogren, 2006] [8] is free and open source Java plugin for the ontology editor `Protégé`[9], which was originally developed for biomedical annotation. Its latest version dates from 2010, but it is still used on a number of annotation projects (for example, the CRAFT corpus [Bretonnel Cohen et al., 2015]) involving an ontology.

Its backend code is directly inspired from that of `WordFreak` [Morton and LaCivita, 2003] and it allows for the standoff annotation of units and relations. It also offers a "consensus" mode allowing the comparison and merging of annotations from several annotators.

`Knowtator`'s annotations can be exported into any format supported by `Protégé`. However, it only accepts raw text as input and does not take into account existing XML tags. The tags are therefore visible and interfere with the text to annotate. Another limitation of this tool is that it is not compatible with the latest `Protégé` version.

A.1.5 MMAX2

`MMAX2` [Müller and Strube, 2006][10] is an open source Java program. It does not seem to be maintained anymore.

`MMAX2` provides an integrated tokenizer. It supports XML and the annotations are standoff, which allows for the annotation of relations. It distinguishes

---

[6]Found on February 10th, 2016 on the `Amazon Mechanical Turk` Web site.

[7]There are other reasons for this success, including the production speed, but most papers emphasize the cost.

[8]http://knowtator.sourceforge.net/

[9]http://protege.stanford.edu/

[10]http://mmax2.sourceforge.net/

between two types of relations: non-oriented relations (sets) and oriented re-
lations (*pointers*). It also allows to annotation discontinuities, partial overlaps
and factorization.

Moreover, it allows to add annotations from different levels in the same
project, like *phrase*, *chunk* and *POS*. Finally, it offers a *AnnotationDiffWindow*
plugin to visualize concurrent annotations of co-reference chains.

MMAX2 is therefore a richly featured tool, but it is difficult to use, due to
a complex multi-windows interface. Like many other annotation tools, it gets
really slow when the files are longer than a couple of pages. Finally, its docu-
mentation is limited.

A.1.6 UAM CorpusTool

UAM CorpusTool [O'Donnell, 2008] is an annotation tool originally designed for
linguists, which is easy to install and configure. It is a Python program, which
is only available for MS Windows and Mac OS. It is not open source, but it is
free. It is still maintained and a new version was released in November 2015.

UAM CorpusTool supports different linguistic annotation levels, the annota-
tions are stored in standoff XML format, with one file per annotation level.

It is user-friendly and allows to create annotation projects rapidly, through
an interface dedicated to the organization of the corpus files and the creation of
the annotation scheme. It allows to modify the annotation scheme during the
project and to add glosses to each note.

It provides a powerful search engine (using CQL), that supports lexical
patterns and searches through several levels of annotation. Automatic pre-
processing is also available (sentence segmentation, chunking, POS annotation,
syntactic taggins) as well as various statistical results.

However, the tool does not distinguish between roles (for example, manager
and annotators). It also suffers from important limitations: it is not possible to
annotate relations, no *Undo* command is available.

A.1.7 Glozz

Glozz [Widlöcher and Mathet, 2009] was created for the *Annodis* project, which
aimed at creating a reference French corpus for discourse analysis. It is a generic
tool, with specific features for discourse annotation (like a general view of the
text being annotated). It is a freely available (for research) Java program.[11]

Glozz takes text as input and provide standoff annotations in XML. Its
(point-and-click) interface is user-friendly and includes a customizable stylesheet
to parameterize tags colors. Interestingly, it offers a *glue note* mode to add free
comments. It also allows to hide some annotations (for example, POS when
annotating named entities).

Its generic meta model proposes three types of components: *units* (adjacent
textual elements), *relations* (binary relation between two units) and *schemas*
(complex textual configuration including units and relations). Pre-annotations
can be integrated, but they have to be standoff and to comply with this meta
model.

---

[11]http://www.glozz.org/

`Glozz` includes a powerful search engine allowing to search both the text and the annotations, with *GlozzQL*, a specific query language. It also provides the possibility to compute the inter-annotator agreement using γ [Mathet et al., 2015], which is very useful.

However, `Glozz` is still not very robust (like many annotation tools) and gets very slow with large files (more than a couple of hundred kilobytes).

A.1.8 CCASH

`CCASH` (*Cost-Conscious Annotation Supervised by Humans*) [Felt et al., 2010] is, as stated in its name, a cost-oriented annotation tool. It was designed to integrate automatic processing, like tag dictionary [Carmen et al., 2010], to limit the work of the annotators and to track the cost of annotation. Interestingly, this cost is not only evaluated through the time spent on the task, but also in terms of the number of interactions needed to perform an action.

It is a Web application, which can handle several annotators in parallel. It includes a limited administration interface, allowing an administrator to create annotation tasks and to assign them to users.

Annotations are standoff which allows for the annotation of relations. It is highly flexible and customizable by developers, as it is open source.[12] A minimal documentation is available[13] and the tool is still maintained and improved.

A.1.9 brat

`brat` (*brat rapid annotation too*l) [Stenetorp et al., 2012][14] was originally designed as an extension for an annotation visualization tool. It is now a very specific, fully collaborative annotation tool.

It is a well alive and maintained open source web application, with a server written in Python and a CGI interface. Annotations are standoff.

`brat` provides a user-friendly interface allowing to annotate both units and relations. An important feature of the tool is that each annotation is associated to a unique URL, which makes it one of the few tools supporting linked data.

Another specifics of `brat` is the fact that it is fully and directly collaborative: annotators can work on the same corpus or the same document simultaneously and visualize the modifications made by the others while they are added (this corresponds to the collaborative mode in `SYNC3`). In this tool, direct collaboration is not an option, it is the rule.

In the same spirit, the tool does not distinguish between roles and does not provide inter-annotator agreement metrics. However, it is possible to configure it to track the time spent on a document, or on each editing and categorizing action.

For the moment, it is fully supported only on `Chrome` and `Safari` Web browsers, but it seems to work fine on `Firefox` too.

---

[12]`http://sourceforge.net/projects/ccash/`
[13]`https://facwiki.cs.byu.edu/nlp/index.php/CCASH`
[14]See: `http://brat.nlplab.org/index.html`.

A.2   Task-oriented Tools

A.2.1   LDC Tools

The Linguistic Data Consortium (LDC)[15] has developed a number of tools to annotate speech and text, based on the Annotation Graph Toolkit (AGTK), which instantiates a formalism derived from the annotation graphs [Bird and Liberman, 2001]. These tools are all freely available and open source.[16]

`AGTK` is a set of software that can be used to develop annotation tools and which includes APIs allowing to manipulate annotation graphs as well as specialized graphical components.

These tools are detailed in [Maeda and Strassel, 2004] and include a transcription tool (`XTrans`), a tool dedicated to named entity annotation for the Automatic Content Extraction (ACE) campaign, tools allowing to compare and adjudicate concurrent annotations.

More recently, they developed a Web application (in PHP), the Annotation Collection Toolkit (ACK) [Maeda et al., 2006] for simple annotation campaigns.

The LDC chose to specialize tools and to use a common formalism. This choice allows for a great flexibility and adaptability to the needs of each campaign, but it implies to develop new components regularly.

A.2.2   `EasyRef`

`EasyRef` [de la Clergerie, 2008] is a collaborative syntactic annotation tool. It was designed for the PASSAGE project (the acronym means "producing large scale syntactic annotations") project, which planned the manual annotation of 500,000 French words to serve as reference for the systems participating to the project.

Since several teams from different locations were to use it, it was developed as a Web application. It still exists, is running and its main creator, Eric de la Clergerie, is planning to make it open source.

`EasyRef` [17] allows to:

- visualize the annotations,

- add and edit annotations while preserving a precise trace of the modifications: each edit action creates a new version of the sentence, with an incremented revision number,

- check the syntactic coherence of the annotation by applying constraints,

- manage bug reports related to a sentence, including with discussions concerning a specific bug,

- search the annotations with regular expressions and administrative criteria (presence/absence of a bug report for example).

The traceability and collaboration features offered by `EasyRef` make it a very specific, very original tool that should inspire the next generation of annotation tools.

---

[15]http://www.ldc.upenn.edu/
[16]See: https://www.ldc.upenn.edu/language-resources/tools.
[17]http://atoll.inria.fr/easyrefpub/login

A.2.3 `Phrase Detectives`

`Phrase Detectives` [Chamberlain et al., 2008][18] is a Web application designed for the manual annotation of anaphora. Its form is somewhat specific, as it is a gamified interface, a Game With A Purpose (GWAP, see Section 3.3). The application itself is not freely available.

The texts proposed for annotation are automatically pre-annotated with the following pipeline of tools [Kruschwitz et al., 2009]: normalization, tokenization, parsing (with the Berkeley parser) and identification of the markables.

On `Phrase Detectives`, the players (the detectives) have to go through a training phase, playing on reference sentences. When they make an error, a feedback is given to help them improve their understanding of the task. Once they reach a sufficient level, they can start playing, i.e. annotating for real. The game includes two modes, on in which they have to select the antecedent of the highlighted element in the text and another in which they have to validate annotations that were added by others.

