Confidence Measures for Alignment and for Machine Translation

Yong Xu

NNT : 2016SACLS270

THESE DE DOCTORAT
DE
L'UNIVERSITE PARIS-SACLAY
PREPAREE A
L'UNIVERSITE PARIS-SUD

ECOLE DOCTORALE N° 580
Sciences et Technologies de l'Information et de la Communication (STIC)

Spécialité de doctorat : Informatique

Par

M. Yong Xu

Confidence Measures for Alignment and for Machine Translation

Titre en français
Mesures de Confiance pour l'Alignement et pour la Traduction Automatique

Thèse présentée et soutenue à Orsay, le 26/09/2016

Composition du Jury :

M. ZWEIGENBAUM Pierre	Directeur de Recherche, LIMSI-CNRS	Président du jury
M. LANGLAIS Philippe	Professeur, Université de Montréal	Rapporteur
M. KRAIF Olivier	Maître de Conférences, Université Grenoble Alpes	Rapporteur
M. ESTÈVE Yannick	Professeur, Université du Maine	Examinateur
M. HUET Stéphane	Maître de Conférences, Université d'Avignon et des Pays de Vaucluse	Examinateur
M. YVON François	Professeur, Université Paris-Sud	Directeur de thèse

献给徐维平，熊华中，徐文文，以及周益娴。

Acknowledgements

The graduate school was for sure the most challenging life experience I have been through. I have lost plenty of hair during the last three years. Now that I look back, however, I am mostly amazed by the process, and realize how grateful I am towards a lot of people.

I owe a lot to Professor François Yvon, my thesis supervisor. When I first came to France as an undergraduate student, Prof. Yvon kindly offered me an internship opportunity to work on statistical machine translation. I was so shocked by the entire methodology that I spent two other Master internships in his group to grasp a better understanding, during which I realized the importance of alignment, and decided to work on it for a PhD thesis. I wasn't so well prepared, though, on both technical and mental aspects, leading to frustrations of Prof. Yvon during the four years. However, he has always kept his cool, always been supportive and encouraging, and has taught me, with great patiences, literally everything about research: how to formulate scientific ideas, how to design sound experiments, how to write clear papers, how to prepare a presentation and make slides, even how to correctly pronounce some words. For all these reasons, I have always kept high respect to Prof. Yvon. He is certainly the best thesis supervisor I can imagine of. And I consider myself tremendously lucky to be his student.

I sincerely thank the members of my thesis jury. In particular, I am very grateful to the two reviewers. Professor Phillippe Langlais is a recognized expert on the alignment problem, with a deep understanding on both theoretical and practical aspects. Doctor Olivier Kraif provided excellent analyses of my work from the point of view of corpus linguistics. Their reports were both very encouraging, both filled with insightful comments and suggestions. I thank Professor Yannick Estève and Doctor Stéphane Huet for having read carefully the manuscript, for the inspirational discussions during the defence, and for the warm encouragements. Doctor Pierre Zweigenbaum has honoured me by presiding the thesis jury. His questions and suggestions were very valuable. He has also made the effort of manually correcting things in the manuscript, which I appreciate fondly.

We were three in the interns' office in the summer of 2012: Nicolas, Benjamin and me. Khanh joined a bit later. And the four of us all began the graduate school together. I consider it a great privilege to pass the trajectory with such wonderful fellows, from whom I have learned a lot. The permanent members in the building, in particular Guillaume, Aurélien, Hélène, Marianna and Éric, are always ready to answer questions and provide

suggestions. I thank particularly Alexandre for tolerating me being a very silent office mate and asking naive questions, Tomas for explaining to me twice the design choices of his excellent Wapiti implementation of the CRFs. Other PhD students, both "olders" such as Nadi, Thiago, Haison and "youngers" such as Elena, Julia, Mattieu, Laurianne and Rachel, have together kept the TLP group a pleasant work environment. I apologize for being a bit drunk after one of our night drinks. Special thanks to Li, to whom I have asked numerous questions and have learned a lot.

I have stayed in France for seven years now. This is perhaps the period that I have learned the most. I thank France for having a beautiful territory and a nice people. The experiences in the French society have taught me to grow mature. Back in China, my father, mother and sister have always been warmly supportive, to which I can't express enough my gratitude.

Thanks to Yixian, the kind, beautiful, joyful and generous soul mate of mine.

Abstract

In computational linguistics, the relation between different languages is often studied through automatic alignment techniques. Such alignments can be established at various structural levels. In particular, sentential and sub-sentential bitext alignments constitute an important source of information in various modern Natural Language Processing (NLP) applications, a prominent one being Machine Translation (MT).

Effectively computing bitext alignments, however, can be a challenging task. Discrepancies between languages appear in various ways, from discourse structures to morphological constructions. Automatic alignments would, at least in most cases, contain noise harmful for the performance of application systems which use the alignments. To deal with this situation, two research directions emerge: the first is to keep improving alignment techniques; the second is to develop reliable confidence measures which enable application systems to selectively employ the alignments according to their needs.

Both alignment techniques and confidence estimation can benefit from manual alignments. Manual alignments can be used as both supervision examples to train scoring models and as evaluation materials. The creation of such data is, however, an important question in itself, particularly at sub-sentential levels, where cross-lingual correspondences can be only implicit and difficult to capture.

This thesis focuses on means to acquire useful sentential and sub-sentential bitext alignments. Chapter 1 provides a non-technical description of the research motivation, scope, organization, and introduces terminologies and notation. State-of-the-art alignment techniques are reviewed in Part I. Chapter 2 and 3 describe state-of-the-art methods for respectively sentence and word alignment. Chapter 4 summarizes existing manual alignments, and discusses issues related to the creation of gold alignment data. The remainder of this thesis, Part II, presents our contributions to bitext alignment, which are concentrated on three sub-tasks.

Chapter 5 presents our contribution to gold alignment data collection. For sentence-level alignment, we collect manual annotations for an interesting text genre: literary bitexts, which are very useful for evaluating sentence aligners. We also propose a scheme for sentence alignment confidence annotation. For sub-sentential alignment, we annotate one-to-one word links with a novel 4-way labelling scheme, and design a new approach for facilitating the collection of many-to-many links. All the collected data is released on-line.

Improving alignment methods remains an important research subject. We pay special attention to sentence alignment, which often lies at the beginning of the bitext alignment pipeline. Chapter 6 presents our contributions to this task. Starting by evaluating state-of-the-art aligners and analyzing their models and results, we propose two new sentence alignment methods, which achieve state-of-the-art performance on a difficult dataset.

The other important subject that we study is confidence estimation. In Chapter 7, we propose confidence measures for sentential and sub-sentential alignments. Experiments show that confidence estimation of alignment links is a challenging problem, and more works on enhancing the confidence measures will be useful.

Finally, note that these contributions have been applied to a real world application: the development of a bilingual reading tool aimed at facilitating the reading in a foreign language.

Key words: Confidence Measure, Confidence Estimation, Bitext Alignment, Sentence Alignment, Word Alignment, Annotation Scheme, Reference Corpus, Machine Translation

Résumé

En linguistique informatique, la relation entre langues différentes est souvent étudiée via des techniques d'alignement automatique. De tels alignements peuvent être établis à plusieurs niveaux structurels. En particulier, les alignements de bi-textes aux niveaux phrastiques et sous-phrastiques constituent des sources importantes d'information pour diverses applications du Traitement Automatique du Language Naturel (TALN) moderne, la Traduction Automatique étant un exemple proéminent.

Cependant, le calcul effectif des alignements de bi-textes peut être une tâche compliquée. Les divergences entre les langues sont multiples, de la structure de discours aux constructions morphologiques. Les alignements automatiques contiennent, majoritairement, des erreurs nuisant aux performances des applications. Dans cette situation, deux pistes de recherche émergent. La première est de continuer à améliorer les techniques d'alignement. La deuxième vise à développer des mesures de confiance fiables qui permettent aux applications de sélectionner les alignements selon leurs besoins.

Les techniques d'alignement et l'estimation de confiance peuvent toutes les deux bénéficier d'alignements manuels. Des alignements manuels peuvent jouer un rôle de supervision pour entraîner des modèles, et celui des données d'évaluation. Pourtant, la création de telles données est elle-même une question importante, en particulier au niveau sous-phrastique, où les correspondances multilingues peuvent être implicites et difficiles à capturer.

Cette thèse étudie des moyens pour acquérir des alignements de bi-textes utiles, aux niveaux phrastiques et sous-phrastiques. Le chapitre 1 fournit une description de nos motivations, la portée et l'organisation du travail, et introduit quelques repères terminologiques et les principales notations. L'état-de-l'art des techniques d'alignement est revu dans la Partie I. Les chapitres 2 et 3 décrivent les méthodes respectivement pour l'alignement des phrases et des mots. Le chapitre 4 présente les bases de données d'alignement manuel, et discute de la création d'alignements de référence. Le reste de la thèse, la Partie II, présente nos contributions à l'alignement de bi-textes, en étudiant trois aspects.

Le chapitre 5 présente notre contribution à la collecte d'alignements de référence. Pour l'alignement des phrases, nous collectons les annotations d'un genre spécifique de textes : les bi-textes littéraires. Nous proposons aussi un schème d'annotation de confiance. Pour l'alignement sous-phrastique, nous annotons les liens entre mots isolés avec une nouvelle catégorisation, et concevons une approche innovante de segmentation itérative pour faciliter

l'annotation des liens entre groupes de mots. Toutes les données collectées sont disponibles en ligne.

L'amélioration des méthodes d'alignement reste un sujet important de la recherche. Nous prêtons une attention particulière à l'alignement phrastique, qui est souvent le point de départ de l'alignement de bi-textes. Le chapitre 6 présente notre contribution. En commençant par évaluer les outils d'alignement d'état-de-l'art et par analyser leurs modèles et résultats, nous proposons deux nouvelles méthodes pour l'alignement phrastique, qui obtiennent des performances d'état-de-l'art sur un jeu de données difficile.

L'autre sujet important d'étude est l'estimation de confiance. Dans le chapitre 7, nous proposons des mesures de confiance pour les alignements phrastique et sous-phrastique. Les expériences montrent que l'estimation de confiance des liens d'alignement reste un défi remarquable. Il sera très utile de poursuivre cette étude pour renforcer les mesures de confiance pour l'alignement de bi-textes.

Enfin, notons que les contributions apportées dans cette thèse sont employées dans une application réelle : le développement d'une liseuse qui vise à faciliter la lecture des livres électroniques multilingues.

Mots clés : Mesure de Confiance, Estimation de Confiance, Alignement de Bi-textes, Alignement Phrastique, Alignement de Mots, Schème d'Annotation, Corpus de Référence, Traduction Automatique

Contents

Chapter 1

Introduction

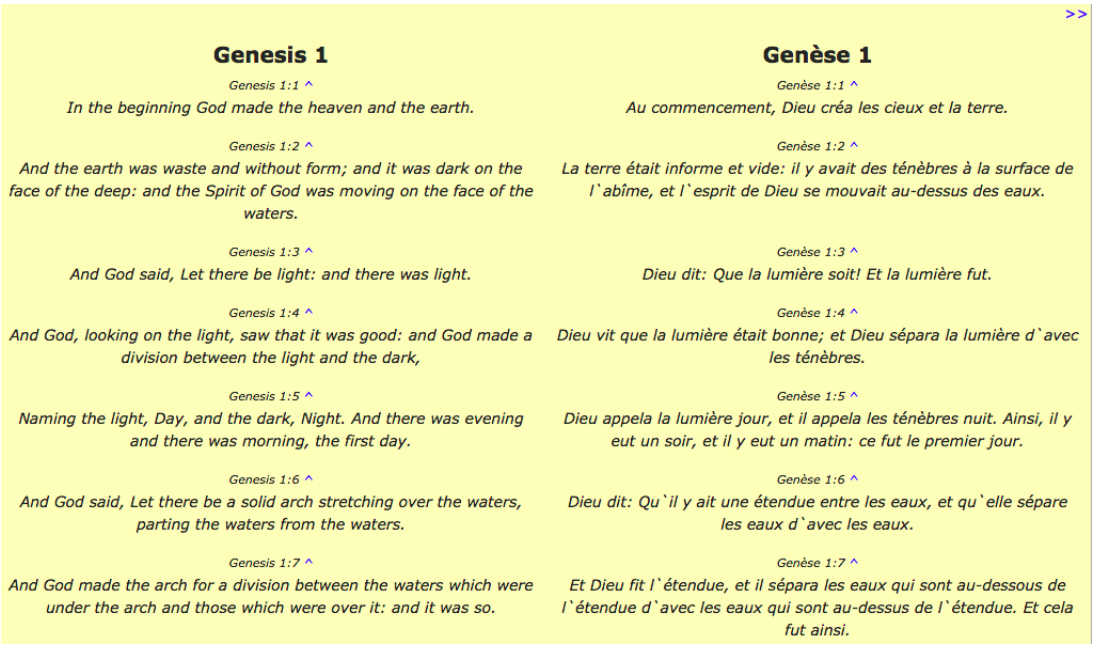
Automatic alignment refers to the task of establishing cross-lingual correspondences for multi-lingual comparable corpora. Such correspondences constitute important knowledge sources for cross-lingual studies. Modern Natural Language Processing (NLP) techniques make substantial uses of alignments. For instance, Machine Translation (MT) systems, both statistical [Koehn, 2010] and neural [Bahdanau et al., 2015], rely on large amounts of parallel sentences to train translation models. Statistical MT (SMT) further dives into fine-grained sub-sentential alignments.

Bitexts constitute an important subset of comparable corpora. We refer to a pair of texts of different languages as a *bitext*, if one text is the translation of the other, or both are translations of a (perhaps unseen) third one. Figure 1.1 displays an excerpt of an English-French bitext of the Bible.[1] We call each single text one side of the bitext, and refer to the two sides as *source* and *target* in case of necessity. While a translation process is in general directional (from the source to the target), we make no such restrictions in our discussions: we only impose that the two sides are indeed mutual translations (even poor ones). If this assumption does not hold, the task is no longer considered as bitext alignment. Mining sentential or sub-sentential correspondences from loosely-comparable corpora are interesting research problems in their own right. [Munteanu and Marcu, 2005; Prochasson and Fung, 2011] are two representative works, respectively at the sentence and at the word levels.

In this thesis, we focus on sentential and sub-sentential alignments of bitexts. Cross-lingual correspondences also exist at other structural levels: documents, paragraphs, characters, etc. Since our studies are based on bitexts, we will not discuss document alignment. Paragraph information is not always present in documents, so we will skip paragraph alignment as well.

Evaluation and confidence estimation (CE) of automatic alignments are also very relevant, as alignment quality impacts significantly the performance of downstream applications

[1] Taken from © `http://www.transcripture.com/`. Permission granted.



>>

Genesis 1

Genesis 1:1 ^
In the beginning God made the heaven and the earth.

Genesis 1:2 ^
And the earth was waste and without form; and it was dark on the face of the deep: and the Spirit of God was moving on the face of the waters.

Genesis 1:3 ^
And God said, Let there be light: and there was light.

Genesis 1:4 ^
And God, looking on the light, saw that it was good: and God made a division between the light and the dark,

Genesis 1:5 ^
Naming the light, Day, and the dark, Night. And there was evening and there was morning, the first day.

Genesis 1:6 ^
And God said, Let there be a solid arch stretching over the waters, parting the waters from the waters.

