Modeling the Complexity of Manual Annotation Tasks: a Grid of Analysis

Karën Fort[1,2] *Adeline Nazarenko*[1] *Sophie Rosset*[3]
(1) LIPN, Université Paris 13 & CNRS, 99 av. J.B. Clément, 93430 Villetaneuse
(2) LORIA, Campus Scientifique, 54506 Vandœuvre-lès-Nancy
(3) LIMSI-CNRS Rue John von Neumann, Université Paris-Sud 91403 Orsay
karen.fort@loria.fr, adeline.nazarenko@lipn.univ-paris13.fr,
sophie.rosset@limsi.fr

ABSTRACT
Manual corpus annotation is getting widely used in Natural Language Processing (NLP). While being recognized as a difficult task, no in-depth analysis of its complexity has been performed yet. We provide in this article a grid of analysis of the different complexity dimensions of an annotation task, which helps estimating beforehand the difficulties and cost of annotation campaigns. We observe the applicability of this grid on existing annotation campaigns and detail its application on a real-world example.

KEYWORDS: manual corpus annotation, annotation campaign management, annotation campaign cost estimate.

1 Introduction

With the development of NLP applications, the annotation campaigns are becoming more numerous and more varied. Annotated corpora are used for acquiring knowledge as well as for testing theories, models and tools. They can be directly used in end-applications or for specific internal tasks.

Manual annotation is actually a widespread practice in NLP. It consists in adding labels of linguistic nature or reflecting the usage of NLP technologies on some oral or written discourse. This corresponds to a great diversity of phenomena, as these annotations vary in nature (phonetic, morpho-syntactic, semantic or task-oriented labels), in the range they cover (they can concern a couple of characters, a word, a paragraph or a whole text), in their degree of coverage (all the text is annotated or only a part of it) and in their form (atomic value, complex feature structures or relations and even cross-document alignment relations).

It has been a long road since the big pioneer annotation campaigns like the Penn Treebank (Marcus et al., 1993), but one problem remains: manual annotation is expensive. Various strategies have been implemented to reduce or control annotation costs. Tools have been developed to assist and guide the work of annotators. Automatic annotation methods, sometimes based on machine learning, have been introduced to relieve the annotator of the most trivial and repetitive work and to allow him/her to focus on the hardest annotation tasks where human interpretation is critical. For simple tasks, the use of crowdsourcing is developed with the idea of dividing up tedious work and exploiting the number of annotators to compensate for the heterogeneity of their competence and reliability. Significant efforts have also been made to develop evaluation protocols and to measure intra and inter-annotator agreements, which allow for a better control of the quality of the produced annotated data.

Despite all this work and the experience gained in annotation, we lack a global picture or an overall methodology to *a priori* determine the various costs of an annotation campaign (task definition, data preparation, recruitment and management of annotators, annotation itself, quality control, etc.), make the necessary compromises and choose the appropriate solutions to alleviate the annotators' work.

This article relies on the analysis of many annotation campaigns, which are described in the state of the art or were managed by the authors. It does not offer a tool or a ready-made solution to tell you "how to build your next annotation campaign". Instead, we propose an analytical framework, a grid of analysis, to understand the complexity of annotation tasks. It is based on a decomposition of these tasks into elementary ones and a decomposition of complexity into 6 dimensions that are orthogonal to each other except for one. We also provide concrete metrics to measure the complexity of annotation tasks, *a priori* when possible, from already annotated samples or by comparison with similar annotation tasks. Our approach is pragmatic, as our aim is to provide practical tools to analyze the complexity in all its dimensions. Obviously, it gives a simplified view of complex annotation tasks, but it enables to compare different campaigns on a common basis.

This article is as follows. Section 2 shows that the notion of complexity is present in many works in the state of the art but that no comprehensive analysis has been proposed so far. The third section presents our analysis of the complexity dimensions of annotation. Section 4 illustrates the practical benefit of this method of analysis and shows how to apply it on a complex task.

2 An important but implicit issue

If the question of the complexity of annotation tasks has deserved little attention as such, it is implicitly present in most of the issues related to annotation.

2.1 Feedback on large-scale campaigns

An effort has been made to document large-scale projects and encountered issues. For example, concerning the Penn Treebank, (Marcus et al., 1993) explains that the POS tagset has been largely reduced as compared to that of the Brown corpus, in order to eliminate the categories that could be deduced from the lexicon or the syntactic analysis. It is also noted that reducing the size of the tagset allows to reducing inconsistencies. Finally, in order to "avoid arbitrary decisions", the annotators were allowed to associate several categories to a same token. The same principle was used for the syntactic layer, allowing multiple binding in case of ambiguity. For the same reason, no distinction was made between arguments and circumstants. (Abeillé et al., 2003) presents the main issues encountered during the French Treebank annotation and the found solutions. One difficulty concerns multi-word expressions that the campaign managers finally chose to annotate both as multi-word expressions and as distinct elements to avoid "linguistic debates". Another interesting issue concerns the hierarchical structure of the tagset. It allows to simplify the annotation as most of the sub-categories end up being unambiguous once the main category is selected. In the speech domain, the MEDIA annotation campaign gave rise to an in-depth reflexion on the methodology (Bonneau-Maynard et al., 2005), but is addresses the management of annotation campaigns rather than the analysis of annotation tasks.