Players gain points if other players agree with them. They are also regularly evaluated on the reference corpus. Several incentives are proposed to motivate players: a leaderboard, levels (and associated avatars), funny corpora (which the players can choose from) and vouchers for the best player and for a randomly picked annotation [Chamberlain et al., 2009b]. Although the vouchers cannot be considered as a remuneration *per se* and are not so much emphasized on the Web site, they still represent a step aside from citizen science.

`Phrase Detectives` allowed to annotate approximately 200,000 tokens, with a satisfying quality (see Section 3.3).

This type of tool could probably be adapted to other annotation tasks involving (at least) binary relations, but as we already said, it is not available.

A.2.4 `ZombiLingo`

`ZombiLingo` [Fort et al., 2014b][19] is a Game With A Purpose designed for dependency syntax annotation. It is a Web application written in PHP and it is meant to become open source as soon as the code is stabilized enough. It could probably be adapted to different sorts of binary relations annotations.

The game is presented in details in Section 3.3.3.

A.3 NLP Annotation Plaforms

The main advantage of these platforms is that they provide access to NLP tools, like tokenizers, POS taggers, chunkers or named entity extractors. These tools can be used for automatically pre-annotating the corpus. However, when these platforms support manual annotation, it is only a marginal feature, often added late (`GATE`) or still in development (`UIMA`).

---

[18]See and play here: `https://anawiki.essex.ac.uk/phrasedetectives/`.
[19]See and play here: `http://zombilingo.org`.

A.3.1   GATE

`GATE` [Cunningham et al., 2002] (*General Architecture for Text Engineering*) is
an open source platform developed by the University of Sheffield.[20] It provides
access to existing NLP tools and resources and to develop new tools. In order to
do this, `GATE` a manual annotation interface (`GATE Developer`). The platform is
maintained (version 8.1 was released in June 2015), documented, and dedicated
training and even certification, is provided.

Since the tool was primarily designed for automatic processing rather than
for manual annotation, the annotation interface remained poorly documented
until version 5. Dedicated efforts have been made to document the following
versions of the interface, but manual annotation in `GATE` remains marginal and
limited to the correction of automatic annotations. A simple test shows that it
is still impossible to undo an action. Also, the original text can (too) easily be
modified, which generates errors, in particular when comparing annotations.[21]

Still, `GATE` provides a wide choice of useful options, such as toggling a source
document between editable and uneditable states (*Read Only*). The annotator
can also create new categories for manual annotation (*Restricted/Unrestricted
annotation set*). The `ANNIE` module (a Nearly new Information extraction sys-
tem) proposes a number of useful automatic processing: sentence segmentation,
tokenization, POS tagging, named entity recognition (using gazeteers). Finally,
rules can be written with `JAPE` (Java Annotation Patterns Engine).

`GATE`'s features, while not always easy to handle, are varied and numerous.
For example, the annotator can automatically annotate all the occurrences of a
same term in a text(*Annotate all*); visualizing annotation and searching in the
text are easily performed; `GATE` includes an editor for co-reference chains.

Supported input formats include raw text as well as major markup languages
(XML, HTML). The annotations are standoff in the TIPSTER [Grishman, 1998]
format and are stored either as a relational database or as a file (XML or Java
serialization). In addition, `GATE` provides an API to the applications it proposes
and is embedded in `UIMA`.

`GATE` includes a nice interface to activate or not and to compare annotations
from different annotators (the `Annotation Diff` tool). The agreement can be
defined more or less leniently. However, the inter-annotator agreement mea-
surements is for the moment restricted to the coefficients of the kappa family
and the F-measure.

The `GATE` team has recently released `Teamware` [Bontcheva et al., 2010] for
the management of annotation campaigns which is further presented in sec-
tion A.4.

A.3.2   EULIA

`EULIA` [Artola et al., 2004] is a Web-based collaborative annotation tool dedi-
cated to linguistic analysis. It inputs and outputs XML TEI documents and
manipulates feature structures. The annotations are standoff. In addition to
providing an environment for annotation, `EULIA` includes a search engine and
also NLP tools for tokenizing, POS tagging, and chunking.

---

[20]See: `http://gate.ac.uk/`.
[21]This, among others, is reported here: `http://mpqa.cs.pitt.edu/annotation/gate_basics/`.

The annotation of relations is apparently not possible. `EULIA` is not open source and is apparently no longer maintained.

A.3.3 `UIMA`

The goal of the *Apache* `UIMA` [22] (*Unstructured Information Management Applications*) project is to support the development of an architecture and applications for the processing of unstructured (text) information. Initially developed by *IBM* [Ferrucci and Lally, 2004], `Apache UIMA` is embedded in the `Eclipse` IDE.

The effort required to install the environment is compensated by the large number of off the shelf *annotators* (automatic annotation tools) including a space-based tokenizer, a stemmer (`Snowball`), a HMM-based POS tagger, a regular expressions based annotation tool, a named entity extractor (`OpenCalais`), and a dictionary compiler to identify predefined text segments. Any other available tool can be installed.

The `CAS Editor` provides basic features for the manual annotation of texts. The annotation of relations can not be performed with its interface. It is somewhat complex to install, even though it is provided as an `Eclipse` plugin since `UIMA` version 2.3.0. As far as we know, the `CAS Editor` has not been yet used for a full-fledge manual corpus annotation campaign.

A.3.4 `SYNC3`

`SYNC3` [Petasis, 2012] is designed to support collaborative annotation. This open source software is freely available under a LGPL licence, as part of the `Ellogon`[23] NLP platform.

`SYNC3`, though it is a collaborative tool, is not a Web application but a desktop application which synchronizes with a central server and stores documents and metadata in a database. This original approach is, according to the designer, justified as it caters for a richer user interface (possibility to define keyboard shortcuts or launch local treatments, advanced customization...) while allowing annotators to work offline.

`SYNC3` exposes `Ellogon` features such as computation of inter-annotator agreement, but does not provide any specific user interface to manage annotation or user rights.

Interestingly, the tool induces customizable regular expressions from recorded annotations to automatically annotate text segments.

`SYNC3` supports two annotation modes: distributed and collaborative. In the former, annotators work on on different versions of the document while in the latter, all the annotators work on the same document, in a way similar to `brat` (see Section A.1.9). In the "collaborative" mode, all the annotations are stored.

A.4 Annotation Management Tools

There are only few real annotation management tools. They are more or less available and their features can differ, but they have in common to be much

---

[22]See: `http://uima.apache.org/`.
[23]`http://www.ellogon.org/`

more than an annotation interface and to support the whole campaign process
and not only the annotation.

A.4.1 `Slate`

`Slate` [Kaplan et al., 2010, Kaplan et al., 2012] (*Segment and Link-based An-
notation Tool Enhanced*)[24], is a very complete annotation management tool. It
is now open source and freely available[25] It is a Web application, with a Java
server and an annotation interface in Flash. `Slate` is therefore very visual, very
pleasing to use for an annotator, but requires the Flash player. It supports
`Chrome` and `Safari`, but not Firefox (like `brat`).

In `Slate` there are two types of users, the administrators and the annotators.
The administrator can create annotation projects, manage the users, assign an-
notation tasks (and remove them), monitor the progression of the work, import
and export data. The administrator is provided with an interface highlight-
ing the differences between parallel annotations and allowing to merge them to
create a reference.

`Slate` offers the possibility to version the annotation project, much like
a standard development project. All the annotation instances are identified
with the version number of the project, hence of the tagset, at the time of its
annotation. From our point of view, this is essential. Note that this system is
different from that implemented in `brat`, as the versioning allows to link the
tagset (and the guidelines) to the annotations.

Interestingly, in `Slate` the administrator cannot annotate. As for the anno-
tators they cannot see the work carried on by others. The two roles are therefore
completely independent.

In `Slate`, an annotation is a note, added either on a span of text, or on a
relation between spans of text (*links*). The relations can be oriented or not.
The schema creation is flexible and multiple annotations can be added on a text
span with any number of name/value attributes.

`Slate` generalizes a powerful feature of `EasyRef`: it allows to define con-
straints on the annotation of segments and relations.

It also allows for the addition of new annotation layers, with new tagsets,
referencing the previous layers. The created annotations are obviously standoff.

However, `Slate` does not provide any way of computing the inter-annotator
agreement or to see the time spent by annotators on their task.

A.4.2 `Djangology`

`Djangology` [Apostolova et al., 2010] is a Web-based annotation management
tool written in Python using the Django framework.[26] It is intended to be
easy to install and to allow for distributed and collaborative annotation of text
documents. It is freely available and open source [27] and seems little maintained
(the last modification in the source code dates from 2010. The documentation
is very limited too.

---

[24]See: `http://www.cl.cs.titech.ac.jp/slate`.
[25]The source code can be found here: `https://bitbucket.org/dainkaplan/slate/
overview`.
[26]See: `https://www.djangoproject.com/`.
[27]See: `http://sourceforge.net/projects/djangology/`.