Genesis 1:7 ^
And God made the arch for a division between the waters which were under the arch and those which were over it: and it was so.

Genèse 1

Genèse 1:1 ^
Au commencement, Dieu créa les cieux et la terre.

Genèse 1:2 ^
La terre était informe et vide: il y avait des ténèbres à la surface de l`abîme, et l`esprit de Dieu se mouvait au-dessus des eaux.

Genèse 1:3 ^
Dieu dit: Que la lumière soit! Et la lumière fut.

Genèse 1:4 ^
Dieu vit que la lumière était bonne; et Dieu sépara la lumière d`avec les ténèbres.

Genèse 1:5 ^
Dieu appela la lumière jour, et il appela les ténèbres nuit. Ainsi, il y eut un soir, et il y eut un matin: ce fut le premier jour.

Genèse 1:6 ^
Dieu dit: Qu`il y ait une étendue entre les eaux, et qu`elle sépare les eaux d`avec les eaux.

Genèse 1:7 ^
Et Dieu fit l`étendue, et il sépara les eaux qui sont au-dessous de l`étendue d`avec les eaux qui sont au-dessus de l`étendue. Et cela fut ainsi.

Figure 1.1: An excerpt of a English-French bitext of the Bible.

(at least in most cases) [Lambert et al., 2005], and the amount of alignments employed in modern NLP systems is prohibitively large for manual evaluation. Automatic evaluation requires gold standard data and metrics, which, unfortunately, are not trivial to design for alignments, especially at the sub-sentential level. Confidence estimation aims at evaluating the quality of alignments without using gold data. While quality estimation of MT has received much research attention, less work has been done on confidence estimation of alignments. Yet this is an important question: reliable alignments prove useful for many applications, such as in cross-lingual transfer learning.

This thesis is organized as follows. In the first part, we review sentence-level (in Chapter 2) and word-level alignment (in Chapter 3), as well as the creation of evaluation resources (in Chapter 4). Compared to [Wu, 2010; Tiedemann, 2011] which cover similar subjects, our discussions concentrate on alignment methods based on probabilistic models, which facilitate subsequent developments of confidence estimation techniques. In the second part, we present our contributions. We first describe, in Chapter 5, novel annotation schemes for bitext alignment, at both sentential and sub-sentential levels, as well as the resources we created following these schemes. We then present our work on sentence alignment in Chapter 6, including a large-scale evaluation of existing tools, and two new methods which achieve state-of-the-art performances. Confidence measures for bitext alignments are discussed in Chapter 7.

The idea of bitext alignment can be generalized to multi-dimensional texts, that is, texts with more than two sides. At the sentence level, Simard [1999] presents an early study. Kumar et al. [2007]; Lardilleux et al. [2013]; Eger [2015] study sub-sentential alignments of multi-parallel texts.

Terminologies and Notations: we unify terminologies and notations that will be used throughout the thesis. We use Fraktur letters $\mathfrak{E}$ and $\mathfrak{F}$ to denote the two languages of a bitext. We frequently instantiate $\mathfrak{E}$ to be English and $\mathfrak{F}$ to be French for convenience. For a positive integer I, denote $[I] = \{i : i \in \mathbb{N}, 1 \leq i \leq I\}$ and $[I]_0 = \{i : i \in \mathbb{N}, 0 \leq i \leq I\}$, respectively for the set of strictly positive integers and the set of non-negative integers smaller than I. We note:

- *words*: a lower-case letter e or f. To indicate the index of a word in a sentence, we attach a subscript, such as e_i or f_j.
- *ordered sequences*: a bold letter representing a sequence of objects placed inside parentheses. If the sequence is continuous, we use a subscript and a superscript to shorten the notation. For instance, a continuous ordered sequence of words is $\mathbf{e}_o^q = (e_o, .., e_p, .., e_q)$.
- *sentences*: a capital letter E or F, with subscript if necessary. We view a sentence as a sequence of words, hence $E_m = \mathbf{e}_1^I = (e_1, .., e_i, .., e_I)$.
- *texts*: a sequence of sentences, e.g. $\mathbf{E}_1^M = (E_1, .., E_m, .., E_M)$ denotes a text of M sentences.
- *alignment links*: an alignment link contains a group of source units and a group of target units, with the two sides corresponding to each other. A link is denoted by two sequences inside brackets, separated by a semicolon. For example, a sentence alignment link can be $l = [\mathbf{E}_m^{m+1}; F_n]$. For one-to-one word alignment links, sometimes we use z_{ij} to indicate the link between the i-th source and the j-th target words.
- *null links*: if some units do not have a counter-part in the other side (e.g. not translated), they belong to null links, denoted as $[E_m;]$ or $[; F_n]$.
- *alignment*: an alignment is a either a set or an ordered list of alignment links. We use a caligraphic capital letter to represent an alignment.

For sentence alignment, we characterize an alignment link by the number of sentences on the two sides. This property defines *link type*. For instance, a link composed of three English and two French sentences is of the type 3:2. Importantly, this notion is directional: another link of two English and three French sentences has type 2:3. For word alignment, we often distinguish among null, one-to-one, one-to-many, many-to-one and many-to-many links. Again this notion is directional.

Part I

Bitext Alignment: Methodologies and Evaluations

Chapter 2

Sentence Alignment

The sentence level is a reasonable starting point for bitext alignment studies. First, sentences are relatively easy to identify in almost all documents, while other levels are less obvious, for instance, paragraph information could be missing due to issues such as poor formatting. Second, compared to paragraphs, sentence-level correspondences are non trivial, even difficult to establish, posing interesting research challenges. Sentence-level alignment enables the studies on finer-grain alignments, which is very relevant for many applications, the most prominent one being Statistical Machine Translation (SMT).

In this chapter, we give a thorough overview of sentence alignment. We start by informally describing the task, presenting several conventions, and making a formal definition of sentence alignment. Then we discuss state-of-the-art methods in detail. The conclusions contain our view on further developments of sentence alignment.

2.1 Sentence Alignment: Descriptions and Definitions

Given a bitext $\mathbf{E}_1^M = (E_1, ..E_m, .., E_M)$ (source side) and $\mathbf{F}_1^N = (F_1, .., F_n, .., F_N)$ (target side), where each E_m or F_n is a sentence, sentence alignment is the task of recovering sentence-level alignment links between the two sides, i.e. finding the corresponding sentence groups. An alignment link has two sides, each containing any number (including zero) of consecutive sentences. A basic example of sentence alignment, composed of three 1:1 links, is in Figure 2.1. Figure 2.2 displays a slightly harder case, where the English sentence E_2 aligns to two French ones (F_2, F_3), leading to a 1:2 link $[E_2; F_2, F_3]$. Figure 2.3 shows an alignment with a null link $[E_3;]$.

Before going into specifications and technical details, we should note that in order to perform sentence alignment, we must first define the notion of sentence and segment texts into sentences. In NLP applications, this step is typically done by some rule-based programs, such as the toolkit distributed with the Europarl Corpus [Koehn, 2005]. Perhaps surprisingly, sentence segmentation is not a trivial task. Exactly which kinds of text seg-

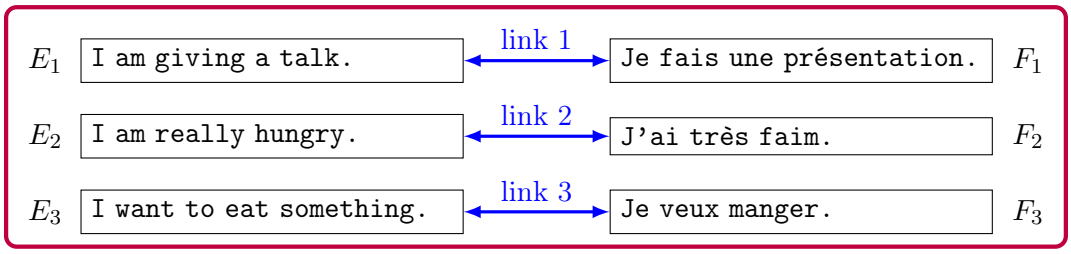


Figure 2.1: A basic example of sentence alignment.

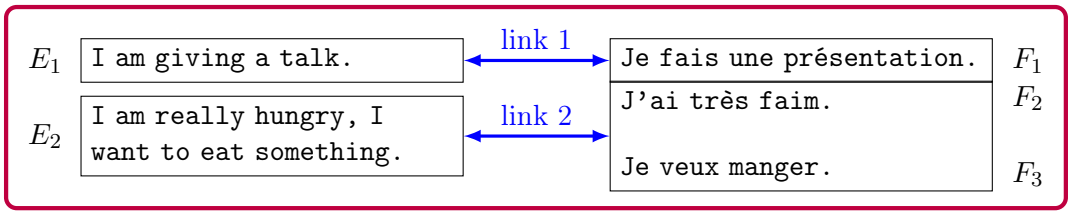


Figure 2.2: A sentence alignment with a 1:2 link.

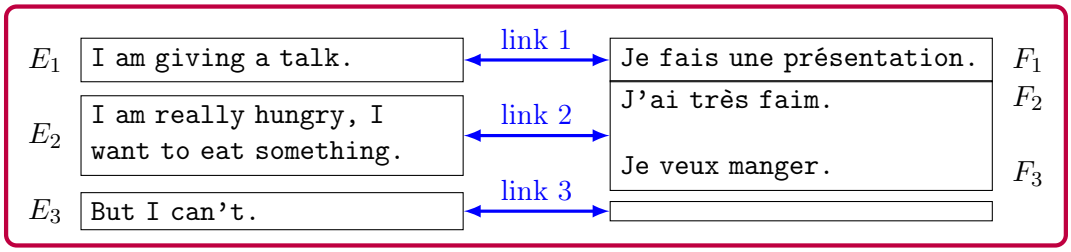


Figure 2.3: A sentence alignment with a null link.

ments can be considered as sentences is an issue that requires care to solve, and the rules can vary depending on languages [Simard, 1998]. Tokenization and segmentation errors have a direct impact on the quality of sentence alignments, even for manual annotations. Simard [1998] reports that, during the process of creating manual sentence alignments, most annotator disagreements are caused by questions of sentence segmentation rather than questions of translational equivalence. Indeed, in the BAF corpus created by Simard [1998], some "sentences" contain a single punctuation token, which might confuse the annotators. Depending on the application of sentence alignment, the segmentation issue can be more or less relevant. For tasks that require high-precision alignments, it might be helpful to be equipped with a clear definition of sentences, and to manually check the sentence segmentations. We note this issue here, but in the work presented in this thesis, we will assume the tokenization and the segmentation have been done before the alignment process, to simplify our discussions. It will be interesting to study quantitatively the level of importance of this problem by manual evaluations.

In most cases, the translation from one language to another is performed in a relatively stable, monotonic way, but there exist also translation choices which make sentence alignment difficult. For example, insertions and/or deletions of a small block of sentences are not rare. Translators make these choices because of many possible reasons, such as editorial policies. In this thesis, we view these less regular cases as natural phenomenons, and deal with them in the alignment process, without trying to understand why they happen. We refer interested readers to [Lecluze, 2011] which gives a discussion on this subject.

Whether or not the final list of alignment links should cover all the sentences in the bitext depends on the application. Most, if not all, early sentence alignment research is driven by SMT, which requires large amounts of parallel sentences as the training corpus. For this purpose, it is not essential to cover all sentences, rather, only high-confidence pairs should be extracted. In other scenarios, such as translation validation [Macklovitch, 1994b] and cross-lingual reading [Yvon et al., 2016], the whole bitext should be covered.

2.1.1 Conventions of Sentence Alignment

The research community has reached three main conventions for sentence-level alignments [Simard, 1998; Tiedemann, 2011]:

- Each side of an alignment link should be a *flat contiguous segment*, that is, a consecutive group of sentences, unless it is empty. If E_i and E_{i+2} are both inside a link, then so must be E_{i+1}. Figure 2.4 illustrates this rule.
- Alignment links are *monotonic*. If $[E_i; F_j]$ is a link, then no source sentences *following* E_i (e.g. E_{i+1}) can link to target sentences *preceding* F_j (e.g. F_{j-1}). This is illustrated in Figure 2.5.
- Links must be *minimal*, meaning that they cannot be decomposed into strictly smaller links that do not violate the previous constraints. For example, if both $[E_i; F_j]$

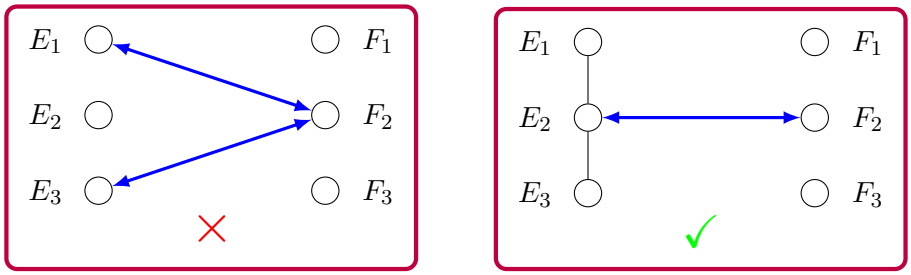


Figure 2.4: In the left figure both E_1 and E_3 align to F_2 but not E_2, which makes the English side discontiguous; the correct alignment is on the right, where we include E_2 into the link (even though E_2 is not translated in F_2).

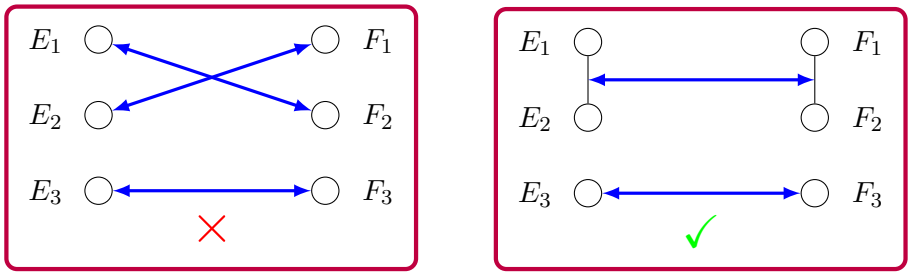


Figure 2.5: In the left figure the monotonicity is violated; the correct alignment is in the right figure, where one 2:2 link, rather than two crossing 1:1 links, has been created.

and $[E_{i+1}; F_{j+1}]$ are good alignment links, then it is incorrect to form a larger link $[\mathbf{E}_i^{i+1}; \mathbf{F}_j^{j+1}]$. Furthermore, a sentence belongs to exactly one (null or non-null) alignment link.

The three conventions are well motivated, though from different perspectives. The first two are based on observations of bitexts. Sentences in bitexts exhibit stronger translational regularities than words and phrases. They are generally translated in monotonic order, and for some genres of texts, most sentences are translated one by one. The third convention is a typical rule of fine-grain alignments.

These conventions together turn sentence alignment into a well-defined computational problem, and lead to significant computational simplifications. To our knowledge, most automatic sentence alignment systems adopt these three conventions. Two recent studies of Quan et al. [2013]; Zamani et al. [2016], however, drop the flat contiguous segment and monotonicity constraints, significantly increasing the decoding complexity. To deal with this problem, Quan et al. [2013] assume that only three types of alignment links exist: 0:1, 1:0 and 1:1, thus avoiding to consider non-contiguous segment. They further impose that one sentence can be aligned to at most one sentence of the other language. This rules out 1-to-many and many-to-many links, hence is a strong simplification. Zamani et al. [2016]

use Integer Programming to formulate the problem, and perform approximative inference.