2.2 Good practices in manual annotation

Good practices have progressively emerged to tackle various aspects of annotation tasks. They have been, in particular, inherited from corpus linguistics. (Leech, 1993, 2005) present a list of what a specification for annotation should contain. It represents an effort toward the identification of several levels of difficulty encountered when making decisions during the manual annotation of a corpus: segmentation, inclusion of units (words or phrases) within other units (clauses or sentences), assigning categories to some textual fragments.

Annotation formats Recommendations were published within the framework of the standardization effort of the annotation formats, in particular in (Ide and Romary, 2006), that finally gave birth to the ISO 24612 standard. In this view, the authors are little concerned by manual annotation difficulties as such but, to represent annotations they identify complex structuring elements which are of interest here : segmentation (identification of continuous or discontinuous sequences of characters), several layers of annotations (for example morpho-syntactic, then syntactic), relations between these layers, overlapping issues.

Organization of annotation campaigns An overall schema of the organization of annotation campaigns has also emerged. It involves several actors, among which the client (who can, for example, be the organizer of an evaluation campaign), the annotation manager, that is the person who is going to write the annotation guide from a (clearly) expressed need, and the annotators themselves, who proceed from the annotation guide. The increasing use of so-called "non-experts", through crowdsourcing for example, introduced a further distinction between expert annotators (editors/curators in GATE Teamware (Bontcheva et al., 2010)) and annotators themselves. The multiplication of actors contributes to the difficulty to organize an annotation campaign but not to that of the annotation task *per se*. Note also that the analysis we propose here applies whatever the level of expertise of the annotators. Besides, the annotation guide is recognized as the keystone of annotation campaigns, as it defines what should be annotated. Here, we consider that the need is clearly defined and known from all the participants.

Annotation evaluation Studies concerning the evaluation of the quality of manual annotation allowed to identify some factors influencing inter- and intra-annotator agreements. (Gut and Bayerl, 2004; Bayerl and Paul, 2011) demonstrated that the inter-annotator agreement and the complexity of the annotation task are correlated (the larger the number of categories, the lower the inter-annotator agreement) and that the categories prone to confusions are in limited number. This brings out two complexity dimensions related to the number of categories and to the ambiguity between some categories. This ambiguity-related complexity dimension also appears in (Popescu-Belis, 2007). In the study of the inter-annotator agreements, (Krippendorff, 2004) identified a step of identification of the elements to annotate that is called "unitizing". Similarly, in the Proposition Bank project (Palmer et al., 2005), the organizers separated role "identification" from role "classification" to compute the inter-annotator agreement, in order to "isolate the role classification decisions" from the (supposedly easier) identification.

2.3 Insights from Cognitive Science

Few publications focus on the difficulties of manual annotation. In (Tomanek et al., 2010), the authors used an eye-tracking device to analyze and model the annotation cognitive complexity. Their experiment was carried out on a simple named-entity annotation task (persons, locations and organizations), with some pre-identification of complex noun phrases containing at least one potential named entity. They measured the influence of the syntactic and semantic complexities[1] as well as the size of context that was used by the annotators. The results show that the annotation performance tends on average to "correlate with the [semantic] complexity of the annotation phrase" and less so with its syntactic complexity, and that the size of the needed context also depends on the semantic complexity of the annotation phrase. However interesting, their conclusions only apply to simple named entity annotation task.

3 Measuring the complexity of an annotation task

Manual annotation requires the annotator to determine which units should be annotated and how. Measuring the complexity of an annotation task requires a detailed analysis of these localization and characterization operations.

We propose to analyze the complexity of elementary annotation tasks according to six dimensions: the first two (discrimination and delimitation) relate to the localization of annotations, while the next three concern their characterization (expressiveness, tagset dimension and ambiguity degree). The context is a sixth factor of complexity that impacts annotation decisions: it is presented here as a separate dimension for the sake of simplicity, even though it simultaneously affects discrimination, boundaries delimitation and disambiguation.

Analyzing a task along those six dimensions is artificial in the sense that annotators do not make separate decisions, but this analytic approach is meant for the management of annotation. It is independent from both the volume of annotations to be added and the number of annotators involved: these values participate in the cost of a task but not in its complexity, which is important to evaluate as early as possible in the annotation process.

3.1 Decomposition of annotation tasks

In an annotation task, one or several human annotators are asked to explicit how they interpret a source signal. The guidelines usually explain what kind of interpretation is expected and for which purpose. Depending on the task, the annotators are provided with a closed tagset or

[1] Respectively measured as the "number of nodes in the parse tree" and the "inverse document frequency of the words in the phrase according to a reference corpus".

are allowed to introduce any tag they find relevant. The annotators have to read the source document and tag some or all of its segments with one or several tags.