From the annotation management point of view, `Djangology` provides an administration interface allowing to manage documents and users and to define the data model for the annotation. It also proposes an interface to compare parallel annotations and to compute basic inter-annotator agreement metrics.

As an annotation tool, `Djangology` offers basic annotation capabilities, including the automatic propagation of annotations (the equivalent of *Annotate all* in `GATE`). It does not seem to allow to annotate relations.

A.4.3   `GATE Teamware`

`GATE Teamware` [Bontcheva et al., 2010] is an extension of `GATE` (see Section A.3.1) designed for organizing and performing online "collaborative" annotation. Like `GATE`, `Teamware` is open source (under an Affero GPL license) and freely available.[28]

The architecture of the tool is described in [Bontcheva et al., 2010]. It distinguishes between three processing layers: SOAP services for data storage, Web interfaces and an execution layer managing the flow of annotation projects. Interestingly, like in `SYNC3`, the annotation tool is a locally installed software, while the Web interface is used for campaign management.

Like `Slate`, `Teamware` makes an explicit distinction between the various roles of an annotation campaign. However, `Slate` acknowledges two types of actors, the campaign manager and the annotators, where `Teamware` supports three: *annotation managers*, *editors* or curators (experts) and *annotators*.

Curators measure inter-annotator agreements, carry out the adjudication, are in charge of the training of the annotators and answer their questions. The campaign managers define new annotation projects, monitor their progression, participate in the methodological choices (inter-annotator agreement computation, use of pre-annotation tools, etc.). Annotators are "non-experts" who have to be trained at the beginning of the campaign. The communication between the annotators and the editors is supported through instant messages. This organization is close to the one we propose (see Section 2.1).

`Teamware` offers the same automatic annotation services as `GATE` and its annotation interface is more or less the same. Similarly to `Knowtator` (see Section A.1.4), `Teamware` supports the use of an ontology for annotation. An adjudication interface is provided to the editors, which allows them to select the annotations the best annotations among the ones proposed by the annotators. Finally, the campaign manager interface offers the possibility to create corpora, to define the annotation scheme, to configure pre-annotations and to execute them on the corpora. It also allows to monitor the campaign: progression (number of annotated documents, remaining to annotate, being annotated) and statistics concerning the annotators (time spent on a document, effective working time, inter-annotator agreement, etc).

The application itself and the user rights are managed *via* two roles, *administrator* and *super-administrator*.

This tool is very complete, but it presents the same usability issues as `GATE`. Finally, as for all the other Web-based annotation tools, it does not support Firefox (in principle).

---

[28]`https://gate.svn.sourceforge.net/svnroot/gate/teamware/trunk/`

A.4.4 `WebAnno`

`WebAnno` [de Castilho et al., 2014] is an annotation management tool that uses `brat` visualization and edition engine. It therefore provides the same annotation interface and is not supposed to work on `Firefox` (though it seems to). It supports `Chrome` and `Safari`. It is open source and freely available under the Apache Software License (ASL) version 2.[29] It proposes a test mode, with an embedded database, that is very useful for testing purposes.

   `WebAnno` distinguishes between three roles, which are very similar to that of `Teamware`: *project manager*, *annotators* and *curators*, with the corresponding interfaces. Unlike in `brat`, annotators cannot directly collaborate on a document, as they work separately.

   It supports the annotation of relations and chains (in particular, coreference chains) and provides basic sentence segmentation and tokenization. It also offers the possibility to lock/unlock the tagset. Interestingly, it provides a left-to-right/right-to-left option, in order to be able to annotate right-to-left languages like Arabic. It supports a large variety of import and export formats.

   `WebAnno` uses the `DKPro Statistics` library to compute inter-annotator agreement metrics. By default, it provides Cohen's Kappa, but the upcoming version 3 should include multiple metrics and parameters, as well an export in CSV format.[30]

A.5   (Many) Other Tools

Many other annotation tools are available, but most of them are not usable (for different reasons) or no more available.

   `PALinkA`) [Orasan, 2003] seems to have disappear. The same goes for `Serengeti` [Stührenberg et al., 2007]. It seems that `CSAF` [Kim and Wang, 2012] does not exist anymore. `TextAE` is an "embeddable visual editor of text annotation"[31]. It is usually coupled with the `PubAnnotation` repository.[32] It was designed for community-based annotation in the biomedical domain, it is open source and freely available.

   As for `Annotatornia` [Przepiórkowski and Murzynowski, 2009], if it is still available[33] and is open source under a Mozilla Public License, it does not seem to be maintained anymore. Moreover, its interface and documentation only exist in Polish. To our knowledge, it is the only annotation tool written in Ruby.

   `WordFreak` [Morton and LaCivita, 2003] is an open source Java based annotation tool, available under the Mozilla Public License[34] It is no more maintained but it inspired a number of current annotation tools.

   `ANALEC` [Landragin et al., 2012], is a very interesting and original Java tool, which is compatible with `Glozz`. It offers specific visualization features and geometric analyses. It is still maintained, but is only provided and documented in French.[35]

---

[29]See: `https://webanno.github.io/webanno/`.
[30]Richard Eckart de Castilho, personal communication, February 4th, 2016.
[31]See: `http://textae.pubannotation.org/`.
[32]See: `http://pubannotation.org/`.
[33]Here: `http://zil.ipipan.waw.pl/Anotatornia`.
[34]Here: `http://wordfreak.sourceforge.net`.
[35]See: `http://www.lattice.cnrs.fr/Telecharger-Analec`.

`Egas` [Campos et al., 2014] proposes interesting features as an annotation management tool, but it is provided only as a Web service.

Glossary

Annotation annotation covers both the process of adding a note on a source signal and the whole set of notes or each note that results from this process, without *a priori* presuming what would be the nature of the source (text, video, images, etc), the semantic content of the note (numbered note, value chosen in a reference list or free text), its position (global or local), or its objective (evaluation, characterization, simple comment).

Crowdsourcing "the act of outsourcing tasks, traditionally performed by an employee or contractor, to an undefined, large group of people or community (a crowd), through an open call" [Wikipedia, December 2, 2010].

Game With A Purpose an entertaining setting – a gamified interface, or a full-fledged game – in which voluntary participants produce data that require human knowledge and learning capabilities.

Markables segments of the source signal that could be annotated: all of the tokens for POS annotation, but only the noun phrases for named entity annotation, for example.

Microworking (micro) remunerated crowdsourcing, in which the tasks are decomposed and simplified so as to be carried out by workers, (micro) paid by the (micro) task.

Prevalence a category is prevalent when it is dominant, i.e. when it is more used than the others, without the proportion being very precisely defined.

Requester person (or group of persons) proposing tasks on `Amazon Mechanical Turk`.

Turker worker on `Amazon Mechanical Turk`.

Acronyms

ACE Automatic Content Extraction

ACK Annotation Collection Toolkit

ACL Association for Computational Linguistics

AGTK Annotation Graph Toolkit

API Application Programming Interface

ATALA Association pour le Traitement Automatique des LAngues

EAT Elementary Annotation Task

GWAP Game With A Purpose

HIT `Amazon Mechanical Turk` Human Intelligence Task

LDC Linguistic Data Consortium

NLP Natural Language Processing

POS part-of-speech

Bibliography

[Adda et al., 2013] Adda, G., Mariani, J.-J., Besacier, L., and Gelas, H. (2013). Economic and ethical background of crowdsourcing for speech. In *Crowdsourcing for Speech Processing: Applications to Data Collection, Transcription and Assessment*, pages 303–334. Wiley.

[Adda et al., 2011] Adda, G., Sagot, B., Fort, K., and Mariani, J. (2011). Crowdsourcing for language resource development: Critical analysis of Amazon Mechanical Turk overpowering use. In *Language and Technology Conference (LTC)*, Poznań, Pologne. 5 pages.

[Akrich and Boullier, 1991] Akrich, M. and Boullier, D. (1991). *Savoir faire et pouvoir transmettre*, chapter Le mode d'emploi, genèse, forme et usage, pages 113–131. éd. de la MSH (collection Ethnologie de la France, Cahier 6).

[Alex et al., 2010] Alex, B., Grover, C., Shen, R., and Kabadjov, M. (2010). Agile corpus annotation in practice: An overview of manual and automatic annotation of CVs. In *Fourth Linguistic Annotation Workshop (LAW)*, pages 29–37, Uppsala, Sweden. Association for Computational Linguistics.

[Allauzen and Bonneau-Maynard, 2008] Allauzen, A. and Bonneau-Maynard, H. (2008). Training and evaluation of pos taggers on the french multitag corpus. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, B. M. J. M. J. O. S. P. D. T., editor, *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco. European Language Resources Association (ELRA). http://www.lrec-conf.org/proceedings/lrec2008/.

[Alphonse et al., 2004] Alphonse, E., Aubin, S., Bessières, P., Bisson, G., Hamon, T., Laguarigue, S., Nazarenko, A., Manine, A.-P., Nédellec, C., Vetah, M. O. A., Poibeau, T., and Weissenbacher, D. (2004). Event-based information extraction for the biomedical the CADERIGE project. In *JNLPBA COLING 2004 Workshop*, Geneva, Switzerland.