2.1.2 Bitext Space

A geometrical interpretation of bitexts will prove useful, as one can visualize bitext alignment (or more broadly, bitext mapping) techniques and get better understandings. Melamed [1999] proposed the notion of *bitext space*. A bitext is viewed as a rectangle, where the lower left corner represents the beginnings of both sides, and is called the *origin* of the bitext space. The upper right corner represents the ends, and is called the *terminus*. The straight line between the origin and the terminus is referred to as the *bitext diagonal*. The two sides of the bitext expand respectively on the two axes. Without loss of generality, we assume that the source text lies on the horizontal axis, and the target lies on the vertical. The lengths of the axes are the lengths of the corresponding text, measured in number of characters (a space is also considered as a character). For each token, Melamed [1999] defines its position as the index of its middle character. Figure 2.6 shows a bitext space example.

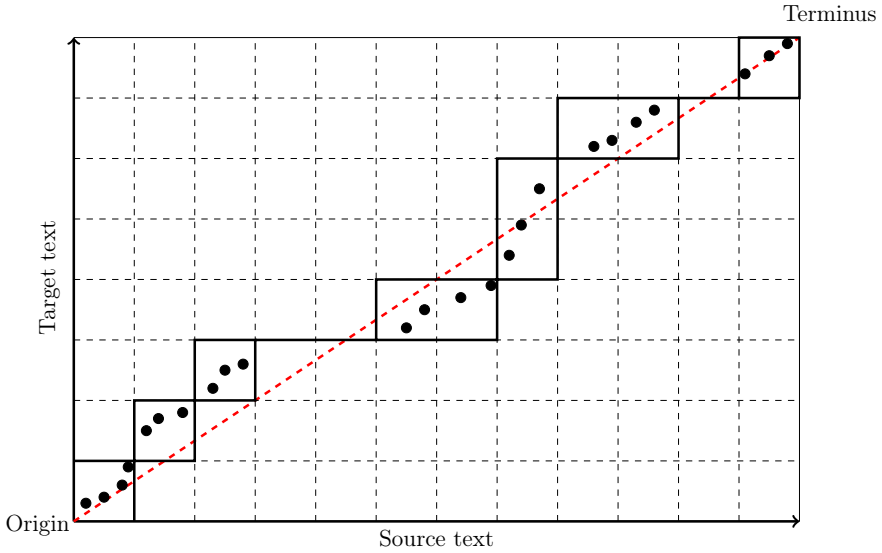


Figure 2.6: An alignment in a bitext space. The source (resp. target) text lies on the horizontal (resp. vertical) axis. The red dashed line is the bitext diagonal. Filled dots are corresponding text units (e.g. a pair of words that are mutual translations). A sub-rectangle represents an alignment link, whereas a horizontal (resp. vertical) line represents a null link without target (resp. source) element. The chain of sub-rectangles and the horizontal lines together constitute a full alignment of the bitext.

In the bitext space representation, the sentence alignment problem is converted into finding a chain of suitable sub-rectangles in the space, each one representing an alignment

link. According to the conventions, this chain of sub-rectangles should cover all sentences on both sides, should not overlap with each other, and should satisfy the monotonicity constraint. For null links, the corresponding sub-rectangles reduce to horizontal or vertical lines. This interpretation of sentence alignment is also illustrated in Figure 2.6.

2.1.3 A Formal Definition

We can give a formal formulation of sentence alignment based on the descriptions above, which is necessary for technical discussions. We start with a few definitions in the context of bitexts.

Definition 2.1.1 (Sentence Segment) *A sentence segment is an ordered list of consecutive sentences appearing in the same order as in the text. We note a sentence segment as* $\mathbf{E}_o^q = (E_o, .., E_{p-1}, E_p, .., E_q)$*, with* $o, p, q \in \mathbb{N}_+$ *being sentence indices in the text, satisfying* $o \leq p \leq q$*. A sentence segment* $\mathbf{E}_o^q$ *can be empty, denoted as* $\mathbf{E}_o^q = \varnothing$*.*

We define a function $\texttt{NUM\_SENT}(\cdot)$ over sentence segments which returns the number of sentences inside a sentence segment:

$$\texttt{NUM\_SENT}(\mathbf{E}_o^q) = \begin{cases} q - o + 1 & \text{if} \quad \mathbf{E}_o^q \neq \varnothing \\ 0 & \text{if} \quad \mathbf{E}_o^q = \varnothing \end{cases}$$

We further define a strict order over sentence segments. For two segments $\mathbf{E}_{o_1}^{q_1}$ and $\mathbf{E}_{o_2}^{q_2}$ extracted from the same text, we have:

$$\mathbf{E}_{o_1}^{q_1} \prec_s \mathbf{E}_{o_2}^{q_2} = \begin{cases} \texttt{true} & \text{if} \quad o_2 > q_1 \\ \texttt{true} & \text{if} \quad \mathbf{E}_{o_1}^{q_1} = \varnothing \text{ or } \mathbf{E}_{o_2}^{q_2} = \varnothing \\ \texttt{false} & \text{otherwise} \end{cases}$$

Definition 2.1.2 (Sentence Alignment Link) *A sentence alignment link is composed of a source language sentence segment and a target language sentence segment. At most one of the two sequences can be empty. We denote a sentence alignment link as* $l = [\mathbf{E}_o^q; \mathbf{F}_r^t]$*, thus* l *must satisfy* $\texttt{NUM\_SENT}(\mathbf{E}_o^q) + \texttt{NUM\_SENT}(\mathbf{F}_r^t) > 0$*.*

We also define a strict order over sentence alignment links:

$$[\mathbf{E}_{o_1}^{q_1}; \mathbf{F}_{r_1}^{t_1}] \prec_l [\mathbf{E}_{o_2}^{q_2}; \mathbf{F}_{r_2}^{t_2}] = \begin{cases} \texttt{true} & \text{if} \quad \mathbf{E}_{o_1}^{q_1} \prec_s \mathbf{E}_{o_2}^{q_2} \text{ and } \mathbf{F}_{r_1}^{t_1} \prec_s \mathbf{F}_{r_2}^{t_2} \\ \texttt{false} & \text{otherwise} \end{cases}$$

Definition 2.1.3 (Sentence Alignment) *A sentence alignment is an ordered list of sentence alignment links* $\mathcal{A} = \mathbf{l}_1^K = (l_1, ..l_{k-1}, l_k, .., l_K)$*, with* $l_{k-1} \prec_l l_k \ \forall k : 1 < k \leq K$*.*

With these definitions, we can turn sentence alignment into a computational problem. For a bitext $[\mathbf{E}_1^M; \mathbf{F}_1^N]$ and a generic scoring function `SENT_ALIGN_SCORE`, we can formulate the following optimization problem:

$$\begin{array}{ll} \max\limits_{\mathcal{A}=(l_1,..,l_k,..,l_K)} & \texttt{SENT\_ALIGN\_SCORE}(\mathcal{A}, \mathbf{E}_1^M, \mathbf{F}_1^N) \\ \texttt{subject to} & l_{k-1} \prec_l l_k \quad \forall k : 1 < k \leq K \end{array} \tag{2.1}$$

This definition is very general. Basically, it imposes the first two conventions in § 2.1.1. The third one (the minimal link rule) can only be implicitly incorporated into the scoring function. We have not asked the alignment to cover all sentences of the bitext, since some methods presented later do not have this property. But it is straightforward to extend this optimization problem to incorporate the completeness constraint.

2.2 Overview of Sentence Alignment Methods

In § 2.1.3, we have formulated a constrained optimization framework for bitext sentence alignment, which can be decomposed into two sub-problems: alignment scoring and decoding. To solve the first one, we need to come up with a reasonable scoring scheme for alignments, while the second one is a search problem, which consists of searching for the best alignment among all possibilities, a combinatorial problem that is generally very difficult. Fortunately, real world bitexts often have several properties that can be used to reduce the computational complexity. During two decades of developments of bitext alignment, the research community has improved its procedures with several practical techniques. These techniques play important roles in the construction of real sentence alignment systems. So we detail them first, before going into the two central problems: alignment scoring and decoding.

2.2.1 Practical Techniques

We present several techniques that are widely used in the community. More precisely, what we will discuss are technical principles, since each of them can lead to many kinds of practical implementations.

Anchor Points in the Bitext Space

Anchor points, or pairs of units that are known to be parallel, constitute an important piece of information for bitext alignment. When aligning long parallel texts, a particular difficulty is the propagation of errors. That is, a mistake made in the middle of the alignment process might propagate to all downstream actions. If in the bitext there exist some pairs of units guaranteed to be parallel, then they must form true alignment links. For sentence alignment, according to the monotonicity constraint (see § 2.1.1), no other links could overlap with true

alignment links. Essentially, the alignment process terminates at these links, and resume after them. Hence, mistakes made before these points stop being propagated. Anchor points can be in different forms: a pair of words, a pair of sentences, even a pair of mark-ups, etc.

In practice, anchor points can significantly alter system performance. Gale and Church [1991] align the Union Bank of Switzerland (UBS) reports in English, French and German. The corpus contains reliable paragraph boundary marks, which divide 725 sentences into 118 paragraphs (6.14 sentences per paragraph on average). The program aligns sentences within paragraphs, and obtains a full alignment composed of 1316 links, among which 55 are wrong. They then run the same program on the whole bitext without the paragraph boundaries, and find a threefold degradation in performance : the error count is increased from 55 to 170. Koehn [2005] also takes advantages of paragraph boundaries to align the Europarl corpus.

Anchor points have been used since the very beginning of sentence alignment research, in the early 1990s. Most early work aimed at extracting large amounts of parallel sentences from large corpora for applications such as SMT [Brown et al., 1991] and bilingual lexicography [Klavans and Tzoukermann, 1990]. Although the translations in the corpora were relatively regular, their sizes still challenged the search process of sentence alignment, due to the limited computational power back then. This situation required researchers to employ simple heuristics to align the corpora, which were naturally prone to mistakes. Fortunately, the corpora used were mainly parliament debate recordings, with more or less mark-ups inserted to provide meta-information about the debates, such as session beginning time, speakers' names, paragraph endings, etc. These mark-ups were present in both languages. Some of them could easily be identified as anchors. Figure 2.7 shows an excerpt from the Hansard corpus (Canadian parliament debate recordings, in English and French), taken from [Brown et al., 1991].[1] The question number in line 62 is an obvious anchor, while the "Paragraph" tokens are more difficult to be determined as anchors, since there are too many of them (e.g. in lines 21 and 66) on both sides and they are not numbered, thus the correspondences are less obvious to establish.

Such explicit anchor points do not always exist, so other ways of identifying anchors have been explored. If one sentence on one side is exactly the same as one on the other side, then they might constitute anchors. This can be the case, for example, in dialogues referring to named entities, time, numbers, etc. Caution must be taken for these examples, though, as they can easily introduce false negatives if repeated. On the contrary, if one token appears only once on one side, and its verbatim copy appears on the other, then the sentences containing them tend to be mutual translations.

Another way to construct anchor points is through a multi-pass strategy, where the first pass finds high confidence links that can be used in the following. Brown et al. [1991] align the Hansard corpus, in which the mark-ups (especially the mark-up "Paragraph") exist in the form of special comments, but no reliable links between them are present, thus they can

[1] Unfortunately, the French part is missing. But it resembles the English part.

```
21. \SCM{} Paragraph \ECM{}
22. \SCM{} Author = Mr. Speaker \ECM{}
23. The hon. member's motion is proposed
    to the House under the terms of
    Standing Order 43.
44. Is there unanimous consent?
45. \SCM{} Paragraph \ECM{}
46. \SCM{} Author = Some hon. Members
    \ECM{}
47. Agreed.
61. \SCM{} Source = Text \ECM{}
62. \SCM{} Question = 17 \ECM{}
63. \SCM{} Author = Mr. Mazankowski
    \ECM{}
64. 1.
65. For the period April 1, 1973 to
    January 31, 1974, what amount of
    money was expended on the operation
    and maintenance of the Prime
    Minister's residence at Harrington
    Lake, Quebec?
66. \SCM{} Paragraph \ECM{}
```

Figure 2.7: An excerpt of the English Hansard corpus, reproduced from [Brown et al., 1991].

not readily be used as anchors. Brown et al. [1991] dedicate the first step of the algorithm to align the comments and pick the good scoring pairs as anchors. They then align actual sentences between the anchor points. In [Yu et al., 2012a], the first pass is to run the method of [Moore, 2002], which extracts 1:1 alignment links with high precision. These links are used as anchors, and the following steps consist to align the blocks between them. Similarly, Li et al. [2010a] choose high-confidence 1:1 links from the result of the algorithm of [Brown et al., 1991].

Cognates

For related languages, cognates provide reliable, low-cost word-level alignments, thus they can help sentence-level alignment in various ways. Cognates can be used as anchor points. [Simard et al., 1993a] use (word-level) cognates as an indicator of sentence alignment link quality. Cognates can also help to prune the search space: we can restrict the search space to be around them [Melamed, 1999; Lamraoui and Langlais, 2013]. For these methods, cognates play a critical role.

Correctly identifying cognates from bitexts is important for methods relying on them. Melamed [1999] defines cognates as word pairs in different languages that have the same

meaning and similar spellings. A special kind of cognates is verbatim copies across languages. Some bitexts contain mark-up segments which are copies. As already mentioned, such mark-ups are likely to occur for formatting or structural reasons. Other kinds of copies also exist: dates, named entities, numerals, punctuations, etc. However, generally verbatim copies account only for a very small portion in a bitext. Simard et al. [1993a] first propose to use cognates to align sentences. They consider two types of cognates :

- an exact match of numerals and punctuations form a pair of cognates;
- any pair of alphabetical tokens sharing a prefix of length four or more are considered as a pair of cognates.[2] Tokens with less than four letters are not taken into consideration, as they can easily introduce false positives.

This is of course an approximation, as it can, on the one hand, miss true cognates, such as "government" and "gouvernement"; on the other hand, this heuristic accepts wrong pairs, such as "computation" and "comprendre". Despite the roughness of the definition, it proves to be a highly effective criterion, and is adopted in some subsequent studies [Yu et al., 2012a; Lamraoui and Langlais, 2013]. There are other ways to identify cognates. Brew et al. [1996] compare six variants of Dice's coefficient. Melamed [1999] proposes a measure based on the Longest Common Subsequence Ratio (LCSR). Two tokens are deemed to be cognates if the ratio between the length of their longest common subsequence (not necessarily contiguous) and the length of the longer token exceeds a certain threshold. For language pairs that have totally different writing systems, it is possible to detect cognates according to phonetic and semantic similarity [Knight and Graehl, 1997]. Mitkov et al. [2007] give a comprehensive overview of cognate extraction methods.

Mono-pass vs Multi-pass Strategies

We have discussed the use of the multi-pass strategy to extract anchor points in § 2.2.1. In fact, we can obtain various other kinds of information during a first pass of analysis of the full bitext. Utsuro et al. [1994] extract word correspondences from the results of the first pass, which help to realign the bitext. Moore [2002] employs the first pass to find high confidence 1-to-1 links, then trains a simple lexical translation model with them. Melamed [1999] and Lamraoui and Langlais [2013] identify good candidate word pairs in the bitext space, then search for the best path visiting these pairs, and search around them for the sentence alignment.