Instead of decomposing annotations tasks into *levels* or *layers* (Goecke et al., 2010), in order to analyze the task complexity, we propose to decompose a campaign into *elementary annotation tasks* (EATs). The complexity of the various EATs is computed independently and the complexity of a whole campaign is computed as the combination of the complexity of the elementary tasks. We consider that an annotation task can be decomposed into two or more EATs if the tagset itself is made of smaller independent ones. [2] This decomposition in EATs is formal in the sense that it is independent of the pragmatic organization of the work: the annotators can handle different EATs as separate steps on the source signal or all at once depending on the specific nature of the work and the tools they use. The decomposition in EATs does not result in a simplification of the original task as it is often the case for the Human Intelligence Tasks (HITs) to be performed by Turkers (workers) in Amazon Mechanical Turk (Cook and Stevenson, 2010).

To take a simple example, the annotation of gene renaming relations can be analyzed as a combination of two EATs. The first one identifies the gene names in the source signal. The second one relies on the first level of annotation and indicates which of the genes hold in a renaming relation. Obviously, the annotators can add both types of annotations at the same time, but the tagsets are independent and it is easier, from a formal point of view, to analyze the annotation task as a combination of two EATs than as a unique, but complex one.

3.2 What to annotate?

Localizing the units to be annotated consists in distinguishing what is to be annotated from what is not, and potentially adjusting the boundaries of the identified units.

3.2.1 Discrimination

In some annotation experiments, the question of "what to annotate" is straightforward, for instance when the units to annotate have already been marked in an automatic pre-annotation phase or when all the units are to be annotated, as in a POS-tagging task. However, for the annotators, the corpus is often a haystack within which they must find what to annotate, and discriminating what should be annotated from what should not is a complex task.

Identifying the units on which the annotation work should focus is all the more complex as the units to consider are heterogeneous. (Erk et al., 2003) emphasizes that semantic role annotation and discourse annotation mix several levels of segmentation (from less than a word to more than a sentence). As a simple example, it is easier to identify in a text negatively connoted adverbs, in particular, than all the negative expressions, as the latter can be words, phrases, or even entire parts of a text. In the second case several segmentation levels are actually to be considered. This gives a first scale of difficulty. An annotation task is considered difficult to the extent that the discrimination factor, defined by the following formula, is high:

$$Discrimination_a(F) = 1 - \frac{|A_a(F)|}{\sum_{i=1}^{n} |D_i(F)|}$$

where F is the flow of data to annotate, a is an annotation task, n is the number of segmentation levels that are potentially relevant, $|D_i(F)|$ is the number of units obtained during the

[2] Independence, here, means that the tags of two different tagsets are globally compatible (even if some specific combinations may be forbidden), whereas the tags of a single tagset are mutually exclusive (except for encoding ambiguity).

segmentation of F at level i and $|A_a(F)|$ is the number of units to be annotated in the relevant annotation task.

Intuitively, this measure indicates that the discrimination weight is high when the units to annotate or mark are "submerged" within others, and when the proportion of what is to be annotated ($|A_a(F)|$) as compared to what could be annotated or is "markable" ($\sum_{i=1}^{n} |D_i(F)|$) is low. The identification factor is 0 when all the units of the flow are to be annotated, and approaches 1 when only a few units are actually to be annotated while many could be annotated.

For the classification of pronouns as anaphoric or impersonal, the discrimination factor is 0 if all the occurrences of pronouns have been identified beforehand. By contrast, for gene renaming relations, the discrimination factor is high because only a small proportion of gene name couples participate in a renaming relation and are to be annotated as such.[3]

It is generally easy to estimate the number of units to be annotated – from the definition of the annotation task, on a sample of the relevant corpus or by reference to comparable tasks – but more difficult to estimate what could be annotated. This requires the choice of a reference segmentation that divides the flow to annotate into units, of which some are to be annotated, and others not. This reference segmentation can be chosen in different ways:

1. The simplest solution is to rely on an obvious segmentation which can be automatically computed or which is intuitive for the annotator, even if it implies reviewing the boundaries of the units to be annotated which do not correspond to this segmentation. For example, with named entities, starting from a segmentation into words leads to consider compound named entities as "modified" units. This strategy reduces the discrimination weight, but increases the issue of boundary delimitation (see section 3.2.2 below).

2. When the units to annotate are too heterogeneous, several reference segmentations can be considered ($n > 1$ in the preceding formula): this increases the number of "markables", but avoids the need to modify many boundaries. In the case of named entities annotation, one could for instance consider all the words and all the phrases as "markables". This approach must be used if it seems less costly than the previous one.

3. Finally, the annotation task can be decomposed into several layers, corresponding to distinct EATs. In that case, the discrimination weights of the layers are computed independently, each one with a specific reference segmentation. Such a decomposition is needed only when the different types of "markables" resulting from the various segmentations are annotated differently. It would be artificial for instance to decompose the task of named entities annotation into several EATs if the same tagset is used for words and phrases.