[Apostolova et al., 2010] Apostolova, E., Neilan, S., An, G., Tomuro, N., and Lytinen, S. (2010). Djangology: A light-weight web-based tool for distributed collaborative text annotation. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, B. M. J. M. J. O. S. P. M. R. D. T., editor, *International Conference on Language Resources and Evaluation (LREC)*, Valletta, Malta. European Language Resources Association (ELRA).

[Artola et al., 2004] Artola, X., de Ilarraza, A. D., Ezeiza, N., Gojenola, K., Sologaistoa, A., and Soroa, A. (2004). EULIA: a graphical web interface for creating, browsing and editing linguistically annotated corpora. In *International Conference on Language Resources and Evaluation (LREC) Workshop on XML-based richly annotated corpora*, Lisboa, Portugal.

[Artstein and Poesio, 2008] Artstein, R. and Poesio, M. (2008). Intercoder agreement for computational linguistics. *Computational Linguistics*, 34(4):555–596.

[Baker et al., 1998] Baker, C. F., Fillmore, C. J., and Lowe, J. B. (1998). The Berkeley FrameNet project. In *Annual Meeting of the Association for Computational Linguistics (ACL) and International Conference on Computational Linguistics (ICCL)*, ACL'98, pages 86–90, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Bakhouche et al., 2010] Bakhouche, B., Beys, B., Delattre, D., Guérin, C., and Tran, T. (2010). De l'annotation aux marginalia. http://meticebeta.univ-montp3.fr/lelivre/partie2/de_lannotation_aux_marginalia.html.

[Barque et al., 2010] Barque, L., Nasr, A., and Polguère, A. (2010). From the definitions of the trésor de la langue française to a semantic database of the French language. In *14th EURALEX International Congress*, Leeuwarden, Netherlands.

[Bartle, 1996] Bartle, R. (1996). Hearts, clubs, diamonds, spades: Players who suit MUDs. *The Journal of Virtual Environments*.

[Bayerl and Paul, 2011] Bayerl, P. S. and Paul, K. I. (2011). What determines inter-coder agreement in manual annotations? a meta-analytic investigation. *Computational Linguistics*, 37(4):699–725.

[Beck, 2011] Beck, K. (2011). Manifesto for agile software development. http://agilemanifesto.org/.

[Benzitoun et al., 2012] Benzitoun, C., Fort, K., and Sagot, B. (2012). TCOF-POS : un corpus libre de français parlé annoté en morphosyntaxe. In *Proceedings of the Traitement Automatique des Langues Naturelles (TALN)*, pages 99–112, Grenoble, France.

[Bestgen, 2009] Bestgen, Y. (2009). Quels indices pour mesurer l'efficacité en segmentation thématique? In *Proceedings of the Traitement Automatique des Langues Naturelles (TALN)*, page p. 10, Senlis, France.

[Bhardwaj et al., 2010] Bhardwaj, V., Passonneau, R., Salleb-Aouissi, A., and Ide, N. (2010). Anveshan: A tool for analysis of multiple annotators' labeling behavior. In *fourth linguistic annotation workshop (LAW IV)*, Uppsala, Sweden.

[Bird et al., 2000] Bird, S., Day, D., Garofolo, J. S., Henderson, J., Laprun, C., and Liberman, M. (2000). ATLAS: a flexible and extensible architecture for linguistic annotation. In *International Conference on Language Resources and Evaluation (LREC)*, pages 1699–1706, Athens, Greece. ELRA.

[Bird and Liberman, 2001] Bird, S. and Liberman, M. (2001). A formal framework for linguistic annotation. *Speech Communication*, 33(1-2):23–60.

[Böhmová et al., 2001] Böhmová, A., Hajič, J., Hajičová, E., and Hladká, B. (2001). The prague dependency treebank: Three-level annotation scenario. In Abeillé, A., editor, *Treebanks: Building and Using Syntactically Annotated Corpora*. Kluwer Academic Publishers.

[Bohnet et al., 2013] Bohnet, B., Burga, A., and Wanner, L. (2013). Towards the annotation of penn treebank with information structure. In *Proceedings of the Sixth International Joint Conference on Natural Language Processing*, pages 1250–1256. Asian Federation of Natural Language Processing.

[Bonneau-Maynard et al., 2005] Bonneau-Maynard, H., Rosset, S., Ayache, C., Kuhn, A., and Mostefa, D. (2005). Semantic annotation of the French Media dialog corpus. In *InterSpeech*, Lisboa, Portugal.

[Bontcheva et al., 2010] Bontcheva, K., Cunningham, H., Roberts, I., and Tablan, V. (2010). Web-based collaborative corpus annotation: Requirements and a framework implementation. In Witte, R., Cunningham, H., Patrick, J., Beisswanger, E., Buyko, E., Hahn, U., Verspoor, K., and Coden, A. R., editors, *workshop on New Challenges for NLP Frameworks (NLPFrameworks 2010)*, Valletta, Malta. ELRA.

[Bowman et al., 2015] Bowman, S. R., Angeli, G., Potts, C., and Manning, C. D. (2015). A large annotated corpus for learning natural language inference. *arXiv preprint arXiv:1508.05326*.

[Bretonnel Cohen et al., 2015] Bretonnel Cohen, K., Verspoor, K., Fort, K., Funk, C., Bada, M., Palmer, M., and Hunter, L. (2015). The Colorado Richly Annotated Full Text (CRAFT) Corpus: Multi-Model Annotation In The Biomedical Domain. In *Handbook of Linguistic Annotation*.

[Burghardt, 2012] Burghardt, M. (2012). Usability recommendations for annotation tools. In *Proceedings of the Sixth Linguistic Annotation Workshop*, pages 104–112, Jeju, Republic of Korea. Association for Computational Linguistics.

[Callison-Burch, 2014] Callison-Burch, C. (2014). Crowd-workers: Aggregating information across turkers to help them find higher paying work. In *The Second AAAI Conference on Human Computation and Crowdsourcing (HCOMP-2014)*.

[Callison-Burch and Dredze, 2010] Callison-Burch, C. and Dredze, M. (2010). Creating speech and language data with Amazon's Mechanical Turk. In *CSLDAMT '10: Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, Morristown, NJ, USA. Association for Computational Linguistics.

[Calzada and Dekhtyar, 2010] Calzada, G. D. L. and Dekhtyar, A. (2010). On measuring the quality of wikipedia articles. In *in 4th International Workshop on Information Credibility on the Web Proceedings*, pages 11–18.

[Calzolari et al., 2012] Calzolari, N., Gratta, R. D., Francopoulo, G., Mariani, J., Rubino, F., Russo, I., and Soria, C. (2012). The LRE Map. harmonising community descriptions of resources. In *Proc. of the Eight International Conference on Language Resources and Evaluation (LREC)*, Istanbul, Turkey.

[Campos et al., 2014] Campos, D., Lourenço, J., Matos, S., and Oliveira, J. L. (2014). Egas: a collaborative and interactive document curation platform. *Database.*

[Candito et al., 2014] Candito, M., Perrier, G., Guillaume, B., Ribeyre, C., Fort, K., Seddah, D., and de la Clergerie, E. (2014). Deep syntax annotation of the sequoia french treebank. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC 2014)*, Reykjavík, Iceland.

[Candito and Seddah, 2012] Candito, M. and Seddah, D. (2012). Le corpus Sequoia : annotation syntaxique et exploitation pour l'adaptation d'analyseur par pont lexical. In *Proceedings of the Traitement Automatique des Langues Naturelles (TALN)*, Grenoble, France.

[Carletta, 1996] Carletta, J. (1996). Assessing agreement on classification tasks: the kappa statistic. *Computational Linguistics*, 22:249–254.

[Carmen et al., 2010] Carmen, M., Felt, P., Haertel, R., Lonsdale, D., McClanahan, P., Merkling, O., Ringger, E., and Seppi, K. (2010). Tag dictionaries accelerate manual annotation. In *International Conference on Language Resources and Evaluation (LREC)*, Valletta, Malta. European Language Resources Association (ELRA).

[Chamberlain et al., 2013] Chamberlain, J., Fort, K., Kruschwitz, U., Lafourcade, M., and Poesio, M. (2013). Using games to create language resources: Successes and limitations of the approach. In Gurevych, I. and Kim, J., editors, *The People's Web Meets NLP*, Theory and Applications of Natural Language Processing, pages 3–44. Springer Berlin Heidelberg.

[Chamberlain et al., 2009a] Chamberlain, J., Kruschwitz, U., and Poesio, M. (2009a). Constructing an anaphorically annotated corpus with non-experts: assessing the quality of collaborative annotations. In *Proceedings of the 2009 Workshop on The People's Web Meets NLP: Collaboratively Constructed Semantic Resources*, People's Web '09, pages 57–62, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Chamberlain et al., 2008] Chamberlain, J., Poesio, M., and Kruschwitz, U. (2008). Phrase Detectives: a web-based collaborative annotation game. In *Proceedings of the International Conference on Semantic Systems (I-Semantics'08)*, Graz, Austria.