The fundamental reason for the popularity of the multi-pass strategy is that we can gather endogenous corpus-level regularities through analyzing the corpus as a whole, and such information is extremely useful for sentence aligners. Early sentence alignment systems were expected to be light pre-processing tools, and are presumed to be working on

[2]Character identity is independent of capitalization and diacritics.

highly regular bitexts. Some simple heuristics such as sentence length ratio between languages worked well. However, on complex bitexts it is difficult to achieve good performance relying only on information as simple as pre-defined sentences length ratio. To successfully align difficult corpora, subsequent methods tend to collect corpus-specific knowledge. For example, word frequency distribution varies a lot from corpus to corpus. Moore [2002] demonstrates that a small translation table built on endogenous parallel sentence pairs is very effective to help improve the overall performance, avoiding to rely on large external lexicons.

A natural concern about the multi-pass strategy is its efficiency. Some early methods (e.g. [Gale and Church, 1991; Chen, 1993]) avoid the multi-pass solution due to computational limitations, which have become less and less severe.

2.2.2 Sentence Alignment Scoring

Our formulation of the sentence alignment problem in § 2.1.3 contains a generic scoring function $\texttt{SENT\_ALIGN\_SCORE}(\mathcal{A}, \mathbf{E}_1^M, \mathbf{F}_1^N)$. This function must be instantiated to be operational. Since sentence alignment is composed of alignment links, generally, we consider that an alignment with more correct links and fewer wrong ones is better. This is reflected in the now standard evaluation metric of sentence alignment: Precision (P), Recall (R) and F-score (F) [Véronis and Langlais, 2000]. Hence, most studies, if not all, decompose the generic alignment scoring function into the product of alignment link scores. We again use a generic sentence alignment link scoring function:

$$\texttt{SENT\_LINK\_SCORE}([\mathbf{E}_o^q; \mathbf{F}_r^t], \mathbf{E}_1^M, \mathbf{F}_1^N) \tag{2.2}$$

and the sentence alignment problem becomes:

$$\begin{aligned} \max_{\mathcal{A}=(l_1,..,l_k,..,l_K)} \quad & \prod_k \texttt{SENT\_LINK\_SCORE}(l_k, \mathbf{E}_1^M, \mathbf{F}_1^N) \\ \texttt{subject to} \quad & l_{k-1} \prec_l l_k \quad \forall k : 1 < k \leq K \end{aligned} \tag{2.3}$$

The goal of sentence alignment being to find the best set of alignment links, candidate link scoring is central to the final performance. A good scoring function should be able to distinguish between correct and wrong links. In the sentence alignment literature, many substantially different link scoring metrics exist. We will discuss them according to the types of knowledge that they rely on.

Length-based Metrics

For some language pairs, sentence lengths exhibit a certain level of translation regularity. Gale and Church [1991] observe that:

"longer sentences in one language tend to be translated into longer sentences in the other language, and that shorter sentences tend to be translated into shorter sentences."

They conjecture that true corresponding sentence pairs usually have correlated lengths. Gale and Church [1991] conduct experiments on economical reports of the Union Bank of Switzerland (UBS), showing that the correlation between character lengths of the English version and the German version is 0.991. Gale and Church [1991] claim the correlation is strong enough to be the basis of a probabilistic model. Thus they develop a sentence alignment system which scores alignment links based on character lengths of its two sides.

The probabilistic model captures the regularity of *length ratios*. For a candidate link $[\mathbf{E}_o^q; \mathbf{F}_r^t]$ with respective length ℓ_E and ℓ_F (measured in characters), the system assigns a probability according to a generative probabilistic model relying only on the lengths. Based on the assumption that a source character is responsible for a random number of target characters, they assume that ℓ_F verifies a normal distribution. The particular values of the mean and the variance of the normal distribution depend on ℓ_E and ℓ_F. Gale and Church [1991] propose to use the following parameters:

- the expectation of ℓ_F is given by $c \cdot \ell_E$, where c is a constant.
- the variance is proportional to $\frac{\ell_E+\ell_F}{2}$, where the slope is a constant denoted by s^2.

The parameter c models the expectation of the ratio between ℓ_F and ℓ_e. The expression of the variance captures the fact that the ratio between longer pairs tends to have a larger variation. In [Gale and Church, 1991], the parameter c is set by taking the average ratio of lengths between parallel paragraphs, and s^2 is obtained from a regression analysis. They find the performance of the system is not very sensitive to the precise values of these parameters, thus use the values: $c = 1, s^2 = 6.8$. They further define

$$\delta = \frac{\ell_F - c \cdot \ell_E}{\sqrt{\frac{\ell_E+\ell_F}{2} s^2}} \tag{2.4}$$

which is a function of $\mathbf{E}_o^q$ and $\mathbf{F}_r^t$, and arrive at a standard normal distribution

$$\delta \sim \mathcal{N}(0, 1)$$

Now Equation (2.2) is instantiated to:

$$\begin{aligned} \texttt{SENT\_LINK\_SCORE}([\mathbf{E}_o^q; \mathbf{F}_r^t], \mathbf{E}_1^M, \mathbf{F}_1^N) &:= P_{GC}([\mathbf{E}_o^q; \mathbf{F}_r^t] | \mathbf{E}_1^M, \mathbf{F}_1^N) \\ &:= P_{GC}([\mathbf{E}_o^q; \mathbf{F}_r^t] | \mathbf{E}_o^q, \mathbf{F}_r^t) \\ &\propto P_{GC}(\mathbf{E}_o^q, \mathbf{F}_r^t | [\mathbf{E}_o^q; \mathbf{F}_r^t]) P_{GC}([\mathbf{E}_o^q; \mathbf{F}_r^t]) \end{aligned} \tag{2.5}$$

At this stage, Gale and Church [1991] make two supplementary assumptions:

- $P_{GC}(\mathbf{E}_o^q, \mathbf{F}_r^t | [\mathbf{E}_o^q; \mathbf{F}_r^t])$ is determined by $P_{GC}(\delta(\mathbf{E}_o^q, \mathbf{F}_r^t))$. Its value can be obtained from the standard normal distribution.

– $P_{GC}([\mathbf{E}_o^q; \mathbf{F}_r^t])$ is determined by the link type, that is, the quantities $q - o + 1$ and $t - r + 1$.

Gale and Church [1991] hand-code a prior probability table over link types. They assign non-zero probabilities to five types: 1:0, 0:1, 1:1, 1:2, 2:1. All other link types are assigned zero prior probability. This design choice is motivated by analyses of the corpus, and leads to significant complexity reductions in the decoding.

We note that Gale and Church [1991] characterize the link scoring model with only one feature based solely on the lengths of involved sentences. One can easily generalize this model by using multiple features:

$$\texttt{SENT\_LINK\_SCORE}([\mathbf{E}_o^q; \mathbf{F}_r^t], \mathbf{E}_1^M, \mathbf{F}_1^N) \propto P(\mathbf{f}(\mathbf{E}_o^q, \mathbf{F}_r^t) | [\mathbf{E}_o^q; \mathbf{F}_r^t]) P([\mathbf{E}_o^q; \mathbf{F}_r^t]) \tag{2.6}$$

where $\mathbf{f}(\mathbf{E}_o^q, \mathbf{F}_r^t)$ is a vector of features. Many methods differ mainly in the way they instantiate Equation (2.6).

The link scoring method of [Brown et al., 1991] is based on a similar but more direct model of the length ratio. They also restrict the link types under consideration, by only assigning non zero prior probability to types 1:0, 0:1, 1:1, 1:2, 2:1. The probability of a null link $[\mathbf{E}_o^o;]$ with source length ℓ_E is represented by :

$$P_B([\mathbf{E}_o^o;]) \approx P_B(1{:}0) P_B(\ell_E | [\mathbf{E}_o^o;])$$

where the first term of the right hand side only depends on the link type and is pre-defined, the second term is approximated by the empirical frequency :

$$P_B(\ell_E | [\mathbf{E}_o^o;]) = \frac{\texttt{count}(\ell_E)}{\sum_\ell \texttt{count}(\ell)}$$

where $\texttt{count}(\cdot)$ is a function mapping a length to its number of occurrences in the bitext. We have similar models for $[; F_r^r]$. The probability of a non-null link $[E_o^q; F_r^t]$ with source side length ℓ_E and target side length ℓ_F is given by:

$$P_B([\mathbf{E}_o^q; F_r^t]) \approx P_B(q - o + 1{:}t - r + 1) P_B(\ell_E | q - o + 1{:}t - r + 1) P_B(\ell_F | \ell_E)$$

where the second term of the right hand side is computed as the empirical frequency of ℓ_E (in the 2:1 case, each of the two source lengths is drawn from the distribution $P(\ell_E|1{:}0)$). The third term is modeled by a normal distribution. If we note $r = \log(\ell_F/\ell_E)$, then

$$P_B(\ell_F | \ell_E) = \alpha \exp\left\{ -\frac{(r - \mu)^2}{2\sigma^2} \right\}$$

with α a normalizing constant. Model parameters are estimated on a small development bitext using the EM algorithm.

These two length-based methods are very simple, still, the reported results are surprisingly good. Gale and Church [1991] claim the model score of a link is a reasonable predictor of its correctness. These ideas have been borrowed in many subsequent works. For instance, Moore [2002] uses the method in [Brown et al., 1991] as a first pass to identify a sample of high confidence 1:1 pairs.

Liu et al. [2014] introduce concepts from information theory to measure the distance between source and target sentences. Given an alignment link, they use the Prediction by Partial Matching (PPM) scheme to compress sentences, encoding each byte based on its occurrence history. The length (in bytes) of the compression of a sentence reflects its cross entropy. The authors claim that parallel sentences should have similar amount of information. Thus, they use the ratio of compression code lengths and absolute code length differences as the link scoring metric. This is another way to measure links without any reference to lexical content, thus very quick and language independent. This method is quite innovative.

Length-based link scoring methods have some intrinsic drawbacks. First, given that the only information they rely on is the correlation of lengths, the metrics can easily be tricked by a pair of non-parallel sentences with a good length correlation totally by chance. Second, these methods tend to be language dependent. Wu [1994] remarked that for languages pairs not belonging to the same language family, e.g., English-Chinese, the sentence length correlation is much weaker and the δ in Equation (2.4) does not follow any smooth distribution. Gale and Church [1991] admit that to handle difficult situations (e.g. difficult passages or non-related language pair), lexical constraints are to be incorporated into the model. So, in the following we discuss such lexical-based scoring models.

Lexical-based Metrics

The ultimate goal of alignment link scoring is to identify sentence pairs that express similar meaning. Given that the meaning of sentence is mostly the results of composing the meaning of individual word tokens, a natural idea for link scoring is to compare whether the two sides contain some related words. Many methods which consider different aspects of words of the source and target sentences have been proposed and tested.

Cognates Cognates, as discussed above, are one type of lexical clues. Simard et al. [1993a] propose a sentence pair similarity metric which relies purely on cognates. They call this measure the *cognateness*. Their definition of cognates has been introduced in § 2.2.1. Given a candidate alignment link $[\mathbf{E}_o^q; \mathbf{F}_r^t]$ with source side length (number of tokens) ℓ_E and target side length (again number of tokens) ℓ_F, they first count the number of pairs of cognates C, in such a way that the largest possible number of pairs of cognates can be obtained and no token is used twice (this is the bi-partite graph maximum matching

problem). Given C, the "cognateness" of the link is:

$$\gamma = \frac{2C}{\ell_E + \ell_F}$$

Simard et al. [1993a] estimated the empirical mean of γ_t for true sentence alignment links and the mean of γ_r for random links. They then use a binomial distribution to model the probability that C cognate pairs occur in a link with average length N, with parameters $p_t = E[\gamma_t]$ and $p_r = E[\gamma_r]$:

$$P_S(C|N,t) = \left(\begin{array}{c} N \\ C \end{array} \right) \cdot (p_t)^C \cdot (1-p_t)^{(N-C)}$$

$$P_S(C|N,r) = \left(\begin{array}{c} N \\ C \end{array} \right) \cdot (p_r)^C \cdot (1-p_r)^{(N-C)}$$

They find that, although the assumption that the translation relation and the cognateness level are correlated is valid, this correlation is a weaker measure of parallelism than the length ratio of [Gale and Church, 1991]. Cognate-based metrics are intuitive and efficient, but they are by nature language dependent. Also, even though Simard et al. [1993a] show that cognate-based measures tend to make different mistakes than length-based ones, they do not help to identify null links, which is one of the most severe problems of the method of [Gale and Church, 1991]. In fact, the measure of [Simard et al., 1993a] would always assign negligible probability mass to any null link. Thus they remain a problem.

POS tags Part-of-speech (POS) tags are another type of lexical information. Papageorgiou et al. [1994] use POS tags on the two sides of an alignment link to define a link scoring measure. They argue that the similarity between the two sides of a link must be reflected by some semantic correspondence, which, according to Basili et al. [1992], can be measured by the patterns of POS tags of its content words (that is, verbs, nouns, adjectives, and adverbs). They define four tag patterns, each containing several tags. They then obtain the cardinality of each pattern, and use linear regression to build a model relating pattern cardinalities to the total count of tags on the target side. The regression model contains a normal noise term. They estimate the parameters of the normal distribution from manually aligned data, and use the distribution to assign a probability score to candidate links. The evaluation on a highly institutionalized bitext shows that the method achieves accuracy around 100%. In particular, it finds 4 null links out of 5, and 35 of the 36 links of types 1:2 and 2:1. However, given the modest size of the evaluation corpus and the lack of comparison with other methods on the same test set, it is difficult to assert the effectiveness of the measure of [Papageorgiou et al., 1994]. Chen and Chen [1994] also use POS criteria to measure the parallelism degree of bitext fragments. The idea is that, for a good alignment link, the numbers of "important" POS tags of the two sides should be close. They count the number of nouns, verbs, adjectives, as well as numbers and quotation marks in each sentence, and

use it as the "weights". A candidate link has an "energy" value, which is the difference of the sums of weights of its two sides. A full sentence alignment is measured by the total energy of its member links. Kutuzov [2013] also proposes to improve English-Russian sentence alignment by POS-tagging, based on the hypothesis that corresponding source and target sentences should also correspond in the number and order of content POS tags.[3] Kutuzov [2013] tags each sentence with four labels: nouns, adjective, verbs and pronouns. For each candidate alignment link, Kutuzov [2013] computes the Damerau-Leveshtein distance (normalized by length of the target sentence) between the two tag sequences. A threshold on this distance is applied to filter out bad pairs.

Word translation tables Mainstream lexical-based link scoring methods are based on word translation distributions. The intuition behind these methods is that the two sides of a good link should contain pairs of corresponding words, and corresponding words tend to have similar distributions in their respective text. By "similar distribution", we mean that the pair of words have similar numbers of occurrences, and the positions of such occurrences are related. Many studies propose different ways to model word distributions (see [Kay and Röscheisen, 1993; Chen, 1993; Fung and Church, 1994], etc).

When external resources are available, they can greatly help the alignment system. The most obvious word correspondences are dictionary entries. Utsuro et al. [1994] define the score of a link as the ratio between the number of dictionary matches and the number of content words in the link, and use this metric to obtain a first alignment. Then new word correspondences are extracted from the alignment and are added into the dictionary. The bitext is realigned with the updated dictionary. This method requires external dictionaries. If no dictionary is available, but instead large-scale parallel corpora exist (which is the case nowadays), one can also estimate the word similarity distribution using existing parallel corpora. Simard and Plamondon [1998] pre-train an IBM Model 1 [Brown et al., 1993] from a large parallel corpus, and use it to score each candidate link.