3.2.2 Boundary delimitation

Identifying points of interest in the flow of data is not enough, as the elements to be annotated are often data segments. To identify what to annotate, one should also delimit the boundaries of the segment to be annotated.

Here again, the task is easy when a reliable reference segmentation can be computed. However, segmentations that can be computed automatically are often approximations and the annotator must locally modify the boundaries of the discriminated units: for instance, if one starts with a segmentation into words to annotate named entities or terms, the segmentation must be

[3] We assume here that the gene names are pre-annotated (or identified in a different EAT) and that all gene name couples in the same abstract are "markable", *i.e.* subject to expressing a renaming relation.

corrected for all the multi-word expressions. In most cases, delimiting the boundaries consists in enlarging or reducing the discriminated unit based on the reference segmentation, but there are also cases in which a discriminated unit should be decomposed in several units or where several contiguous discriminated units are grouped together into one annotation.[4]

Delimitation of boundaries represents a second complexity factor, $Delimitation_a(F)$, which is computed by comparing the segmentation obtained after the annotation to the reference segmentation. This factor is inspired from the slot error rate (Makhoul et al., 1999), a metric that is used in named entities recognition and classification and which takes boundary errors into account. Once the discriminated units before and after the boundary delimitation are optimally aligned, we can compute the following discrimination rate:

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

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.

This delimitation factor is worth 0 when no discriminated unit has been modified and it rises with the number of decompositions, groupings and boundary modifications performed by the annotator with respect to the reference segmentation. The value is limited to 1 to keep the same interval $[0, 1]$ for the six complexity factors.

The delimitation cost is measured *a posteriori*, but it can be estimated based on a sample of annotated corpus or by comparison with a similar task.

3.3 How to annotate?

Once the units have been discriminated and delimited, they have to be characterized.

3.3.1 Expressiveness of the annotation language

We distinguish between three types of annotation languages: type, relational and higher order languages.

In the simplest case, annotation consists in associating a type with a data segment, *i.e.* in labeling it. Many annotation tasks rely on such a type language: words of a text are associated with part-of-speech, speech turns are associated with interlocutors or with rhetorical functions, phrases are associated with named entities types. In some cases, the label that is used is itself structured, as with morpho-syntactic labels associating a part-of-speech with a lemma and its morpho-syntactic features, but such structuring increases the size of the tagset (see section 3.3.2) without changing the expressiveness of the annotation language.

Establishing relations between units is also a common task, but it is more complex. The complexity of relational annotations should not be underestimated: even if annotators do not always proceed in exactly this way, they have to locate and type the arguments of the relation, discriminate the couples, triplets, and more generally the n-uplets of segments to annotate among the set of n-uplets that are markable, and finally type the relation existing between the elements of the n-uplet.

[4] We neglect here the case of discontinuous units, which are often not annotated as such and which are not numerous enough to impact the present analysis.

A higher-order language is used when annotations are added to other annotations, but the complexity of this type of language, which is difficult to formalize and manipulate, is such, that it is often reduced to a simpler language. For example, to qualify an annotation as uncertain, the generally preferred option is to increase the tagset size so as to add the qualifier as an attribute associated with a main type. Alternatively, the decomposition of an annotation task into EATs also allows to qualify or relate, in a later step, annotations added in a previous one: you obtain several EATs, each one relying on a first order annotation language.

The degree of complexity entailed by the expressiveness of the annotation language is naturally represented by an ordinal qualitative scale, but for the sake of homogeneity with the previous factors, described by numeric values, we associate the different levels of expressiveness with graduations on a numeric scale from 0 to 1. In this scale, 0.25 corresponds to type languages, 0.5 and 0.75 to relational languages, respectively of arity 2 and higher than 2, while the maximal value, 1, is dedicated to higher-order languages.

3.3.2 Tagset dimension

The annotator generally selects the value of an annotation in a predefined tagset that is presented in the annotation guide and this choice is all the more difficult to make as it is open. The size of the tagset is therefore a new factor of complexity.

In the simplest case, the choice is boolean and annotating amounts to assigning the discriminated units into two categories:[5] sentences may be marked as relevant or not; the occurrences of the *it* pronoun are marked as anaphoric or impersonal.

However, the choice is often more open, for instance for representing the diversity of morpho-syntactic units, annotating syntactic dependencies or typing named entities. For the richest annotations, structured labels are often proposed: the annotator adds several labels on a single given unit, the combination of which forms the final annotation (Dandapat et al., 2009).

Finally, there are annotation tasks for which the choice of a label is entirely left to the annotator, as in speech transcription, where there may be as many labels as words. In such cases, we consider that we have a huge tagset, even though the annotation effort is probably of a slightly different nature for the annotator.

If an annotation A is formed of a sequence of m labels ($A = E_1 E_2 \ldots E_m$) and each label E_i can take n_i different values, the complete tagset theoretically contains n different labels, with $n = n_1 * n_2 * \ldots * n_m$. However, in practice, constraints are defined which reduce the number of combinations: the annotator does not have to choose one label from n at once, but instead first select 1 label among n_1 labels, then 1 among at most n_2, etc. up to 1 among at most n_m. The size of the tagset does not depend on the total number of possible labels but on the degrees of freedom of the successive choices that the annotator has to make. 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 i^{th} label ($\nu_i = n_i - 1$).[6] For instance, the tagset for the POS part of the Penn Treebank contains 36

[5] This boolean annotation is similar to the discrimination task, though the units to be annotated should be not only located but labeled.