[Chamberlain et al., 2009b] Chamberlain, J., Poesio, M., and Kruschwitz, U. (2009b). A new life for a dead parrot: Incentive structures in the phrase detectives game. In *Proceedings of WWW 2009*, Madrid, Spain.

[Church, 2011] Church, K. (2011). A pendulum swung too far. *Linguistic Issues in Language Technology - LiLT*, 6.

[Church, 1988] Church, K. W. (1988). A stochastic parts program and noun phrase parser for unrestricted text. In *Proceedings of the Second Conference on Applied Natural Language Processing*, ANLC '88, pages 136–143, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Cimiano and Handschuh, 2003] Cimiano, P. and Handschuh, S. (2003). Ontology-based linguistic annotation. In *Proceedings of the ACL 2003 Workshop on Linguistic Annotation*, pages 14–21, Morristown, NJ, USA. Association for Computational Linguistics.

[Coates-Stephens, 1992] Coates-Stephens, S. (1992). The analysis and acquisition of proper names for the understanding of free text. *Computers and the Humanities*, 26(5-6):441–456.

[Cohen, 1960] Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20(1):37–46.

[Cohen, 1968] Cohen, J. (1968). Weighted kappa: Nominal scale agreement with provision for scaled disagreement or partial credit. *Psychological Bulletin*, 70(4):213–220.

[Cohen et al., 2005] Cohen, K. B., Fox, L., Ogren, P. V., and Hunter, L. (2005). Corpus design for biomedical natural language processing. In *Proceedings of the ACL-ISMB workshop on linking biological literature, ontologies and databases: mining biological semantics*, pages 38–45.

[Colombat, 1988] Colombat, B. (1988). Présentation: Eléments de réflexion pour une histoire des parties du discours. *Langages*, 23(92):5–10.

[Cook and Stevenson, 2010] Cook, P. and Stevenson, S. (2010). Automatically identifying changes in the semantic orientation of words. In Chair), N. C. C., Choukri, K., Maegaard, B., Mariani, J., Odijk, J., Piperidis, S., Rosner, M., and Tapias, D., editors, *International Conference on Language Resources and Evaluation (LREC)*, Valletta, Malta.

[Cooper et al., 2010] Cooper, S., Treuille, A., Barbero, J., Leaver-Fay, A., Tuite, K., Khatib, F., Snyder, A. C., Beenen, M., Salesin, D., Baker, D., and Popović, Z. (2010). The challenge of designing scientific discovery games. In *Proceedings of the Fifth International Conference on the Foundations of Digital Games*, FDG '10, pages 40–47, New York, NY, USA. ACM.

[Couillault et al., 2014] Couillault, A., Fort, K., Adda, G., and De Mazancourt, H. (2014). Evaluating Corpora Documentation with regards to the Ethics and Big Data Charter. In *International Conference on Language Resources and Evaluation (LREC)*, Reykjavik, Islande.

[Cunningham et al., 2002] Cunningham, H., Maynard, D., Bontcheva, K., and Tablan, V. (2002). GATE: a framework and graphical development environment for robust NLP tools and applications. In *Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics*, Philadelphia, USA.

[Dandapat et al., 2009] Dandapat, S., Biswas, P., Choudhury, M., and Bali, K. (2009). Complex linguistic annotation - no easy way out! a case from bangla and hindi POS labeling tasks. In *Proceedings of the third ACL Linguistic Annotation Workshop*, Singapore.

[Davies and Fleiss, 1982] Davies, M. and Fleiss, J. L. (1982). Measuring agreement for multinomial data. *Biometrics*, 38(4):1047–1051.

[Day et al., 2004] Day, D., McHenry, C., Kozierok, R., and Riek, L. (2004). Callisto: A configurable annotation workbench. In *International Conference on Language Resources and Evaluation*, Lisboa, Portugal.

[de Castilho et al., 2014] de Castilho, R. E., Biemann, C., Gurevych, I., and Yimam, S. M. (2014). Webanno: a flexible, web-based annotation tool for clarin. In *Proceedings of the CLARIN Annual Conference (CAC) 2014*, page online, Utrecht, Netherlands. CLARIN ERIC. Extended abstract.

[de la Clergerie, 2008] de la Clergerie, E. V. (2008). A collaborative infrastructure for handling syntactic annotations. In *First International Workshop on Automated Syntactic Annotations for interoperable Language Resources*, Hong-Kong, China.

[Denis and Sagot, 2009] Denis, P. and Sagot, B. (2009). Coupling an annotated corpus and a morphosyntactic lexicon for state-of-the-art POS tagging with less human effort. In *Pacific Asia Conference on Language Information and Computing (PACLIC)*, Hong-Kong, China.

[Desrosières, 1998] Desrosières, A. (1998). *The Politics of Large Numbers: A History of Statistical Reasoning*. Harvard UP, Cambridge, MA.

[Desrosières and Didier, 2014] Desrosières, A. and Didier, E. (2014). *Prouver et gouverner: Une analyse politique des statistiques publiques*. Sciences humaines. LA DECOUVERTE.

[Desrosières and Thévenot, 2002] Desrosières, A. and Thévenot, L. (2002). *Les catégories socio-professionnelles*. Number 62 in Repères – sociologie. La Découverte, Paris, 5e edition.

[Di Eugenio and Glass, 2004] Di Eugenio, B. and Glass, M. (2004). The kappa statistic: a second look. *Computational Linguistics*, 30(1):95–101.

[Dipper et al., 2004] Dipper, S., Götze, M., and Stede, M. (2004). Simple annotation tools for complex annotation tasks: an evaluation. In A. Witt, U. Heid, H. S. T. J. C. and Wittenburg, P., editors, *International Conference on Language Resources and Evaluation (LREC) Workshop on XML-based richly annotated corpora (XBRAC)*, Lisboa, Portugal.

[Estellés-Arolas and González-Ladrón-de Guevara, 2012] Estellés-Arolas, E. and González-Ladrón-de Guevara, F. (2012). Towards an integrated crowdsourcing definition. *Journal of Information Science*, 38(2):189–200.

[Fellbaum, 1998] Fellbaum, C. (1998). *WordNet: An Electronic Lexical Database*. MIT Press.

[Felt et al., 2010] Felt, P., Merkling, O., Carmen, M., Ringger, E., Lemmon, W., Seppi, K., and Haertel, R. (2010). CCASH: A web application framework for efficient, distributed language resource development. In Chair), N. C. C., Choukri, K., Maegaard, B., Mariani, J., Odijk, J., Piperidis, S., Rosner, M., and Tapias, D., editors, *International Conference on Language Resources and Evaluation (LREC)*, Valletta, Malta. European Language Resources Association (ELRA).

[Fenouillet et al., 2009] Fenouillet, F., Kaplan, J., and Yennek, N. (2009). Serious games et motivation. In *4ème Conférence francophone sur les Environnements Informatiques pour l'Apprentissage Humain (EIAH'09), vol. Actes de l'Atelier "Jeux Sérieux: conception et usages*, pages 41–52.

[Ferrucci and Lally, 2004] Ferrucci, D. and Lally, A. (2004). UIMA: an architectural approach to unstructured information processing in the corporate research environment. *Nat. Lang. Eng.*, 10:327–348.

[Fleiss, 1971] Fleiss, J. L. (1971). Measuring nominal scale agreement among many raters. *Psychological Bulletin*, 76, 5:378–382.

[Fort, 2012] Fort, K. (2012). *Les ressources annotées, un enjeu pour l'analyse de contenu : vers une méthodologie de l'annotation manuelle de corpus*. PhD thesis, Université Paris XIII, LIPN, INIST-CNRS.

[Fort et al., 2011a] Fort, K., Adda, G., and Cohen, K. B. (2011a). Amazon Mechanical Turk: Gold mine or coal mine? *Computational Linguistics (editorial)*, 37(2):413–420.

[Fort et al., 2014a] Fort, K., Adda, G., Sagot, B., Mariani, J., and Couillault, A. (2014a). Crowdsourcing for Language Resource Development: Criticisms About Amazon Mechanical Turk Overpowering Use. In Vetulani, Z. and Mariani, J., editors, *Human Language Technology Challenges for Computer Science and Linguistics*, pages 303–314. Springer International Publishing.

[Fort and Claveau, 2012] Fort, K. and Claveau, V. (2012). Annotating football matches: Influence of the source medium on manual annotation. In *International Conference on Language Resources and Evaluation (LREC)*, Istanbul, Turkey. 6 pages.

[Fort et al., 2009] Fort, K., Ehrmann, M., and Nazarenko, A. (2009). Towards a methodology for named entities annotation. In *3rd ACL Linguistic Annotation Workshop (LAW III)*, pages 142–145, Singapore. Poster.