When no external resource is available, one can use a multi-pass strategy to extract word correspondence information from the bitext under study. Kay and Röscheisen [1993] propose one of the first lexical-based sentence alignment methods that do not rely on external dictionaries. They first establish an *alignable sentence table (AST)* inside the bitext. Each entry of the AST is a source-target sentence pair, which can be viewed as a candidate link. They then estimate a simple *word alignment table (WAT)* from the AST. Every pair of words belonging to one candidate link in the AST is assigned a score indicating the level of their similarity. Kay and Röscheisen [1993] use Dice's coefficient as the similarity metric. Word pairs in the WAT are ranked according to their similarity value and frequencies. Then the WAT is used to establish a partial sentence alignment. They scan the WAT in order, for each word pair, in order to find all the sentence pairs in the AST

[3]Note the matching of ordering of POS tags is quite a strong assumption. It is perhaps reasonable for English-Russian, but might be wrong for some other language pairs.

that contain the word pair. These sentence pairs can enter into the final *sentence alignment table (SAT)* if they do not cross any member of the SAT. A new AST is then constructed, based on the current SAT, and the whole procedure is repeated until convergence is reached. Thus, the method of [Kay and Röscheisen, 1993] is a typical application of a multi-pass strategy. It essentially evaluates a link by the largest similarity value of its word pairs according to the word distribution. The method of Haruno and Yamazaki [1996] is quite similar to [Kay and Röscheisen, 1993] in terms of system design. To score a link, they use a dictionary and a derived word correspondence table (similar to the WAT) to obtain the number of dictionary matches, mutual information, and t-score (the latter two are first investigated in [Fung and Church, 1994] to study word distribution) of all possible word pairs. With these three sets of quantities, they use thresholds to judge whether a candidate link should enter the table of links.

More statistically principled word translation models have also been investigated. Chen [1993] proposes a generative sentence alignment model. For a bitext $[\mathbf{E}_1^M; \mathbf{F}_1^N]$, the probability of a valid alignment $\mathcal{A}$ is the product of the probabilities of individual links.

$$P(\mathcal{A}, \mathbf{E}_1^M, \mathbf{F}_1^N) = p(|\mathcal{A}|) \prod_{k=1}^{|\mathcal{A}|} p(l_k)$$

where $|\mathcal{A}|$ is the number of links in the alignment, l_k is the k-th link. The link probability is estimated according to a word-to-word translation distribution, which is constructed on the fly during the alignment process. A link is assumed to be generated by a list of *word beads*. A word bead can be one source word with one target word (1:1 bead), or one source word alone (1:0 bead), or one target word alone (0:1 bead). The actual generative model is relatively complicated. In general, the probability of one link is the sum of the probabilities of all possible word bead lists that are consistent with the link. The probability of a word bead list is in turn estimated by a normalized product of the probability of the word beads. The distribution of word beads is bootstrapped on a small aligned corpus, then reinforced during the alignment. To maintain the one-pass property of the overall procedure, model parameters are estimated by an on-line Viterbi approximation to EM. This model leads to a bilingual lexicon as a by-product.

In general, these pure lexical-based link scoring methods lead to alignments with good precision, at a considerable computation cost. Adding length information into lexical methods hardly hurts its efficiency. Thus, many works use a combination of the two.

Multifeature Metrics

Most sentence alignment methods combine several sources of information to evaluate a candidate link, including lengths and lexical clues. Following Wu [2010], we call them *multifeature* methods. We try to unify our discussions using probabilistic frameworks, even though in some works the formulation is not explicit.

Generative models Many methods can be understood as different ways of instantiating Equation (2.6). Thus they are all included in the framework of generative probabilistic models.

Some methods rely on external resources. Wu [1994] also adopts the normal length feature of [Gale and Church, 1991]. But, observing that English-Chinese sentence pair length ratio does not follow any smooth normal distribution, Wu [1994] adds other parameters to model the number of times that a source word e and a target word f occur in the candidate link, where the pair (e, f) is in a pre-defined list of corresponding word pairs, called a *lexical cue*. The whole model is thus enriched by lexical dependency information. Davis et al. [1995] follow the same intuitions, by combining length feature and dictionary-based word translation features, and further adding cognate and other string matching features into the model.

Some methods induce all required knowledge from the bitext to align. Moore [2002] runs a first pass on the bitext using only length information. A modified version of IBM Model 1 is then trained on high-confidence 1:1 links. Importantly this modified Model 1 can take null links into account. Finally Moore [2002] re-aligns the bitext using both the word translation table and the length distribution. Braune and Fraser [2010] slightly alter the metric of Moore [2002]: the link type prior is estimated on the result of a first pass, and word translation probabilities are computed by a single Viterbi alignment instead of the sum of all alignments. Braune and Fraser [2010] report minor empirical performance gain with these changes. Ma [2006]; Li et al. [2010a] use the *tf-idf* weight, which is widely used in Information Retrieval (IR). Each source sentence segment and target sentence segment are viewed as documents, and the whole bitext is viewed as the corpus. In this way, the alignment link scoring is equivalent to the document relevancy problem in IR. Ma [2006] computes the link score by a combination of tf-idf weights, penalties on link types, and penalties on segment lengths. The authors claim that tf-idf weight increases the robustness of the method.

The methods of Varga et al. [2005]; Sennrich and Volk [2010] are resource-flexible. [Varga et al., 2005] propose, when a dictionary is available, to generate a word-to-word literal translation for every source sentence. The pseudo target language text is then compared against the real target text, sentence by sentence, using a similarity measure which combines the shared token ratio and the length ratio. In [Sennrich and Volk, 2010], every source sentence gets translated by a machine translation (MT) system. Then every pseudo translation is compared with every real target sentence with a modified version of the BLEU metric [Papineni et al., 2002]. These methods can work even without external resources. For [Varga et al., 2005], the first step relies on cognates, and a small lexicon is constructed using the result of the first step, which can be used in downstream steps. In absence of MT systems, Sennrich and Volk [2010] resolve to compare the source sentences directly with target ones using BLEU. However, in general both systems obtain much better performance when resources are available.

Discriminative models Yu et al. [2012a]; Kaufmann [2012] propose to use a Maximum Entropy (MaxEnt) model as the scoring tool. The link scoring model is:

$$P_{ME}([\mathbf{E}_o^q; \mathbf{F}_r^t]|\mathbf{E}_1^M, \mathbf{F}_1^N) = \frac{1}{1 + \exp\{-\lambda^\top \mathbf{f}([\mathbf{E}_o^q; \mathbf{F}_r^t], \mathbf{E}_1^M, \mathbf{F}_1^N)\}} \tag{2.7}$$

where $\mathbf{f}([\mathbf{E}_o^q; \mathbf{F}_r^t], \mathbf{E}_1^M, \mathbf{F}_1^N)$ is the feature vector and λ is the parameter vector. An important advantage of the MaxEnt model is that it estimates a posterior probability of alignment links, which provides a principled framework for subsequent applications, such as confidence estimation. The first step of Yu et al. [2012a] runs the method of Moore [2002], because of its capacity of identifying high-precision 1 : 1 alignment links. The parameters λ are trained using these links, with features encoding length ratio, cognate number, token match information, etc. Kaufmann [2012] learns the MaxEnt model parameters on an external parallel corpus. A problem for MaxEnt models is that they must be trained on both positive and negative examples. For both [Kaufmann, 2012] and [Yu et al., 2012a], there are no negative examples in the bitext. To solve this problem, Kaufmann [2012] pairs source sentences with random target ones to generate negative examples. Yu et al. [2012a] intentionally pair a source sentence with a wrong target one which is close to the correct, in order to make the negative examples less obvious (e.g. the negative examples produced in this way have larger chance to share some tokens with the positives), and hopefully make the model more robust.

Mújdricza-Maydt et al. [2013] use linear chain Conditional Random Fields (CRFs) to model sentence alignment. One variable $a_{m,n}$ is created for each source-target sentence pair E_m and F_n. For every diagonal of the bitext space, a linear-chain CRF is constructed, e.g. ($a_{1,1}$, $a_{2,2}$, $a_{3,3}$, ...) and ($a_{1,2}$, $a_{2,3}$, $a_{3,4}$, ...). Compared to MaxEnt, in CRFs the prediction of one link can make use of information of neighboring links. For instance, the formulation of Mújdricza-Maydt et al. [2013] captures the important Markov property in bitext sentence alignment: if E_m is aligned to F_n, then in most bitexts it is very likely that E_{m+1} aligns to F_{n+1}. Mújdricza-Maydt et al. [2013] use heuristics to resolve prediction conflicts among variables.

Alignment scoring is central in any sentence alignment system, thus it has been extensively studied. The main differences between many methods lie on alignment scoring. Yet most methods, if not all, have measured alignment links independently of any other link, using only surface cues extracted from the sentences involved in the link. This is a strong simplifying assumption. With this assumption it is very difficult to handle null links in any principled way, in particular for measures that include lexical information. Null links remain an important issue for sentence alignment.

2.2.3 Decoding

The other major problem of sentence alignment is the decoding, i.e. the search for the best alignment under the scoring model. Although the conventions outlined in § 2.1.1 greatly

reduce the search space, it is still typically too large to be exhaustively searched. For convenience, we repeat our definition of sentence alignment:

$$\begin{aligned} \max_{\mathcal{A}=(l_1,..,l_k,..,l_K)} \quad & \texttt{SENT\_ALIGN\_SCORE}(\mathcal{A}, \mathbf{E}_1^M, \mathbf{F}_1^N) \\ \texttt{subject to} \quad & l_{k-1} \prec_l l_k \quad \forall k : 1 < k \leq K \end{aligned}$$

Without further assumptions, the computation of this problem is exponential in M and N. Consider the problem of segmenting $\mathbf{E}_1^M$ into non-empty sentence segments (with at least one sentence). There are $O(2^M)$ ways to perform this single-sequence segmentation. Since our definition of sentence alignment allows null alignment links, the number of valid alignments is at least as large as single-sequence segmentations. The actual computational complexity of the search problem depends on the scoring function. If $\texttt{SENT\_ALIGN\_SCORE}(\mathcal{A}, \mathbf{E}_1^M, \mathbf{F}_1^N)$ decomposes as a product of individual link scores, then we can employ dynamic programming, and the computation complexity becomes $O(M^2N^2)$, which is still too high for large-scale bitexts.

Fortunately, bitexts exhibit some properties that might warrant reasonable assumptions for search space reduction. For instance, it is generally unlikely (if not impossible) to translate many sentences together into one. Large deletion/insertion blocks are also rare. Such observations motivate several search space reduction heuristics, which we first discuss. We then present the two main categories of search algorithms: greedy search and Dynamic Programming (DP).

Search Space Reduction

The geometrical point of view of bitexts, i.e. the bitext space illustrated in § 2.1.2, can facilitate the understanding of search space reduction. From this point of view, a valid full-text alignment is a consecutive chain of non-overlapping small rectangles, which begins at the origin (the bottom left corner of the space), growing up and to the right, and ends at the terminal (the top right corner). Clearly, we can draw many such chains of rectangles. The main search space reduction method consists of cutting off regions of bitext space, by restricting the search inside a neighborhood around a pre-defined path.

Bitext diagonal An intuition for search space reduction is that promising sentence alignment rectangle chains tend to lie around the bitext diagonal. In Figure 2.8, the rectangle chain is relatively far away from the bitext diagonal. This is a valid alignment according to our definition, but it is very unlikely a good one, because the rectangles suggest a highly irregular translation pattern: a few beginning source sentences are translated into the large majority of target ones, while most source sentences are contracted into a few ending target ones. Thus, among the many valid chains, we can tell some (like the one in Figure 2.8) are not very probable, even without a rigorous examination of the sentences.

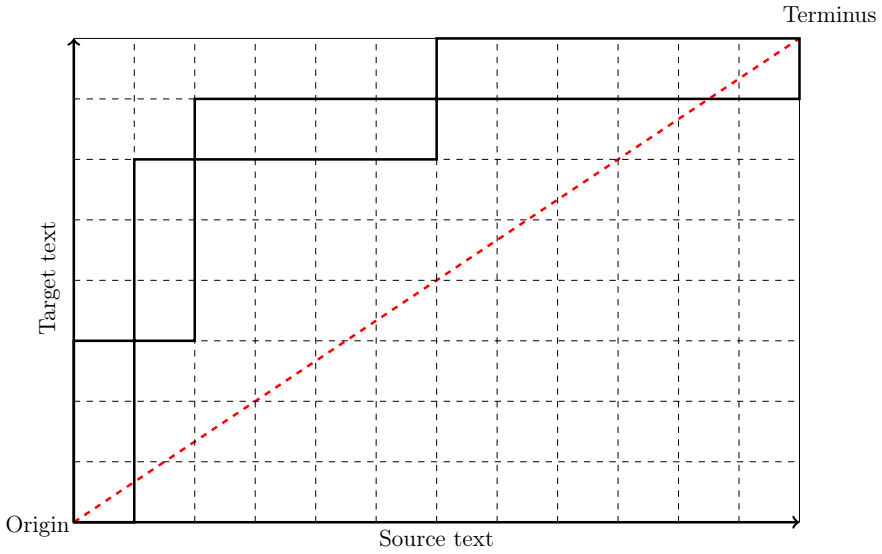


Figure 2.8: A very unlikely sentence alignment.

Some methods restrict the search space around the bitext diagonal. Melamed [1999] uses a fixed-width neighbourhood, when no cognate information is available. Moore [2002] uses a variable-width neighborhood, as displayed in Figure 2.9. If one run of the search does not find any sensible link sequence, the neighborhood width would be expanded, and the search restarts, until an acceptable chain is found.

Preliminary alignments A more elaborate way of defining paths is through preliminary alignments. One choice is to establish an initial sentence alignment. Braune and Fraser [2010] use a length-based model to compute a first alignment path, and restrict the search in a neighborhood around this path. The other, more frequently applied choice is to find sub-sentential alignments before performing sentence-level alignment. In a bitext space, some points (word-level units) can be deemed to be in correspondence, e.g. cognates. Melamed [1999] calls them Likely Points of Correspondence (LPCs). Such LPCs are filtered to obtain a chain of True Points of Correspondence (TPCs), using several heuristics:

– LPCs whose corresponding words are too frequent are filtered out;

– TPCs do not share horizontal or vertical coordinates;

– TPCs should line up straight;

– the slope of the TPC chain should be close to that of the bitext diagonal;

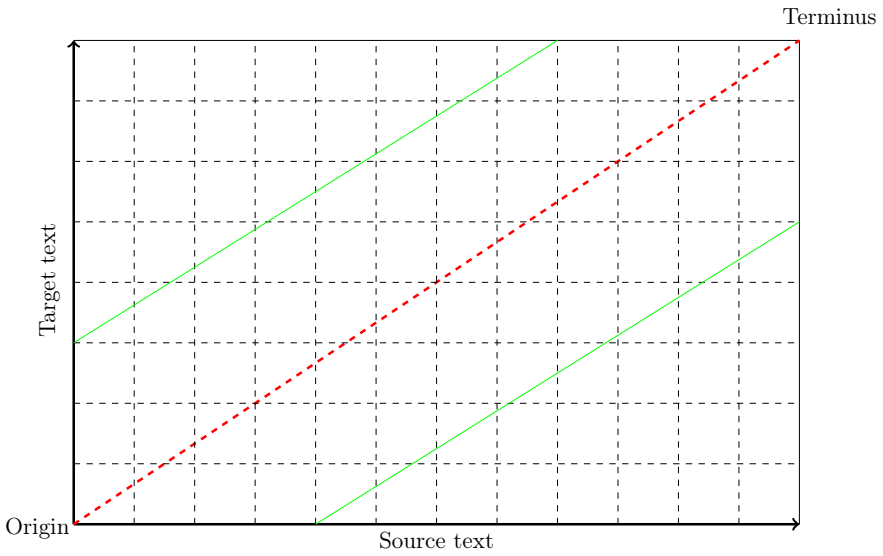


Figure 2.9: The search space reduction method in [Moore, 2002]. The search is only performed inside the region bounded by the two green lines.

and some post-adjustments. Langlais [1998]; Lamraoui and Langlais [2013] identify a set of cognate correspondence points in the bitext space, and compute a cognate-based alignment path using DP. Figure 2.10, taken from [Lamraoui and Langlais, 2013], illustrates this process.