[6] The formula gives a high boundary of the global degree of freedom because the choice of the i^{th} label is often constrained by the labels already added, so the annotator has in practice a degree of freedom that is less than $(n_i - 1)$, if n_i is the number of available labels at this point.

tags (Santorini, 1990), so ν equals 35, but as some tags are subtypes of others (like JJR and JJS for JJ) there are 21 in fact tags corresponding to main categories, and, as the maximum number of subtypes is 6 (for verbs), we may consider that $\nu = 20 + 5 = 25$.

The tagset dimension can 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 label 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.

The tagset dimension is worth 0 for the binary tagsets presenting a degree of freedom of 1 and it increases with the size of the tagset and the correspondingly increasing degree of freedom. It is worth 0.5 for the Penn Treebank POS tag annotation (0.7, if we consider the tagset as flat). It reaches a ceiling at 1. Annotation tasks with large tagsets ($> \tau$) are very difficult to manage.

3.3.3 Degree of ambiguity

The need to disambiguate the units to annotate introduces a fifth complexity factor, which is more difficult to estimate than the previous ones. As the role of the annotator is precisely to resolve ambiguous cases whenever possible, ambiguities are difficult to observe. Still, we can evaluate the ambiguity degree the annotator must resolve for a given task in two ways.

The first method consists in measuring the residual ambiguity degree by observing the traces left by the annotator during the annotation: the annotation protocol may allow the annotator to annotate with several labels in case of ambiguity, to use an under-determined label, or even to add an uncertainty feature to the chosen label. This allows measurement of the degree of residual ambiguity:

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

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, this residual degree of ambiguity can only be measured *a posteriori*, once the annotation has been performed, or from a sample of it.

The degree of residual ambiguity is worth 0 when no ambiguity mark was added by the annotator and would be worth 1 in the case (in real life, absurd) where all the annotations were marked as ambiguous, in one way or another. Obviously, depending on the traces which are used to compute it and on the directions given to the annotators, this metric can be more or less reliable and it should be associated, whenever possible, with results from another method.

This second method consists in measuring the theoretical degree of ambiguity for the tasks where several occurrences of the same vocables (vocabulary units) are annotated: this method applies to morpho-syntactic annotation or to semantic disambiguation but not to speech turn analysis or gene renaming relations. This metric relies on the idea that ambiguous vocables have occurrences that are annotated differently at different locations in the flow of data. The ambiguity factor is given by the proportion of units to be annotated that correspond to ambiguous vocables. The theoretical ambiguity can be measured from a dictionary that lists the possible labels for all the vocables, if the annotation relies on such a dictionary or, directly,

on a sample of annotated text. The theoretical ambiguity also depends on the frequency of ambiguous vocables in the flow of data to be annotated. It is computed in the following way:

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

with

$$Ambig_a(i) = \begin{cases} 1 & \text{if} \quad |Labels_a(i)| > 1 \\ 0 & \text{else} \end{cases}$$

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 labels available for the vocable i for the annotation task a.

When there is no ambiguous vocable, $|Labels_a(i)|$ is worth 1 and $Ambig_a(i)$ is worth 0 for every i, the annotation task is trivial and can be easily automated by projecting on the flow of data a dictionary establishing the correspondence between the vocables and their labels. In this case, $Ambiguity_{Th,a}(F)$ is worth 0. Conversely, if all the vocables are ambiguous, $Ambiguity_{Th,a}(F)$ is worth 1. Note that the weight of an ambiguous vocable influences the degree of theoretical ambiguity in proportion to its frequency.

Theoretical ambiguity tends to overestimate the weight of the ambiguity for the annotator as some ambiguities are probably trivial to solve.

3.4 The weight of the context

The weight of the context is a sixth complexity factor. Even though it is not independent from the previous factors as they were from each other (it makes discrimination, delimitation and disambiguation more complex for the annotator), we represent it here as such, for the sake of simplicity.

The complexity of an annotation task increases with the size of the window to take into account around the unit to annotate and with the number of knowledge elements to be rallied. While it is difficult to determine the number of words participating in the resolution of an annotation task and *a fortiori* the number of knowledge elements at issue[7], we can identify two qualitative scales: the size of the data to be taken into account around the unit to be annotated and the degree of accessibility of the sources of knowledge that are consulted.

For the sake of homogeneity with the previous complexity factors, we translate the two preceding qualitative scales into a common discreet scale from 0 to 1:

- 0 corresponds to cases where no data around the unit to be annotated and no additional knowledge come into play. This is a theoretical case, since an annotation task where no context whatsoever is needed should actually be automated.
- Conversely, 1 is for the most complex cases, where the whole data flow and exterior sources of knowledge are necessary to annotate the units. The gene renaming relations annotation is one such, as the renaming relation, often hardly noticeable in the data flow, can often be confused with an isotopy (resemblance) relation or the membership in a common family. One must read the whole abstract to determine the semantics of the relation and sometimes the annotators must refer to an external source to better understand the properties of the genes and support their decision.