[Fort et al., 2012a] Fort, K., François, C., Galibert, O., and Ghribi, M. (2012a). Analyzing the impact of prevalence on the evaluation of a manual annotation campaign. In *International Conference on Language Resources and Evaluation (LREC)*, Istanbul, Turkey. 7 pages.

[Fort et al., 2014b] Fort, K., Guillaume, B., and Chastant, H. (2014b). Creating Zombilingo, a Game With A Purpose for dependency syntax annotation. In *Gamification for Information Retrieval (GamifIR'14) Workshop*, Amsterdam, Pays-Bas.

[Fort et al., 2014c] Fort, K., Guillaume, B., and Stern, V. (2014c). Zombilingo: eating heads to perform dependency syntax annotation (zombilingo : manger des têtes pour annoter en syntaxe de dépendances) [in french]. In *Proceedings of TALN 2014 (Volume 3: System Demonstrations)*, pages 15–16, Marseille, France. Association pour le Traitement Automatique des Langues.

[Fort et al., 2011b] Fort, K., Nazarenko, A., and Ris, C. (2011b). Corpus linguistics for the annotation manager. In *Corpus Linguistics Conference*, Birmingham, England. 13 pages.

[Fort et al., 2012b] Fort, K., Nazarenko, A., and Rosset, S. (2012b). Modeling the complexity of manual annotation tasks: a grid of analysis. In *International Conference on Computational Linguistics (COLING)*, pages 895–910, Mumbai, India.

[Fort and Sagot, 2010] Fort, K. and Sagot, B. (2010). Influence of pre-annotation on POS-tagged corpus development. In *Fourth ACL Linguistic Annotation Workshop*, pages 56–63, Uppsala, Sweden.

[Frunzeanu and Pons, 2012] Frunzeanu, E. and Pons, P. (2012). Les "encyclopédies" médiévales et les digital humanities: l'évolution du programme sourcencyme. Journée d'étude sur l'annotation collaborative de corpus.

[Gala and Brun, 2012] Gala, N. and Brun, C. (2012). Propagation de polarités dans des familles de mots : impact de la morphologie dans la construction d'un lexique pour l'analyse d'opinions. In *Actes de Traitement Automatique des Langues Naturelles (TALN 2012)*, Grenoble.

[Geertzen et al., 2008] Geertzen, J., Petukhova, V., and Bunt, H. (2008). Evaluating dialogue act tagging with naive and expert annotators. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, B. M. J. M. J. O. S. P. D. T., editor, *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco. European Language Resources Association (ELRA). http://www.lrec-conf.org/proceedings/lrec2008/.

[Geiger et al., 2011] Geiger, D., Seedorf, S., Schulze, T., Nickerson, R. C., and Schader, M. (2011). Managing the crowd: Towards a taxonomy of crowd-sourcing processes. In *AMCIS 2011 Proceedings*.

[Goecke et al., 2010] Goecke, D., Lüngen, H., Metzing, D., Stührenberg, M., and Witt, A. (2010). Different views on markup. In Witt, A., Metzing, D., and Ide, N., editors, *Linguistic Modeling of Information and Markup Languages*, volume 40 of *Text, Speech and Language Technology*, pages 1–21. Springer Netherlands.

[Green, 1997] Green, A. M. (1997). Kappa statistics for multiple raters using categorical classifications. In *Proceedings of the Twenty-Second Annual Conference of SAS Users Group*, San Diego, USA.

[Grishman, 1998] Grishman, R. (1998). TIPSTER architecture design document version 3.1. Technical report, DARPA.

[Grishman and Sundheim, 1996] Grishman, R. and Sundheim, B. (1996). Message Understanding Conference-6: a brief history. In *the 16th conference on Computational linguistics*, pages 466–471, Morristown, NJ, USA. Association for Computational Linguistics.

[Grouin et al., 2011] Grouin, C., Rosset, S., Zweigenbaum, P., Fort, K., Galibert, O., and Quintard, L. (2011). Proposal for an extension of traditional named entities: From guidelines to evaluation, an overview. In *5th Linguistic Annotation Workshop*, pages 92–100, Portland, Oregon, USA. Poster.

[Gupta et al., 2014] Gupta, N., Martin, D., Hanrahan, B. V., and O'Neill, J. (2014). Turk-life in india. In *Proceedings of the 18th International Conference on Supporting Group Work*, GROUP '14, pages 1–11, New York, NY, USA. ACM.

[Gut and Bayerl, 2004] Gut, U. and Bayerl, P. S. (2004). Measuring the reliability of manual annotations of speech corpora. In *Speech Prosody*, pages 565–568, Nara, Japan.

[Gwet, 2012] Gwet, K. L. (2012). *Handbook of Inter-rater Reliability*. Advanced Analytics, LLC, third edition.

[Habert, 2005] Habert, B. (2005). Portrait de linguiste(s) à l'instrument. *Texto!*, vol. X(4).

[Halfaker et al., 2013] Halfaker, A., Geiger, R. S., Morgan, J., and Riedl, J. (2013). The rise and decline of an open collaboration system: How wikipedia's reaction to sudden popularity is causing its decline. *American Behavioral Scientist*, 57(5).

[Hellmann et al., 2013] Hellmann, S., Lehmann, J., Auer, S., and Brümmer, M. (2013). Integrating nlp using linked data. In *Proceedings of the 12nd International Semantic Web Conference (ISWC)*. Springer.

[Hong and Baker, 2011] Hong, J. and Baker, C. F. (2011). How good is the crowd at "real" WSD? In *Proceedings of the 5th Linguistic Annotation Workshop*, pages 30–37, Portland, Oregon, USA. Association for Computational Linguistics.

[Hovy and Lavid, 2010] Hovy, E. H. and Lavid, J. M. (2010). Towards a "science" of corpus annotation: A new methodological challenge for corpus linguistics. *International Journal of Translation Studies*, 22(1).

[Howe, 2006a] Howe, J. (2006a). Crowdsourcing: A definition. http://crowdsourcing.typepad.com/cs/2006/06/crowdsourcing_a.html.

[Howe, 2006b] Howe, J. (2006b). The rise of crowdsourcing. *Wired Magazine*, 14(6).

[Hripcsak and Heitjan, 2002] Hripcsak, G. and Heitjan, D. F. (2002). Measuring agreement in medical informatics reliability studies. *Journal of Biomedical Informatics*, 35(2):99–110.

[Hripcsak and Rothschild, 2005] Hripcsak, G. and Rothschild, A. S. (2005). Agreement, the f measure, and reliability in information retrieval. *Journal of the American Medical Informatics Association (JAMIA)*, 12(3):296–298.

[Hu et al., 2007] Hu, M., Lim, E.-P., Sun, A., Lauw, H. W., and Vuong, B.-Q. (2007). Measuring article quality in wikipedia: Models and evaluation. In *Proceedings of the Sixteenth ACM Conference on Conference on Information and Knowledge Management*, CIKM '07, pages 243–252, New York, NY, USA. ACM.

[Ide et al., 2008] Ide, N., Baker, C., Fellbaum, C., Fillmore, C., and Passonneau, R. (2008). MASC: the manually annotated sub-corpus of american english. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, B. M. J. M. J. O. S. P. D. T., editor, *International Language Resources and Evaluation (LREC)*, Marrakech, Morocco. European Language Resources Association (ELRA).

[Ide and Romary, 2006] Ide, N. and Romary, L. (2006). Representing linguistic corpora and their annotations. In *International Conference on Language Resources and Evaluation (LREC)*, Genoa, Italy.

[Ide and Suderman, 2007] Ide, N. and Suderman, K. (2007). GrAF: A graph-based format for linguistic annotations. In *Linguistic Annotation Workshop, held in conjunction with ACL 2007*, pages 1–8, Prague, Czech Republic.

[Ipeirotis, 2010a] Ipeirotis, P. (2010a). Mechanical turk requester activity: The insignificance of the long tail. http://behind-the-enemy-lines.blogspot.com/2010/10/mechanical-turk-requester-activity.html.

[Ipeirotis, 2010b] Ipeirotis, P. (2010b). The new demographics of mechanical turk. http://behind-the-enemy-lines.blogspot.com/2010/03/new-demographics-of-mechanical-turk.html.

[Jourde et al., 2011] Jourde, J., Manine, A.-P., Veber, P., Fort, K., Bossy, R., Alphonse, E., and Bessières, P. (2011). BioNLP shared task 2011 – bacteria gene interactions and renaming. In *BioNLP Shared Task 2011 Workshop*, pages 65–73, Portland, Oregon, USA. Association for Computational Linguistics.

[Jurafsky and Martin, 2009] Jurafsky, D. and Martin, J. H. (2009). *Speech and Language Processing (2Nd Edition)*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA.

[Kaisser and Lowe, 2008] Kaisser, M. and Lowe, J. B. (2008). Creating a research collection of question answer sentence pairs with Amazon's Mechanical Turk. In *International Conference on Language Resources and Evaluation (LREC)*.

[Kaplan et al., 2012] Kaplan, D., Iida, R., Nishina, K., and Tokunaga, T. (2012). Slate - a tool for creating and maintaining annotated corpora. *Journal for Language Technology and Computational Linguistics*, 26(2):89 – 101.