Restriction of link types Another commonly applied heuristic is the restriction of link types. The third convention of sentence alignment implies to identify alignment links as precisely as possible. This implies that, many-to-many link types seldom exist in gold bitext alignments. For some bitext genres, most correct links are of type 1:1. Some genres, such as literary bitexts, contain more type variations. But in general, m:n types with big m and n, say, both larger than 5, are quite unlikely. Deletions or insertions can be modeled by sequences of 1:0 or 0:1 links. Thus, sentence alignment methods do not explore all possible link types. Often we consider m:n with m and n both smaller than 3. In some papers, such as [Braune and Fraser, 2010], 4:1 and 1:4 types are also considered.

The restriction on links types greatly reduces the number of possible links to explore. For DP-based search, this restriction reduces the complexity from $O(M^2N^2)$ to $O(MN)$ (recall M is the number of source sentences and N is the number of target sentences). In fact, it is perhaps more accurate to discuss this restriction in alignment scoring, since it amounts to assign zero score to "large" links. The discussion here, however, emphasizes the reduction of the complexity of the search problem.

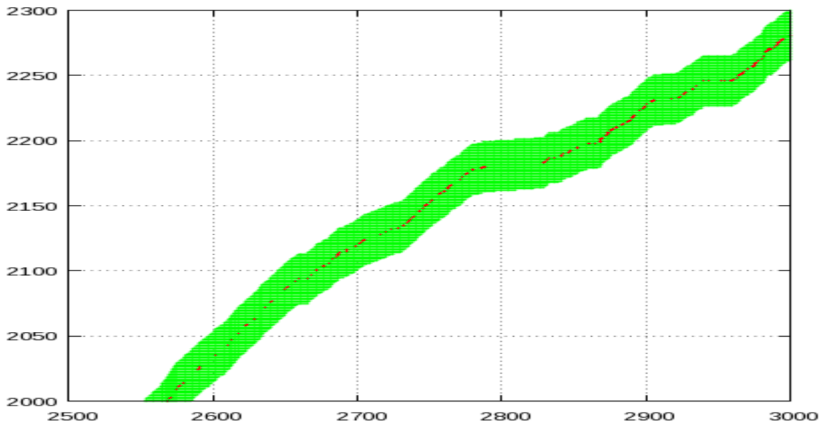


Figure 2.10: The search space reduction method in [Lamraoui and Langlais, 2013]. The dashed path is the cognate-based alignment. The search for sentence alignment is performed within the green region around the cognate alignment path.

Greedy search

For sentence alignment methods, greedy search implies to pick the best link according to the scoring function, include it in the final alignment, delete all competing candidate links, and repeat this process. This method is simple to implement. Kay and Röscheisen [1993]; Melamed [1999] are two representative systems using greedy search. With link type restrictions, the worst-case complexity of greedy search for a bitext $[\mathbf{E}_1^M; \mathbf{F}_1^N]$ is $O(MN)$.

Recent methods tend to favor dynamic programming over greedy search. In theory, DP is exact search under the scoring function, while greedy search is approximative. Compared to the DP search, which aims to find the best overall link list, greedy search focuses on picking locally best links according to the model. However, the MaxEnt-based system of Yu et al. [2012a] uses greedy search instead of DP, probably due to the difficulty of modeling null links under such model.

Dynamic programming

Most systems use Equation (2.3) to replace the generic sentence alignment problem in Equation (2.1), which warrants the use of DP to find the best link sequence. Gale and Church [1991] give perhaps the most classic form of the DP for sentence alignment. With link types restricted to 1:0, 0:1, 1:1, 2:1, 1:2, Gale and Church [1991] solve the search

problem by:

$$D(m,n) = \min \begin{cases} D(m,n-1) & - & \log \texttt{SENT\_LINK\_SCORE}([;\texttt{F}_\texttt{n}], \mathbf{E}_1^\texttt{M}, \mathbf{F}_1^\texttt{N}) \\ D(m-1,n) & - & \log \texttt{SENT\_LINK\_SCORE}([\texttt{E}_\texttt{m};], \mathbf{E}_1^\texttt{M}, \mathbf{F}_1^\texttt{N}) \\ D(m-1,n-1) & - & \log \texttt{SENT\_LINK\_SCORE}([\texttt{E}_\texttt{m};\texttt{F}_\texttt{n}], \mathbf{E}_1^\texttt{M}, \mathbf{F}_1^\texttt{N}) \\ D(m-2,n-1) & - & \log \texttt{SENT\_LINK\_SCORE}([\mathbf{E}_{\texttt{m}-1}^\texttt{m};\texttt{F}_\texttt{n}], \mathbf{E}_1^\texttt{M}, \mathbf{F}_1^\texttt{N}) \\ D(m-1,n-2) & - & \log \texttt{SENT\_LINK\_SCORE}([\texttt{E}_\texttt{m};\mathbf{F}_{\texttt{n}-1}^\texttt{n}], \mathbf{E}_1^\texttt{M}, \mathbf{F}_1^\texttt{N}) \end{cases}$$

where $D(m,n)$ is the partial cumulative score at the position (m,n), that is, the m-th source and the n-th target sentence. We are assuming here the scoring scheme always gives positive scores. One can thus recursively compute the minimal score at the terminal position (M,N), and obtain the best link sequence via backtrace. The complexity of this recursion is $O(MN)$, where the constant factor depends on the number of considered link types. It is easy to generalize this computation to consider other link types.

Classical dynamic programming based search methods often cannot handle large blocks of deletions or insertions well [Gale and Church, 1991]. In these cases, one can modify the DP to enable it to handle large deletions (insertions) specially, as is done in [Chen, 1993]. One can also resort to deletion detecting methods (e.g. [Melamed, 1996]) to find these blocks before performing sentence alignment.

A final remark on the decoding: we have presented greedy search and DP with the assumption of independent links, because the vast majority of methods are included in this family. For the sequence CRFs model of Mújdricza-Maydt et al. [2013], classical forward-backward and Viterbi algorithms are used.

2.3 General Comments

In this chapter, we give a general overview of state-of-the-art sentence alignment methods. This problem is often considered to be solved. From the point of view of MT, this claim seems reasonable. However, as Tiedemann [2011] points out, the utility of sentence alignment is not limited to the collection of parallel sentence pairs. It can also be used in cross-lingual information retrieval, translation study, multi-lingual reading, etc. In some cases, the performance of state-of-the-art systems is not satisfying, and the room for improvement is still large.

Our review of the literature reveals several interesting points on this topic. First, many state-of-the-art systems score a link using information gathered from itself only. In our opinion, while this way certainly leads to computational gains, it will be interesting to investigate the effect of adding contextual information to help link scoring. Second, when the problem was first studied in the early 1990s, both computational and linguistic resources were very limited. This is why many assumptions were made and simple models were preferred. However, such limitations can be no longer necessary now. We have nowadays many kinds of external resources at our disposal, which warrant the investigation of more complex and/or resource-demanding models.

Quan et al. [2013]; Zamani et al. [2016] propose to drop the flat contiguous segment and the monotonicity conventions. In [Quan et al., 2013], a sentence is allowed to be aligned to at most one sentence of the other side. The entire bitext sentence alignment is formulated as a scoring matrix G. An entry G_{ij} contains an alignment score between a source sentence E_i and a target F_j. An optimization process determines values of the entries:

$$G^* = \arg\min_G \mathcal{Q}_m(G) + \lambda \mathcal{Q}_b(G)$$

where $\mathcal{Q}_m(G)$ models monolingual constraints and $\mathcal{Q}_b(G)$ models bilingual constraints. The basic idea behind $\mathcal{Q}_m(G)$ is that, if a source sentence E_i closely relates to another source sentence E_j, and a target F_k closely relates to another target F_l, then the alignment score between E_i and F_k should be close to that between E_j and F_l. $\mathcal{Q}_b(G)$ ensures that G stays close to an initial, relatively coarse alignment matrix A (e.g. computed using simple word matching). Quan et al. [2013] compute the relatedness between two sentences of the same language using TF-IDF score vectors. This work explicitly models non-monotonic cases, and incorporates monolingual information into the alignment scoring method. Hence, it is a very innovative approach. In [Zamani et al., 2016], Integer Programming (IP) is used to formulate the sentence alignment problem, with one variable a_{ij} for each source-target pair E_i and F_j. The formulations in both works are more flexible on alignment links, and can be applied to sentence pair extraction from non-parallel documents. However, the decoding complexity is much higher than conventional algorithms where the monotonicity and the contiguous segment constraints are assumed. The design of reliable and efficient decoding methods for such kinds of formulations is also an interesting new track for sentence alignment research.

This chapter is an overview of sentence alignment methods. In Chapter 4, we will talk about issues concerning sentence alignment evaluation, for which we have created two hand aligned corpora, presented in Chapter 5. In Chapter 6, we discuss our contributions to sentence alignment methods. Finally, in Chapter 7 we investigate confidence measures for sentence alignment links.

Chapter 3

Word Alignment

Sub-sentential alignment amounts to extracting finer-grained correspondences between the two sides of a parallel sentence, typically generated by sentence alignment. Compared to sentence-level correspondences, sub-sentential alignments, especially word alignments, are more directly adapted to the statistical Natural Language Processing (NLP) paradigm: even in very large scale corpora, sentences (except very short ones) are rarely repeated. It is very hard, if not impossible, to build a probabilistic model based on statistics at the sentence level. On the contrary, sub-sentential structures such as words exhibit interesting statistical properties, making probabilistic modelings possible. Such sub-sentential correspondences are extremely useful in many NLP applications, the most prominent one being Statistical Machine Translation (SMT), as well as in cross-lingual transfer learning [Täckström et al., 2013; Aufrant et al., 2016], bilingual reading [Yvon et al., 2016], etc. Thus, a lot of research effort has been devoted to sub-sentential alignment.

In this chapter, we briefly review word alignment, which constitutes an important part of our study on confidence estimation. Other types of sub-sentential alignments also exist, e.g. word-to-phrase alignment [Deng and Byrne, 2008], phrase alignment [Marcu and Wong, 2002; DeNero, 2012], etc. [Wu, 2010; Tiedemann, 2011] give broader discussions on sub-sentential alignments.

In the pioneering work of Brown et al. [1993] on SMT, word alignment is defined in the context of parallel sentence processing.[1] Throughout this chapter, we illustrate our discussion on a pair of parallel sentences $\mathbf{e} = (e_1, ..., e_i, ..., e_I)$ and $\mathbf{f} = (f_1, ..., f_j, ..., f_J)$, where each e_i (respectively f_j) stands for a word in the language $\mathfrak{E}$ (respectively $\mathfrak{F}$). We use the notations $[I] = \{i : i \in \mathbb{N}, 1 \leq i \leq I\}$, and $[I]_0 = \{i : i \in \mathbb{N}, 0 \leq i \leq I\}$. A *word alignment* from $\mathbf{e}_I$ to $\mathbf{f}_J$ is a function:

$$a : [I] \rightarrow [J]_0$$

[1] In this chapter we will only discuss word alignment of a bilingual corpus, but the task can be extended to multi-lingual corpora.

which maps every word index in **e** to one index in **f** or a special index 0. Figure 3.1 illustrates the word alignment from an English sentence **e** to a French one **f**. We have

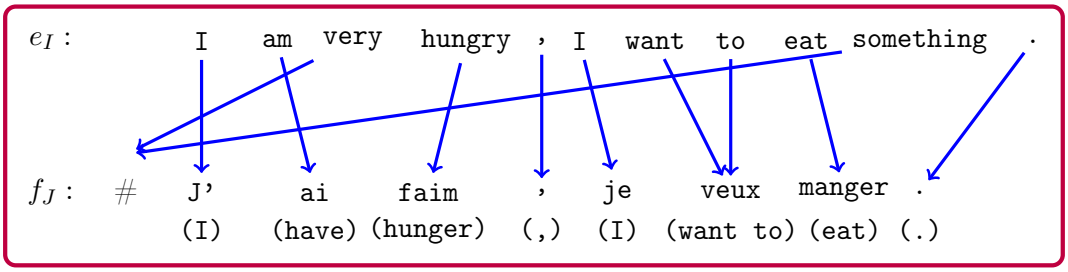


Figure 3.1: The word alignment from an English sentence to a French sentence.

added a special token # at position 0 of **f** (i.e. it is placed before the first real word f_1).

The alignment from **f** to **e** is another function:

$$b : [J] \rightarrow [I]_0$$

Figure 3.2 shows this word alignment.

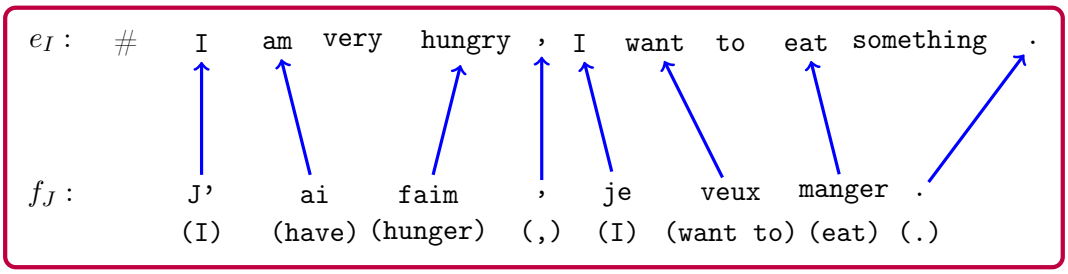


Figure 3.2: The word alignment from a French sentence to an English sentence.

Under this definition, the word alignment mapping is neither injective nor surjective: we can map several indices of the source side to the same index of the target side (e.g. the words "want" and "to" in Figure 3.1), and it is possible that some indices of target side are not mapped to (e.g. the words "very" and "to" in Figure 3.2). This *directional* characteristic of word alignment leads to some interesting consequences, for instance, the English word "to" is mapped to "veux" in Figure 3.1, while no French word is mapped to "to" in Figure 3.2. In other words, for the same pair of sentences, the word alignments in the two directions can be *asymmetric*.