[7] Assuming that the elements of knowledge are countable, which is obviously an oversimplification.

- We estimate at 0.25 the weight of the context in two cases: 1) if the annotation depends on the immediate environment of the unit to be annotated, or 2) if only provided sources, such as annotation guidelines, are to be consulted.

- The complexity is evaluated at 0.5 in three cases: 1) if the two previous difficulties combine; 2) if a larger part of the data is needed; or 3) if a well-identified exterior source of knowledge is to be consulted.

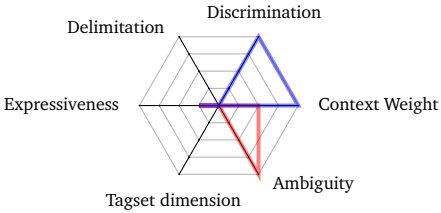
- Finally, the complexity is evaluated at 0.75 in three cases: 1) if the annotator must read the whole sentence and consult exterior sources to add annotations; 2) if s/he must take the entire flow of data into account; 3) if s/he must look for new knowledge sources.

This scale is obviously oversimplifying, but it is important to take that factor into consideration when planing an annotation campaign and the above criteria are meant for guiding the analysis and facilitating the comparison between complexity dimensions.

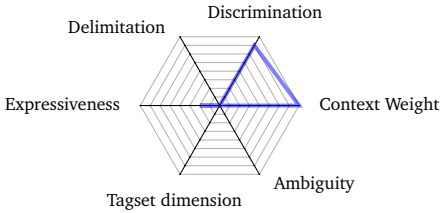
3.5 Synthesis

Since the six complexity factors are normalized on the same scale, once the complexity of different tasks is analyzed, it is easy to represent the various dimensions on a spider graph.[8]

Let us consider for instance the simple task, already mentioned, that consists in classifying pronoun occurrences as impersonal or anaphoric (REF OMITTED). Since the pronouns are previously tagged, both discrimination and delimitation are worth 0. The tagset being composed of two tags, this dimension is also at 0 and the expressiveness of the annotation language is 0.25 (type language). However, the ambiguity degree is high (1) as all occurrences are ambiguous. From our personal experience of annotation, we know that the context is worth 0.5 (or more) as the whole sentence must be considered to understand the role of the pronouns. The complexity of this task is represented on Figure 1a.



(a) Synthesis of the complexity dimensions of the pronouns classification (red) and gene names tagging (blue) campaigns



(b) Synthesis of the complexity dimensions of the whole gene names renaming campaign (2 EATs, double scale)

Figure 1: Two synthesis examples

The case of gene renaming (Jourde et al., 2011) is a little bit more complex and it is best analyzed as a combination of two EATs. Representing each EAT on a separate spider graph gives no idea of the whole task. We rather recommend to represent all EATs of a task in a single graph, which scale must be enlarged in proportion (from 0 to 2), as in Figure 1b. In the present case, the first EAT is the tagging of gene names in the sequence of words that composes the text. The discrimination factor is high (0.9), as only few words are gene names. Delimitation is 0, as

[8]The results presented in all the graphs in the article are rounded up/down to the nearest integer.

gene names, in our example, are simple tokens. On the contrary, the characterizing factors are low: the tagset is boolean (Dimension=0), a type language is used (Expressiveness=0.25) and ambiguity is very low as only few gene names are also common names (theoretical ambiguity can be approximated at 0.01[9] and residual ambiguity is on average of 0.04 for two annotators). In this case, the context is the highest factor as it is often necessary to read the whole PubMed abstract to understand the role of a mentioned entity and as annotators sometimes consult external resources (context weights 1) (Fort et al., 2010).

This first EAT is represented on the same graph as the pronoun classification, thus enabling the comparison of the two tasks (see Figure 1a). If they both show little complexity on three dimensions (delimitation, expressiveness, tagset dimension), the first one (pronouns) presents a high ambiguity dimension and no discrimination problem, while the second one (gene names) shows high complexity levels on the discrimination and context dimensions and no ambiguity. The solutions to alleviate the costs of these campaigns should therefore be adapted (pre-annotation and easy access to context for gene name tagging, and probably a carefully designed documentation for the pronoun classification).

Gene renaming consists in linking the gene name occurrences that hold a renaming relation. The whole task therefore comprises a second EAT, which consists in marking the renaming relations or its absence on any couple of gene names co-occurring in the same abstract. In that case, the discrimination is high (0.95) as only few gene couples are actually renaming each other. Delimitation is null since the gene names have already been annotated. The tagset is composed of 3 tags as the renaming relation is oriented (Dimension=0.04). Even if the annotations carry relational information, the language is a type language (a couple of gene names bears a tag, which expresses the direction of the relation). Ambiguity is very low (residual ambiguity is on average of 0.02 for the two annotators), but the context is high, as in the previous case. Figure 1b shows how the two EATs are combined to provide an analysis of the complexity of the whole task, which proves to be focused on discrimination and context.