[Kaplan et al., 2010] Kaplan, D., Iida, R., and Tokunaga, T. (2010). Annotation process management revisited. In *International Conference on Language Resources and Evaluation (LREC)*, pages 365 – 366, Valletta, Malta.

[Khatib et al., 2011] Khatib, F., DiMaio, F., Cooper, S., Kazmierczyk, M., Gilski, M., Krzywda, S., Zabranska, H., Pichova, I., Thompson, J., Popović, Z., et al. (2011). Crystal structure of a monomeric retroviral protease solved by protein folding game players. *Nature structural & molecular biology*, 18(10):1175–1177.

[Kim et al., 2008] Kim, J.-D., Ohta, T., and Tsujii, J. (2008). Corpus annotation for mining biomedical events from literature. *BMC Bioinformatics*, 9(1):10.

[Kim and Wang, 2012] Kim, J.-D. and Wang, Y. (2012). CSAF - a community-sourcing annotation framework. In *Proceedings of the Sixth Linguistic Annotation Workshop*, pages 153–156, Jeju, Republic of Korea. Association for Computational Linguistics.

[King et al., 2013] King, D. L., Haagsma, M. C., Delfabbro, P. H., Gradisar, M., and Griffiths, M. D. (2013). Toward a consensus definition of pathological video-gaming: A systematic review of psychometric assessment tools. *Clinical Psychology Review*, 33(3):331 – 342.

[Kittur et al., 2007] Kittur, A., Chi, E., Pendleton, B. A., Suh, B., and Mytkowicz, T. (2007). Power of the few vs. wisdom of the crowd: Wikipedia and the rise of the bourgeoisie. *World Wide Web*, 1(2):19.

[Klebanov and Beigman, 2009] Klebanov, B. B. and Beigman, E. (2009). From annotator agreement to noise models. *Computational Linguistics*, 35(4):495–503. From Maud Ehrmann.

[Krippendorff, 1980] Krippendorff, K. (1980). *Content Analysis: An Introduction to Its Methodology*. Sage, Beverly Hills, CA., USA.

[Krippendorff, 2004] Krippendorff, K. (2004). *Content Analysis: An Introduction to Its Methodology,*. Sage, Thousand Oaks, CA., USA, second edition edition.

[Krippendorff, 2013] Krippendorff, K. (2013). *Content Analysis: An Introduction to Its Methodology*. Sage, Thousand Oaks, CA, 3rd edition edition.

[Kruschwitz et al., 2009] Kruschwitz, U., Chamberlain, J., and Poesio, M. (2009). (linguistic) science through web collaboration in the ANAWIKI project. In *Proceedings of the WebSci'09: Society On-Line*, Athens, Greece.

[Lafourcade and Joubert, 2008] Lafourcade, M. and Joubert, A. (2008). JeuxDeMots : un prototype ludique pour l'émergence de relations entre termes. In *Proceedings of the Journées internationales d'Analyse statistique des Données Textuelles (JADT)*, Lyon, France.

[Lafourcade and Joubert, 2015] Lafourcade, M. and Joubert, A. (2015). *Language Production, Cognition, and the Lexicon. Festschrift in honor of Michael Zock.*, volume XI of *Text, Speech and Language Technology*, chapter TOTAKI:

A Help for Lexical Access on the TOT Problem., pages 95–112. Springer, Dordrecht.

[Lafourcade et al., 2015a] Lafourcade, M., Le Brun, N., and Joubert, A. (2015a). Vous aimez ?...ou pas ? likeit, un jeu pour construire une ressource lexicale de polarité. In *Actes de la 22e conférence sur le Traitement Automatique des Langues Naturelles*, pages 330–336, Caen, France. Association pour le Traitement Automatique des Langues.

[Lafourcade and Lebrun, 2014] Lafourcade, M. and Lebrun, N. (2014). éthique et construction collaborative de données lexicales par des gwaps. Journée d'étude ATALA Éthique et TAL.

[Lafourcade et al., 2015b] Lafourcade, M., Lebrun, N., and Joubert, A. (2015b). *Jeux et intelligence collective: résolution de problèmes et acquisition de données sur le web*. Collection science cognitive et management des connaissances. ISTE.

[Laignelet and Rioult, 2009] Laignelet, M. and Rioult, F. (2009). Repérer automatiquement les segments obsolescents à l'aide d'indices sémantiques et discursifs. In *Proceedings of the Traitement Automatique des Langues Naturelles (TALN)*, Senlis, France.

[Landis and Koch, 1977] Landis, J. R. and Koch, G. G. (1977). The measurement of observer agreement for categorical data. *Biometrics*, 33(1):159–174.

[Landragin et al., 2012] Landragin, F., Poibeau, T., and Victorri, B. (2012). ANALEC: a new tool for the dynamic annotation of textual data. In Chair), N. C. C., Choukri, K., Declerck, T., Doğan, M. U., Maegaard, B., Mariani, J., Odijk, J., and Piperidis, S., editors, *International Conference on Language Resources and Evaluation (LREC)*, Istanbul, Turkey. European Language Resources Association (ELRA).

[Latour, 1987] Latour, B. (1987). *Science in Action: How to Follow Scientists and Engineers through Society*. Harvard University Press, Cambridge, MA.

[Lebarbé, 2008] Lebarbé, T. (2008). CLELIA: Building a manucript archive throught interdisciplinary dialogue. In *The Marriage of Mercury and Philology: Problems and Outcomes in Digital Philology*.

[Leech, 1993] Leech, G. (1993). Corpus annotation schemes. *Literary and Linguistic Computing*, 8(4):275–281.

[Leech, 1997] Leech, G. (1997). *Corpus annotation: Linguistic information from computer text corpora*, chapter Introducing corpus annotation, pages 1–18. Longman, Londres, England.

[Liberman, 2009] Liberman, M. (2009). The annotation conundrum. In *Proceedings of the EACL 2009 Workshop on the Interaction Between Linguistics and Computational Linguistics: Virtuous, Vicious or Vacuous?*, ILCL '09, pages 2–2, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Lortal et al., 2006] Lortal, G., Lewkowicz, M., and Todiracu-Courtier, A. (2006). *Annotations dans les documents pour l'action*, chapter Des activités d'annotation : De la glose au document, pages 153–171. Hermes Publishing, Londres-Paris.

[Maeda et al., 2006] Maeda, K., Lee, H., Medero, J., and Strassel, S. (2006). A new phase in annotation tool development at the linguistic data consortium: The evolution of the Annotation Graph Toolkit. In *International Conference on Language Resources and Evaluation (LREC)*, Genoa, Italy.

[Maeda and Strassel, 2004] Maeda, K. and Strassel, S. (2004). Annotation tools for large-scale corpus development: Using AGTK at the Linguistic Data Consortium. In *International Conference on Language Resources and Evaluation (LREC)*, Lisboa, Portugal.

[Makhoul et al., 1999] Makhoul, J., Kubala, F., Schwartz, R., and Weischedel, R. (1999). Performance measures for information extraction. In *Proceedings of DARPA Broadcast News Workshop*, pages 249–252.

[Marcus et al., 1993] Marcus, M., Santorini, B., and Marcinkiewicz, M. A. (1993). Building a large annotated corpus of English : The Penn Treebank. *Computational Linguistics*, 19(2):313–330.

[Marczewski, 2015] Marczewski, A. (2015). *Even Ninja Monkeys Like to Play: Gamification, Game Thinking and Motivational Design*. CreateSpace Independent Publishing Platform, 1 edition.

[Mathet and Widlöcher, 2011] Mathet, Y. and Widlöcher, A. (2011). Une approche holiste et unifiée de l'alignement et de la mesure d'accord interannotateurs. In *Proceedings of the Traitement Automatique des Langues Naturelles (TALN)*, Montpellier, France.

[Mathet et al., 2012] Mathet, Y., Widlöcher, A., Fort, K., François, C., Galibert, O., Grouin, C., Kahn, J., Rosset, S., and Zweigenbaum, P. (2012). Manual corpus annotation: Evaluating the evaluation metrics. In *International Conference on Computational Linguistics (COLING)*, pages 809–818, Mumbai, India. Poster.

[Mathet et al., 2015] Mathet, Y., Widlöcher, A., and Métivier, J.-P. (2015). The unified and holistic method gamma (γ) for inter-annotator agreement measure and alignment. *Computational Linguistics*, 41(3):437–479.

[Morton and LaCivita, 2003] Morton, T. and LaCivita, J. (2003). WordFreak: an open tool for linguistic annotation. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology: Demonstrations - Volume 4*, NAACL-Demonstrations '03, pages 17–18, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Müller and Strube, 2006] Müller, C. and Strube, M. (2006). Multi-level annotation of linguistic data with MMAX2. In Braun, S., Kohn, K., and Mukherjee, J., editors, *Corpus Technology and Language Pedagogy: New Resources, New Tools, New Methods*, pages 197–214. Peter Lang, Frankfurt, Germany.