A more general definition of word alignment consists of abandoning functionality. Under this view, for $\mathbf{e} = (e_1, ..., e_i, ..., e_I)$ and $\mathbf{f} = (f_1, ..., f_j, ..., f_J)$, a word alignment $\mathbf{z}$ is a set

of possible word pairs. In other words, $\mathbf{z} \subseteq [I] \times [J]$.[2] Each element in $\mathbf{z}$ represents an alignment link between one word in $\mathbf{e}$ and one in $\mathbf{f}$. Compared to the definition of Brown et al. [1993], the main difference is that word alignment can now be made symmetric. Under the more general definition, a symmetric word alignment between $\mathbf{e}$ and $\mathbf{f}$ is displayed in Figure 3.3:

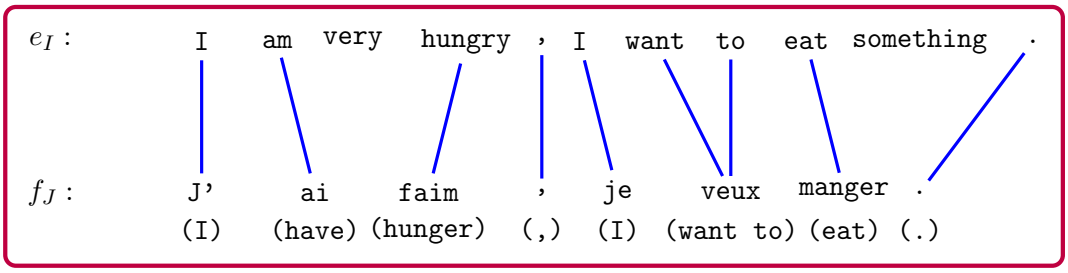


Figure 3.3: The word alignment between a French sentence and an English sentence, under the general definition of word alignment.

In computational linguistics, automatic word alignment is a well-known problem. From the linguistic perspective, correspondences between parallel sentences do not necessarily decompose at the word level. Translations often involve phrase-to-phrase correspondences, not word-to-word. From the computational perspective, the search space of finding the best word alignment is enormous. Using the definition of Brown et al. [1993], the search of the best word alignment from $\mathbf{e}$ to $\mathbf{f}$ amounts to choosing I links from $(J+1)^I$ candidates. Under the more general definition, the search consists of picking one of the $2^{I \times J}$ subsets of $[I] \times [J]$. In both cases, the computational complexity prohibits any alignment algorithm with exhaustive search schemes on the large-scale parallel corpora nowadays employed, which usually contain millions of sentences, e.g. the English side of the Europarl corpus contains 2,218,201 sentences. The contemporary state of the art in word alignment development seems to be focusing on its empirical performance. Generally, the goal of such development is to efficiently deliver word alignment links that would lead to better performance of targeted applications, e.g. SMT. In this chapter, we also focus on the methodological aspect of word alignment.

Our discussion on word alignment is driven by the study on confidence estimation of alignments. Because of the importance of word alignment, a rich literature exists on the subject, which is still under active development. Our discussion is not exhaustive. We focus on probabilistic word alignment models, which provide principled, interpretable scores for confidence estimation, e.g. using the posterior probabilities of links. Excellent non-probabilistic alignment methods have also been proposed. For instance, Ker and Chang [1997] use word class similarity scores to obtain word alignment, Tiedemann [2003a] proposes an alignment

[2] We use $\mathbf{z}$ for alignments under this more general definition, and use $\mathbf{a}$ to emphasize directional alignments. Note directional alignments can also be expressed using the more general definition.

method based on clues (combining several types of association information). Lardilleux et al. [2012] employ an iterative bi-segmentation method. Please refer to [Tomeh, 2012] for an exhaustive presentation of word alignment methods.

3.1 Classical Methods: IBM Models and HMM

The five IBM models proposed by Brown et al. [1993] and the HMM-based model of Vogel et al. [1996] constitute the foundation of studies on word alignment. Och and Ney [2003] present a systematic description and comparison of IBM and HMM models. Several highly-optimized implementations, e.g. GIZA++ [Och and Ney, 2003], MGIZA++ [Gao and Vogel, 2008], are widely used in NLP research practices.

IBM models and HMM are principled probabilistic models, allowing the use of well-established statistical estimation techniques, such as the Expectation-Maximization (EM) algorithm, to adjust model parameters. These models describe a noisy-channel generation process:

$$p(\mathbf{f}|\mathbf{e}) = \sum_{\mathbf{a}} p(\mathbf{a}, \mathbf{f}|\mathbf{e})$$

where each $\mathbf{a}$ is a valid alignment, represented by a sequence of discrete *latent variables* $\mathbf{a} = \{a_j | j \in [J], a_j \in [I]_0\}$. Models differ on the decomposition of $p(\mathbf{a}, \mathbf{f}|\mathbf{e})$, and on the parameterization. Once the model form has been specified, parameters θ are learnt by maximizing the log-likelihood on a parallel corpus:

$$\begin{aligned} \theta' &= \arg\max_{\theta} \sum_{m} \log p(\mathbf{f}^{(m)}|\mathbf{e}^{(m)}; \theta) \\ &= \arg\max_{\theta} \sum_{m} \log \sum_{\mathbf{a}} p(\mathbf{a}, \mathbf{f}^{(m)}|\mathbf{e}^{(m)}; \theta) \end{aligned}$$

where m is an index running over parallel sentences. The EM algorithm is then applied to find the optimal parameters θ. Recently, there have been some works on Bayesian word alignment, using techniques such as Gibbs sampling [Mermer and Saraclar, 2011]. Typically, the training process is ran in both directions, yielding two sets of parameters, and eventually two directional alignments.

In the SMT community, several concerns about IBM and HMM word alignment models have been raised:

- *symmetry*: the definition of word alignment as a function leads to asymmetric alignments. If non-directional alignment links are required, we have to convert directional ones, mostly using heuristics. It might be beneficial to investigate alignment models that are designed to be symmetric.

- *complexity*: Liang et al. [2006] classified IBM and HMM models into two categories: sequence-based (IBM 1, 2 and HMM) and fertility-based (IBM 3, 4 and 5). While

fertility-based models tend to deliver more accurate results on large scale parallel corpora, they are very complex, intractable statistical models, involving many approximation methods. Even with the sophisticated implementations in GIZA++ and MGIZA, fertility-based word alignment is still slow, and very memory-consuming. Thus, an important research direction is to enhance sequence-based models, such that they achieve alignment results comparable to fertility-based models, without significantly decreasing the training efficiency.

– *phrase extraction*: in phrase-based SMT (PBSMT), word alignment is an intermediate step to produce a phrase table, via various heuristics. For this purpose, it is interesting to study word alignment methods that lead to better phrase generation.

These considerations have motivated numerous studies, attempting to improve classical word alignment models from different perspectives.

3.2 Enhancing Sequence-based Word Alignment Models

Sequence-based word alignment models, namely IBM 1, 2 and HMM, are easy to understand, efficient, and admit efficient exact inference. Although outperformed by fertility-based models in SMT, they are often applied in other tasks, e.g. parallel sentence identification [Munteanu and Marcu, 2005] and sentence alignment [Moore, 2002] (see § 2.2.2). Much work has been done to enhance sequence-based models. In the following, we review a few methodologically representative proposals.

3.2.1 Improving the Training Procedure

Using the EM algorithm to maximize data likelihood (MLE) is the main training paradigm for all IBM models and HMM. However, Moore [2004] identifies two important problems of IBM 1, caused by this training method:

– garbage collector effect: in a directional alignment computed by IBM 1, it is often the case that many source words (especially untranslated ones) are spuriously aligned to rare target words (that is, target words which occur very few times in the corpus). The reason of this problem is not hard to see. Since rare target words have few candidate correspondences, the translation distribution conditioned on rare words tend to be sharply peaked, where the few correspondences are attributed large probability masses. In decoding, these large probability values can beat the translation scores with other target words, leading to the garbage collector phenomenon. Although only IBM 1 is discussed here, the garbage collector problem occurs in all IBM models and HMM [Brown et al., 1993; Och and Ney, 2003]. It should be noted that in other models, alignment and fertility distributions also take part in the garbage collector effect.

– null links: IBM 1 tends to align few source words to the target null token. In a sense, this is the counterpart of the garbage collector problem: all source words in the corpus are candidate correspondences of the null token since it is added to every target sentence, thus the translation distribution conditioned on null is very flat. In decoding, other target words with fewer candidate correspondences can rank over null.

For the second problem, Moore [2004] proposes to add a fixed number of extra null tokens to each target sentence. Och and Ney [2003] report that in more complex models (e.g. IBM 3 and 4), too many source words are aligned to the target null token. Thus, this problem seems to be specific to IBM 1.

The first problem has been discussed in several studies. Och and Ney [2003] interpolate alignment and fertility distributions with auxiliary, less peaky distributions. [Moore, 2004] uses the add-n technique to smooth expected counts during each EM iteration, as well as a carefully-chosen starting point for the EM. Vaswani et al. [2012] add an approximated l_0-norm prior to the lexical translation probabilities to reduce overfitting. The objective in the M-step becomes non-convex, so projected gradient descent is employed. Zhang and Chiang [2014] smooth the distributions inside the EM using an extended version of Kneser-Ney smoothing [Kneser and Ney, 1995], which can be applied to *expected counts* instead of integral counts. Wang et al. [2015] argue that the reason of the garbage collector effect is that erroneous alignment links involving rare words enforce themselves during EM iterations. To avoid this effect, they use maximum leave-one-out likelihood estimation instead of the MLE to estimate parameters. Now, every sentence pair has its own parameter values. When updating the parameters of one pair (the M-step), the alignment information from that pair itself is not used, only the expected counts from all other pairs participate in the computation of the update.

3.2.2 Convexifying IBM 1 and 2

Toutanova and Galley [2011] prove that the objective function of IBM 1 is concave but not strictly concave in its variables, namely the lexical translation probabilities (this formula only shows the relevant part of the objective):

$$\sum_m \sum_j \log \sum_i \theta_{\texttt{lex}}(f_j^{(m)}|e_i^{(m)})$$

In other words, several assignments can achieve the same optimal likelihood, while they might differ from each other in terms of alignment quality. In consequence, the initialization of parameters impacts the model performance, which might be an undesirable feature in practice. To remedy this issue, Simion et al. [2015] modify the objective to

$$\sum_m \sum_j \log \sum_i h_{i,j,m}(\theta_{\texttt{lex}}(f_j^{(m)}|e_i^{(m)}))$$

where $h_{i,j,m} : \mathbb{R}_+ \to \mathbb{R}_+$ can be any strictly concave function. The new objective is strictly concave in $\theta_{\texttt{lex}}$. Thus theoretically the initialization does not matter any more. Another nice property of the reformulation is that the function form $h_{i,j,m}$ allows encoding extra information such as positions.

The objective function of IBM 2 is non concave (again only relevant parts of the objective):

$$\sum_m \sum_j \log \sum_i \theta_{\texttt{lex}}(f_j^{(m)}|e_i^{(m)})\theta_{\texttt{dis}}(i|j)$$

Thus, EM is only guaranteed to find a local maximum. Simion et al. [2013] introduce a convex relaxation for the IBM 2 objective:

$$\frac{1}{2}\sum_m \sum_j \log \sum_i q(i,j,m) + \frac{1}{2}\sum_m \sum_j \log \sum_i \theta_{\texttt{lex}}(f_j^{(m)}|e_i^{(m)})$$

with additional constraints:

$$q(i,j,m) \geq 0, \quad q(i,j,m) \leq \theta_{\texttt{dis}}(i|j), \quad q(i,j,m) \leq \theta_{\texttt{lex}}(f_j^{(m)}|e_i^{(m)})$$

Now the objective becomes concave. Simion et al. [2013] use sub-gradient methods to find optimal model parameters.

Convexifying IBM models does not necessarily ensure better alignments qualities, as the objective concerns data likelihood, not alignment quality metrics. Nevertheless this technique works well in practice. In [Simion et al., 2013], the modified IBM 2 achieves comparable performance to the standard IBM 2 (measured by AER and F-score, see § 4.2.2); in [Simion et al., 2015], a carefully chosen form of the new objective leads to 30% AER improvement or 10% F-score improvement over the standard IBM 1.

3.2.3 Reparameterization of IBM 2

Dyer et al. [2013b] argue that the uniform distortion distribution assumption of IBM 1 is too strong, while the multinomial distortion distributions of IBM 2 is an over-parameterization, causing overfitting problems. They propose a reparameterization of IBM 2, in which the distortion distributions take the form of a log-linear model (under the condition $0 < j \leq J$):

$$\begin{aligned} h(i,j,I,J) &= -|\frac{i}{I} - \frac{j}{J}| \\ \theta'_{\texttt{dis}}(j|i,I,J) &= \frac{\exp\left(\lambda h(i,j,I,J)\right)}{Z_\lambda(i,m,n)} \end{aligned}$$

Hence, the new model is parameterized with only the null link prior p_0 and λ, much less than a typical IBM 2. The value of λ controls the level of encouragement of alignment links around the diagonal. Crucially, through clever usages of arithmetic facts, it is possible to

compute each partition function $Z_\lambda(i, m, n)$ in $O(1)$. The computation of gradients is also very efficient. As a result, the new model still admits fast inference, which warrants the use of the standard EM to optimize parameters.

This reparameterization is simple yet surprisingly effective. Dyer et al. [2013b] report that the new IBM 2 leads to better translation quality than the much more complex IBM 4, with approximately 10 times less training time.[3]

3.3 Generative Symmetric Models

Directional alignments generated by IBM and HMM models are typically made symmetric using heuristics. To avoid this step, many have worked on symmetric models. We discuss generative symmetric models in this section, deferring discriminative models to the next section.

3.3.1 Alignment by Agreement

Liang et al. [2006] observe that the intersection of two directional word alignments generally outperforms each model alone. That is, the alignment links that the two directional models agree on are more accurate. They thus propose an alignment method which encourages the two directional models to agree *at training time*.[4] Note $\mathbf{x} = (\mathbf{e}, \mathbf{f})$ a pair of parallel sentences, and note the two directional alignment models as:

$$p_1(\mathbf{x}, \mathbf{z}; \theta_1) = p(\mathbf{e})p(\mathbf{a}, \mathbf{f}|\mathbf{e}; \theta_1) \quad p_2(\mathbf{x}, \mathbf{z}; \theta_2) = p(\mathbf{f})p(\mathbf{b}, \mathbf{e}|\mathbf{f}; \theta_2)$$

where θ_1 and θ_2 are respectively the sets of parameters of the two directional models. Liang et al. [2006] encourage the agreement between the two directional models by coupling them in the training objective:

$$J(\theta_1, \theta_2) = \sum_{\mathbf{x}} \left\{ \log p_1(\mathbf{x}; \theta_1) + \log p_2(\mathbf{x}; \theta_2) + \log \sum_{\mathbf{z}} p_1(\mathbf{z}|\mathbf{x}; \theta_1) p_2(\mathbf{z}|\mathbf{x}; \theta_2) \right\}$$

The objective promotes both data likelihood and alignment agreement. Here, the agreement is at the alignment level. Liang et al. [2006] derive an EM-like algorithm for optimizing model parameters. The E-step is $\#P$-complete, so it is replaced by a simple heuristic. For decoding, Liang et al. [2006] use posterior decoding, instead of intersecting two Viterbi alignments. Both joint training and posterior decoding improve the alignment quality (measured by AER).

One major characteristic of the method of Liang et al. [2006] is the strong tendency to generate 1:1 alignment links [Liu et al., 2015], because of the hard constraint in the

[3] The implementation of this method is at `http://github.com/clab/fastalign`.