4 Validation and illustration

4.1 Experimental validation

Although, as we showed in section 2, this grid of analysis has never been identified as such, some existing results or experiments confirm its applicability.

One first example of this is related to discrimination. In experiments led on the effects of pre-annotation on POS-tagging, the authors of (Fort and Sagot, 2010) showed that if the automatic pre-annotation is of very good quality, *i.e.* if only few tokens disseminated within the text have to be corrected, very good annotators can end up with less good annotation results, due to lapses in their concentration. This corresponds directly to the discrimination dimension we present here, which tends to get higher if the annotations to perform are submerged within the text.

Also, it has been observed that the quality of the annotation decreases with the number of tags involved (Bayerl and Paul, 2011). Structuring the tagset allows to reduce the degree of freedom at one point of choice and an appropriate annotation tool can help efficiently dealing with the hierarchy of tags (Dandapat et al., 2009).

Such an annotation tool can also help with complex language types, like relations, providing a graphical, user-friendly way to annotate them, like with Knowtator (Ogren, 2006). However, the number of manipulations involved is still higher than for simple type languages.

[9] To obtained this, we checked how many of the tagged gene names could also be tokens from the Brown corpus.

From the ambiguity point of view, the most obvious way to reduce it is to identify and remove the cases which are problematic. In the syntactic part of the Penn Treebank annotation, significant time was saved by reducing the ambiguity causes: "It proved to be very difficult for annotators to distinguish between a verb's arguments and adjuncts in all cases. Allowing annotators to ignore this distinction when it is unclear (attaching constituents high) increases productivity by approximately 150-200 words per hour." (Marcus et al., 1993). The analysis we propose here should help identifying these cases sooner in the process.

In the context of the gene renaming annotation campaign, we discovered that in some cases, the annotators needed the whole text to make their final decisions, and not only the abstract they had to annotate, as it was initially planned. If this need could have been identified beforehand, it would have been taken into account and the annotation tool would have been parameterized to give an easy access to the whole documents to annotators.

4.2 Example: structured named entities

We applied this analysis on a structured named entity annotation campaign for French described in (Grouin et al., 2011).

Structured named entities Within the Quaero program[10], a new definition of structured named entities was proposed. This new structure relies on two principles: the entities are both hierarchical and compositional. An entity is then composed of two kinds of elements: the 7 types (and 32 subtypes) which refer to a general segmentation of the world into major categories and the 31 components which allow to tag every word of the entity expression. Following this definition, the annotation campaign can be decomposed into two EATs: types and components. Figure 2 illustrates this definition on a French phrase. The types (EAT 1) are shown in red tags and the components (EAT 2) in blue tags.

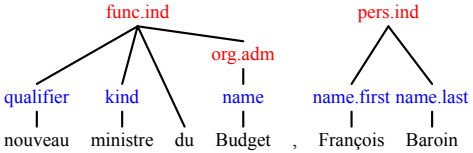


Figure 2: Multi-level annotation of entity types (red tags) and components (blue tags): *new minister of budget, François Baroin*.

Analysis For this illustration, we used the French spoken data provided by the mini-reference corpus (see (Grouin et al., 2011)) comprising 11,532 tokens, 1,161 entity types and 1,778 components.[11] Figure 3 represents the overall scores for this task using a spider graph.

The discrimination score is of 0.90 for the annotation of the types and 0.14 for the annotation of the components. To compute a delimitation score on such data, we first replaced in the annotated file every tag with a simple tag (*annot*) and in the same, but unannotated, file (as a reference file, under the hypothesis where tokens constitute the normal delimitation) we added the same tag around each token. Then we computed a slot error rate (SER) between these two files. The SER for types is over 100% (Delimitation=1) and the SER for components is 30% (Delimitation=0.3).

[10] www.quaero.org

[11] This mini-reference corpus is a sub-part of the whole corpus, that contains 1,291,225 tokens, 113,885 types and 146,405 components.

The annotation language is a type language, the degree of expressiveness is therefore of 0.25 for each of the EATs. Taking into account the types and sub-types structure, the total degree of freedom v for the annotation of the different types (EAT 1) is 10. The dimension score is then of 0.2. Concerning the components EAT, the total degree of freedom is 30 and the dimension score of 0.6. The theoretical ambiguity is computed for each EATs. The score is of 0.15 and 0.12 for the types and components respectively. These scores are low and the tasks do not seem very complex from the point of view of this dimension. The sample size is probably to blame here, as, if we compute this score *a posteriori* using the overall corpus (which is of course not possible in a real situation) the scores are of 0.49 for the components and of 0.36 for the types. Additional experimental results are necessary to take this scaling factor into account in the measure of ambiguity.