[Muzerelle, 1985] Muzerelle, D. (1985). *Vocabulaire codicologique : répertoire méthodique des termes français relatifs aux manuscrits*. Editions CEMI, Paris.

[Nakamura and Csikszentmihalyi, 2002] Nakamura, J. and Csikszentmihalyi, M. (2002). The concept of flow. *Handbook of positive psychology*, pages 89–105.

[Nédellec et al., 2006] Nédellec, C., Bessières, P., Bossy, R., Kotoujansky, A., and Manine, A.-P. (2006). Annotation guidelines for machine learning-based named entity recognition in microbiology. In et C. Nédellec, M. H., editor, *Proceedings of the Data and text mining in integrative biology workshop*, pages 40–54, Berlin, Germany.

[Novotney and Callison-Burch, 2010] Novotney, S. and Callison-Burch, C. (2010). Cheap, fast and good enough: automatic speech recognition with non-expert transcription. In *Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, HLT'10, pages 207–215, Stroudsburg, PA, USA. Association for Computational Linguistics.

[O'Donnell, 2008] O'Donnell, M. (2008). Demonstration of the UAM Corpus-Tool for text and image annotation. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies: Demo Session*, HLT-Demonstrations '08, pages 13–16, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Ogren, 2006] Ogren, P. (2006). Knowtator: A plug-in for creating training and evaluation data sets for biomedical natural language systems. In *Protégé Conference*, Stanford, USA.

[Orasan, 2003] Orasan, C. (2003). Palinka: A highly customisable tool for discourse annotation. In *Proceedings of SIGdial Workshop on Discourse and Dialogue*, Sapporo, Japan.

[Péry-Woodley et al., 2011] Péry-Woodley, M.-P., Afantenos, S., Ho-Dac, L.-M., and Asher, N. (2011). La ressource ANNODIS, un corpus enrichi d'annotations discursives. *Traitement Automatique des Langues*, 52(3):71–101.

[Petasis, 2012] Petasis, G. (2012). The SYNC3 collaborative annotation tool. In *International Conference on Language Resources and Evaluation (LREC)*, Istanbul, Turkey. European Language Resources Association (ELRA).

[Pevzner and Hearst, 2002] Pevzner, L. and Hearst, M. A. (2002). A critique and improvement of an evaluation metric for text segmentation. *Computational Linguistics*, 28(1):19–36.

[Poesio and Artstein, 2005] Poesio, M. and Artstein, R. (2005). The reliability of anaphoric annotation, reconsidered: Taking ambiguity into account. In *Proceedings of the Workshop on Frontiers in Corpus Annotations II: Pie in the Sky*, pages 76–83, Ann Arbor, Michigan, USA. Association for Computational Linguistics. Implicit ambiguity much more frequent than explicit ambiguity.

[Przepiórkowski and Murzynowski, 2009] Przepiórkowski, A. and Murzynowski, G. (2009). Manual annotation of the national corpus of Polish with Anotatornia. In Goźdź-Roszkowski, S., editor, *The proceedings of Practical Applications in Language and Computers PALC 2009*, Frankfurt, Germany. Peter Lang. Forthcoming.

[Pustejovsky and Stubbs, 2012] Pustejovsky, J. and Stubbs, A. (2012). *Natural Language Annotation for Machine Learning*. O'Reilly.

[Reidsma and Carletta, 2008] Reidsma, D. and Carletta, J. (2008). Reliability measurement without limits. *Computational Linguistics*, 34(3):319–326.

[Reidsma et al., 2004] Reidsma, D., Jovanovic, N., and Hofs, D. (2004). Designing annotation tools based on properties of annotation problems.

[Ross et al., 2010] Ross, J., Irani, L., Silberman, M. S., Zaldivar, A., and Tomlinson, B. (2010). Who are the crowdworkers?: shifting demographics in mechanical turk. In *Proceedings of the 28th of the international conference extended abstracts on Human factors in computing systems*, CHI EA '10, pages 2863–2872, New York, NY, USA. ACM.

[Rosset et al., 2012] Rosset, S., Grouin, C., Fort, K., Galibert, O., Kahn, J., and Zweigenbaum, P. (2012). Structured named entities in two distinct press corpora: Contemporary broadcast news and old newspapers. In *6th Linguistic Annotation Workshop (LAW VI)*, pages 40–48, Jeju, Republic of Korea.

[Sampson, 2000] Sampson, G. (2000). The role of taxonomy in language engineering. *Philosophical Transactions of the Royal Society of London. Series A:Mathematical, Physical and Engineering Sciences*, 358(1769):1339–1355.

[Santorini, 1990] Santorini, B. (1990). Part-of-speech tagging guidelines for the Penn Treebank Project. Technical Report MS-CIS-90-47, Department of Computer and Information Science, University of Pennsylvania.

[Schmid, 1997] Schmid, H. (1997). *New Methods in Language Processing, Studies in Computational Linguistics*, chapter Probabilistic part-of-speech tagging using decision trees, pages 154–164. UCL Press, London.

[Scott, 1955] Scott, W. A. (1955). Reliability of content analysis: The case of nominal scale coding. *Public Opinion Quaterly*, 19(3):321–325.

[Siegel and Castellan, 1988] Siegel, S. and Castellan, N. J. (1988). *Nonparametric Statistics for the Behavioral Sciences*. McGraw-Hill, New York, USA, 2nd edition.

[Sinclair, 2005] Sinclair, J. (2005). *Developing Linguistic Corpora: a Guide to Good Practice*, chapter Corpus and Text - Basic Principles, pages 1–16. Oxford: Oxbow Books.

[Snow et al., 2008] Snow, R., O'Connor, B., Jurafsky, D., and Ng., A. Y. (2008). Cheap and fast - but is it good? evaluating non-expert annotations for natural language tasks. In *Proceedings of EMNLP 2008*, pages 254–263.

[Stenetorp et al., 2012] Stenetorp, P., Pyysalo, S., Topić, G., Ohta, T., Anani-adou, S., and Tsujii, J. (2012). brat: a web-based tool for nlp-assisted text annotation. In *Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 102–107, Avignon, France. Association for Computational Linguistics.

[Stenetorp et al., 2011] Stenetorp, P., Topić, G., Pyysalo, S., Ohta, T., Kim, J.-D., and Tsujii, J. (2011). BioNLP shared task 2011: Supporting resources. In *BioNLP Shared Task 2011 Workshop*, pages 112–120, Portland, Oregon, USA. Association for Computational Linguistics.

[Stewart et al., 2015] Stewart, N., Ungemach, C., Harris, A. J. L., Bartels, D. M., Newell, B. R., Paolacci, G., and Chandler, J. (2015). The average laboratory samples a population of 7,300 amazon mechanical turk workers. *Judgment and Decision Making*, X(X):XX–XX.

[Stührenberg et al., 2007] Stührenberg, M., Goecke, D., Diewald, N., Mehler, A., and Cramer, I. (2007). Web-based annotation of anaphoric relations and lexical chains. In *Linguistic Annotation Workshop*, LAW '07, pages 140–147, Prague, Czech Republic. Association for Computational Linguistics.

[Sweetser and Wyeth, 2005] Sweetser, P. and Wyeth, P. (2005). Gameflow: a model for evaluating player enjoyment in games. *Computers in Entertainment (CIE)*, 3(3):3–3.

[Tellier, 2014] Tellier, M. (2014). Quelques orientations méthodologiques pour étudier la gestuelle dans des corpus spontanés et semi-contrôlés. *Discours*, 15.

[Urieli, 2013] Urieli, A. (2013). *Robust French syntax analysis: reconciling statistical methods and linguistic knowledge in the Talismane toolkit*. PhD thesis, Université de Toulouse II le Mirail, France.

[Véronis, 2000] Véronis, J. (2000). Annotation automatique de corpus : panorama et état de la technique. *Ingénierie des langues*, pages 111–129.

[von Ahn, 2006] von Ahn, L. (2006). Games with a purpose. *IEEE Computer Magazine*, pages 96–98.

[Voormann and Gut, 2008] Voormann, H. and Gut, U. (2008). Agile corpus creation. *Corpus Linguistics and Linguistic Theory*, 4(2):235–251.

[Voss, 2005] Voss, J. (2005). Measuring wikipedia. In *Proceedings International Conference of the International Society for Scientometrics and Informetrics ISSI*, Stockholm, Sweden.

[Widlöcher and Mathet, 2009] Widlöcher, A. and Mathet, Y. (2009). La plate-forme Glozz : environnement d'annotation et d'exploration de corpus. In *Proceedings of the Traitement Automatique des Langues Naturelles (TALN)*, Senlis, France.

[Woliński et al., 2011] Woliński, M., Głowińska, K., and Świdziński, M. (2011). A preliminary version of składnica - a treebank of polish. In *Proceedings of the 5th Language and Technology Conference*, Poznań, Poland.

[Wynne, 2005] Wynne, M., editor (2005). *Developing Linguistic Corpora: a Guide to Good Practice.* Oxford: Oxbow Books.