Concerning the weight of the context we estimate it at 0.75 because the annotators had to take into account the entire flow of data (the entity definition is contextual). Moreover, it was sometimes necessary to validate a choice by exploring external data (such as Wikipedia).

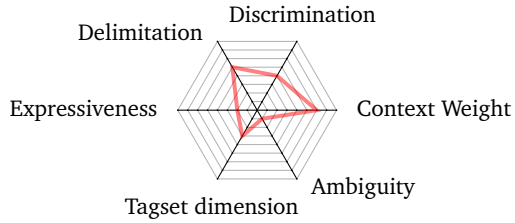


Figure 3: Synthesis of the complexity for the structured named entities campaign (2 EATs, double scale)

This analysis validates the choice of the hierarchical tagset that has been done for the annotation campaign described in(Grouin et al., 2011). Had a flat tagset been chosen, the dimension score would have been 1 ($v = 61$). Moreover, this analysis is in line with what was observed concerning the context weight within the campaign(Rosset et al., 2012). Of course, it presents some limits as it has been shown for the ambiguity score computation.

Conclusion

The grid of analysis we propose here should be used as part of the preparatory work of any annotation campaign, large or small. Obviously, when done *a priori*, on a small sample of annotations, the results will be approximate, but we believe that the analysis itself helps asking the right questions and finding the appropriate solutions.

Obviously, this pre-campaign work should be supported by an appropriate tool, so that this grid of analysis is computed more or less automatically. We are writing specifications for such a module, intended to be plugged into annotation management tools like Slate (Kaplan et al., 2010) or GATE Teamware (Bontcheva et al., 2010).

Acknowledgments

This work was realized as part of the Quæro Programme[12], funded by OSEO, French State agency for innovation.

[12] `http://quaero.org/`

References

Abeillé, A., Clément, L., and Toussenel, F. (2003). Building a treebank for French. In Abeillé, A., editor, *Treebanks*, pages 165 –187. Kluwer, Dordrecht.

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.

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

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, *Proceedings of the workshop on New Challenges for NLP Frameworks (NLPFrameworks 2010)*, Valletta, Malta. ELRA.

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, *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10)*, Valletta, Malta.

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.

Erk, K., Kowalski, A., Padó, S., and Pinkal, M. (2003). Towards a resource for lexical semantics: a large German corpus with extensive semantic annotation. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics - Volume 1*, ACL 03, pages 537–544, Morristown, NJ, USA. Association for Computational Linguistics.

Fort, K., François, C., and Ghribi, M. (2010). Evaluer des annotations manuelles dispersées : les coefficients sont-ils suffisants pour estimer l'accord inter-annotateurs ? In *Proceedings of the Traitement Automatique des Langues Naturelles (TALN)*, Montreal, Canada. 10 pages.

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

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.

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 *Proceedings of the 5th Linguistic Annotation Workshop*, pages 92–100, Portland, Oregon, USA. (poster).

Gut, U. and Bayerl, P. S. (2004). Measuring the reliability of manual annotations of speech corpora. In *Proceedings of the Speech Prosody*, pages 565–568, Nara, Japan.

Ide, N. and Romary, L. (2006). Representing linguistic corpora and their annotations. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation(LREC 2006)*, Genoa, Italy.

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 *Proceedings of the BioNLP Shared Task 2011 Workshop*, pages 65–73, Portland, Oregon, USA. Association for Computational Linguistics.

Kaplan, D., Iida, R., and Tokunaga, T. (2010). Annotation process management revisited. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC 2010)*, pages 365 – 366.

Krippendorff, K. (2004). *Content Analysis: An Introduction to Its Methodology, second edition*, chapter 11. Sage, Thousand Oaks, CA., USA.

Leech, G. (1993). Corpus annotation schemes. *Literary and Linguistic Computing*, 8(4):275–281.

Leech, G. (2005). *Developing Linguistic Corpora: a Guide to Good Practice*, chapter Adding Linguistic Annotation, pages 17–29. Oxford: Oxbow Books.

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, M., Santorini, B., and Marcinkiewicz, M. A. (1993). Building a large annotated corpus of English : The Penn Treebank. *Computational Linguistics*, 19(2):313–330.

Ogren, P. (2006). Knowtator: A plug-in for creating training and evaluation data sets for biomedical natural language systems. In *Protégé Conference*, Stanford, USA.

Palmer, M., Gildea, D., and Kingsbury, P. (2005). The proposition bank: An annotated corpus of semantic roles. *Computational Linguistics*, 31(1):71–106.

Popescu-Belis, A. (2007). Le rôle des métriques d'évaluation dans le processus de recherche en tal. *T.A.L. : Traitement automatique de la langue*, vol. 48, n. 1:67–91.

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 *Proceedings of the 6th Linguistic Annotation Workshop (LAW VI)*, pages 40–48, Jeju, Republic of Korea.

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, USA.

Tomanek, K., Hahn, U., Lohmann, S., and Ziegler, J. (2010). A cognitive cost model of annotations based on eye-tracking data. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, ACL '10, pages 1158–1167, Stroudsburg, PA, USA. Association for Computational Linguistics.