[4] Implemented in Berkeley Aligner, available at `https://code.google.com/archive/p/berkeleyaligner/`

objective: any alignment containing 1-to-many, many-to-1 or many-to-many links does not contribute to the third term. Recall that $p_1(\mathbf{x}, \mathbf{z}; \theta_1) = p(\mathbf{e})p(\mathbf{a}, \mathbf{f}|\mathbf{e}; \theta_1)$, thus, if in one alignment $\mathbf{z}$, one word in $\mathbf{f}$ aligns with many words in $\mathbf{e}$, $p_1(\mathbf{x}, \mathbf{z}; \theta_1)$ would be 0. For the same reason, any word alignment $\mathbf{z}$ with one word in $\mathbf{e}$ aligning to many words in $\mathbf{f}$ would obtain $p_2(\mathbf{x}, \mathbf{z}; \theta_2) = 0$. Hence, only alignments containing exclusively 1:1 and null links contribute to the third term of the objective. This will push parameters to move in a way such that these kinds of alignments are favored. To relax this hard constraint, Liu et al. [2015] generalize the agreement framework as:

$$J(\theta_1, \theta_2) = \sum_{\mathbf{x}} \left\{ \log p_1(\mathbf{x}; \theta_1) + \log p_2(\mathbf{x}; \theta_2) + \log \sum_{\mathbf{z}_1} \sum_{\mathbf{z}_2} p_1(\mathbf{z}_1|\mathbf{x}; \theta_1) p_2(\mathbf{z}_2|\mathbf{x}; \theta_2) \Delta(\mathbf{z}_1, \mathbf{z}_2) \right\}$$

where $\Delta(\mathbf{z}_1, \mathbf{z}_2)$ is a loss function measuring the difference between the two alignments. This framework contains the model of Liang et al. [2006] as an instance, with $\Delta(\mathbf{z}_1, \mathbf{z}_2)$ being an indicator function. Other loss functions would impose less severe constraints, and permit the learning of other link types. Liu et al. [2015] further generalized the model to perform joint word alignment and phrase segmentation.

3.3.2 Posterior Constrained EM

Graça et al. [2010] applied the Posterior Regularization (PR) framework introduced in [Graça et al., 2007] to estimate the parameters of the HMM alignment model, in a way such that the posterior probabilities of latent alignment variables $\mathbf{a}$ satisfy predefined constraints.[5] In the PR framework, we first construct a convex set of distributions for latent alignment variables $\mathbf{a}$:

$$\mathcal{Q}_{\mathbf{x}} = \{q(\mathbf{a}|\mathbf{x}) : \exists \xi,\ \mathbf{E}_q[\mathbf{f}(\mathbf{x}, \mathbf{a})] - \mathbf{b}_{\mathbf{x}} \leq \xi;\ ||\xi||_2^2 \leq \epsilon^2\}$$

where $\mathbf{f}$ is a vector of feature functions and $\mathbf{b}$ is a real vector. Each inequality imposes a certain constraint for $\mathbf{a}$. For word alignment, Graça et al. [2010] use two types of constraints:

- bijectivity: only 1-to-1 and null alignment links are allowed;
- symmetry: the two HMMs should return the same set of links.

Both constraints are encoded as linear feature functions. Computing expectations requires marginals $q(a_{ij}|\mathbf{x})$, which can be obtained using forward-backward in HMMs. We then optimize model parameters θ, using a modified EM:

$$\mathbf{E}: \quad q(\mathbf{a}; \mathbf{x}) := \arg\min_{q(\mathbf{a};\mathbf{x}) \in \mathcal{Q}_{\mathbf{x}}} \mathtt{KL}(q(\mathbf{a}; \mathbf{x}) \ ||\ p(\mathbf{a}|\mathbf{x}; \theta))$$

$$\mathbf{M}: \quad \theta' = \arg\max_{\theta} \sum_{\mathbf{x}} \sum_{\mathbf{a}} q(\mathbf{a}; \mathbf{x}) \log p(\mathbf{a}, \mathbf{x}; \theta)$$

[5] Implemented in the PostCAT toolkit: `http://www.seas.upenn.edu/~strctlrn/CAT/CAT.html`.

Graça et al. [2010] solve the optimization problem in the E-step using the dual subgradient method. Experiments confirm the effectiveness of the PR framework, as both bijectivity and symmetry are significantly promoted in resulting alignments.

The goal of Graça et al. [2010] is similar to the work of Liang et al. [2006], in that they both aim at generating symmetric, 1-to-1 word alignments, and both perform a joint training of two directional HMM alignment models. However, the methodologies are very different. Liang et al. [2006] achieve this goal by promoting alignments on which the two HMMs agree, and use standard EM (modulo heuristics) for estimation. Graça et al. [2010] use a standard data likelihood objective, but modify the EM algorithm such that during each iteration, the proposal distribution $q(\mathbf{a})$ satisfies pre-defined constraints, in order to push the posterior distribution $p(\mathbf{a}|\mathbf{x})$ toward the constraint space, namely the expected alignment types. The PR framework is theoretically attractive since it provides means to encode constraints into the model that can otherwise be difficult to impose. Kamigaito et al. [2014] use PR to restrict function and content word matching frequencies, which proves useful for SMT on grammatically different languages pairs such as Japanese-to-English.

3.3.3 Bidirectional Models

DeNero and Macherey [2011] also combine two directional HMM models to promote symmetric word alignments, as well as to enforce 1-to-1 phrase alignments. They embed the two HMMs into a single graphical model, displayed in Figure 3.4. Here, a_j (resp. b_i) are

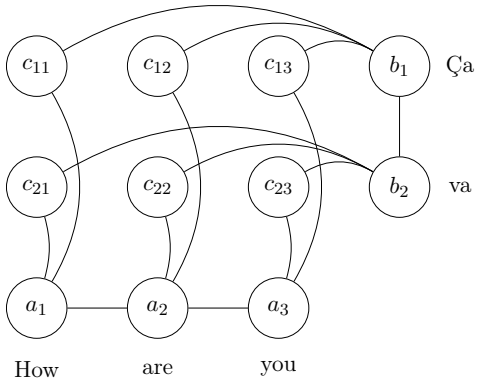


Figure 3.4: The graphical word alignment model of DeNero and Macherey [2011].

variables of the English-French (resp. French-English) HMM, which factors into single node potentials (i.e. factors of the graphical model that involves a single variable) on a_j (resp. b_i) and edge potentials (i.e. factors that involves two variables) between a_{j-1} and a_j (resp. b_{i-1} and b_i). c_{ij} are auxiliary binary variables which represent the final alignment, e.g. if $c_{23} = 1$ in Figure 3.4, then the link "you - va" is member of the result.

By using $\mathbf{c}$ (instead of $\mathbf{a}$ and $\mathbf{b}$) to represent the final alignment, the model becomes symmetric. A number of edge potentials involving c_{ij} (called the *coherence edges*) have been defined to ensure that final alignments are well-formed. This formulation allows imposing further structural constraints on the final alignment, through proper definitions of coherence edge potentials. For instance, DeNero and Macherey [2011] use the following edge potential:

$$\mu^{(c)}_{a_j,c_{i'j}}(i,k) = \begin{cases} 1 & i \neq i' \wedge k = 0 \\ e^{-\alpha} & |i - i'| = 1 \wedge k = 1 \\ 0 & |i - i'| > 1 \wedge k = 1 \end{cases}$$

where α is a hyper-parameter and k is a binary indicator. Here, when the j-th English word aligns to the i-th ($a_j = i$) and the i'-th ($c_{i'j} = 1$) French words with $|i - i'| > 1$, the edge potential gets zero value. In other words, such an alignment is attributed zero probability. Hence, the model only admits phrase alignments with a maximum length of 3 on each side.

Decoding in this graphical model consists of finding the maximum probability assignment:

$$\max_{\mathbf{a},\mathbf{b},\mathbf{c}} \mathcal{L}(\mathbf{a},\mathbf{b},\mathbf{c})$$

which is intractable since the model structure is not a tree. However, noticing that only $\mathbf{c}$ couples $\mathbf{a}$ and $\mathbf{b}$ together in the log-likelihood, DeNero and Macherey [2011] adopt the dual decomposition approach [Rush et al., 2010] to perform the inference, by transforming the optimization problem into:

$$\begin{array}{ll} \max_{\mathbf{a},\mathbf{b},\mathbf{c}^{(a)},\mathbf{c}^{(b)}} & f(\mathbf{a},\mathbf{c}^{(a)}) + g(\mathbf{b},\mathbf{c}^{(b)}) \\ \texttt{subject to} & \mathbf{c}^{(a)}_{ij} = \mathbf{c}^{(b)}_{ij} \;\; \forall\, 1 \leq i \leq I, 1 \leq j \leq J \end{array}$$

whose Lagrange dual is

$$\min_{\mathbf{u}} \left(\max_{\mathbf{a},\mathbf{c}^{(a)}} \left[f(\mathbf{a},\mathbf{c}^{(a)}) + \sum_{i,j} u_{ij} c^{(a)}_{ij} \right] + \max_{\mathbf{b},\mathbf{c}^{(b)}} \left[g(\mathbf{b},\mathbf{c}^{(b)}) - \sum_{i,j} u_{ij} c^{(b)}_{ij} \right] \right)$$

Crucially, now the two sub-problems are independent and both tractable (with tree structures), as displayed in Figure 3.5. Thus one can use dual subgradient methods to solve the optimization problem. Ideally, one would obtain $\mathbf{c}^{(a)} = \mathbf{c}^{(b)}$ upon convergence of the dual optimization. In practice, the convergence is difficult to achieve: DeNero and Macherey [2011] report a 6.2% convergence rate after 250 iterations. In consequence, heuristic methods such as intersection, union, and grow-diagonal-final-and are still needed to merge $\mathbf{c}^{(a)}$ and $\mathbf{c}^{(b)}$. Nonetheless, experimental results show that the bidirectional model does promote symmetric alignments, leading to better performance than directional baselines (measured with word alignment precision, recall and AER). The model also enjoys a better phrase extraction accuracy, though the impact on phrase-based SMT systems seems modest.

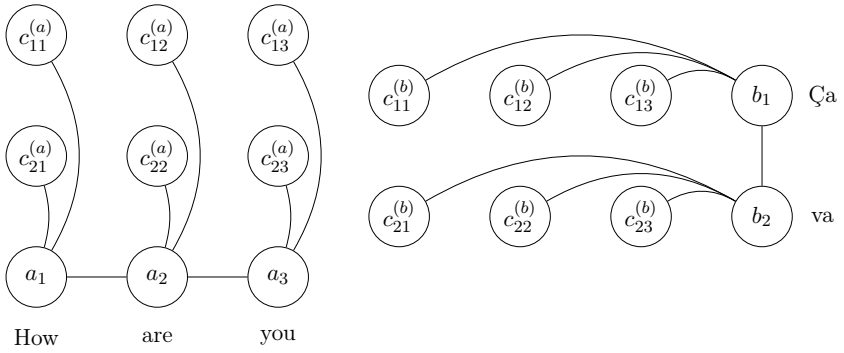


Figure 3.5: The graphical model of the dual problem.

Following DeNero and Macherey [2011], Chang et al. [2014] propose another bidirectional word alignment model. In a nutshell, Chang et al. [2014] combine two directional HMMs to generate symmetric alignments, by introducing constraints on alignment variable assignments. The model departs from [DeNero and Macherey, 2011] in two ways: first, in [DeNero and Macherey, 2011], constraints are encoded into the model through edge potentials, while in this work constraints are expressed explicitly as a part of the optimization program; second, [Chang et al., 2014] deal with the intractable inference problem by selectively relaxing constraints, so that modified versions of the Viterbi algorithm can be applied to efficiently solve the relaxed problem. Together with additional techniques such as search space pruning, the new model is able to solve 86% of sentence pairs of the same data set of DeNero and Macherey [2011], which is a large improvement. This method is theoretically attractive, efficient, and does not require post-hoc heuristic merging steps. Unfortunately, the resulting alignments are slightly worse than the best results of [DeNero and Macherey, 2011], as measured by AER and phrase extraction accuracy.

3.3.4 The Monolink Model

The aforementioned models promote symmetry by combining directional alignments. On the contrary, the monolink alignment model of Melamed [2000]; Cromières and Kurohashi [2009] does not embed any directional subcomponent. A major modeling feature of monolink is the abandon of the noisy-channel assumption, replaced by the following generation process:

$$p(\mathbf{x}) = \sum_{\mathbf{z}} p(\mathbf{z}, \mathbf{x}) \propto \sum_{\mathbf{z}} \prod_{(i,j)\in\mathbf{z}} p(e_i, f_j)$$

where $i \in [I]_0$, $j \in [J]_0$, and z is any accepted alignment. In [Cromières and Kurohashi, 2009], a valid alignment z should not contain any one-to-many or many-to-many link. That is, if e_i is aligned to f_j, then neither should be aligned to any other word. In consequence, only 1-to-1 and null links are modeled, hence the name "monolink".

Cromières and Kurohashi [2009] construct a graphical model for each sentence pair. The graph contains one variable ve_i for each e_i, one variable vf_j for each f_j, and an edge between any source-target word pair, as illustrated in Figure 3.6. The monolink alignment

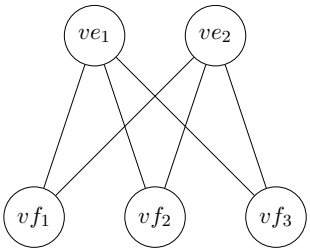


Figure 3.6: The monolink graphical model.

constraints are encoded into the model via edge potential functions, similar to the way of DeNero and Macherey [2011]. Node potentials contain parameters $p(e_i, f_j)$. EM for parameter learning requires:

$$E((i,j)|\mathbf{x}) = \sum_{\mathbf{z}|(i,j)\in\mathbf{z}} p(\mathbf{z}|\mathbf{x})$$

which in turn requires the marginal probabilities of the nodes ve_i or vf_j. This inference is intractable since the graphical model structure is not a tree. Cromières and Kurohashi [2009] apply the sum-product Loopy Belief Propagation (LBP) algorithm [Murphy et al., 1999] to perform approximate inference. After parameter optimization is finished, one can use max-product LBP to decode the best monolink alignment.

Cromières and Kurohashi [2009] further enrich the model by adding distortion information. Constraints on the distortions are also encoded into the extended graphical model via edge potentials. Again LBP is used for inference. With these simple extensions, monolink is able to achieve comparable word alignment qualities with the much more complicated IBM 4, in terms of AER, precision and recall, on an English-French data set.

3.4 Discriminative Word Alignment

Discriminative models constitute another important body of work on word alignment. Compared to the generative models presented in previous sections, discriminative models avoid the (potentially) complicated generation process, and can accommodate rich feature sets. Many discriminative models require supervision gold alignment data, a matter that we discuss extensively in § 4.2.3.

3.4.1 Link-level Models

Ayan and Dorr [2006]; Tomeh et al. [2014] make the assumption that the existence of one alignment link is independent of other links. Hence they model each possible link between $\mathbf{e}$ and $\mathbf{f}$ by a Maximum Entropy (MaxEnt) model:

$$p(z_{i,j}|\mathbf{x}) = \frac{\exp\left(\lambda^\top \mathbf{h}(z_{i,j}, \mathbf{x})\right)}{\sum_{z_{i,j}} \exp\left(\lambda^\top \mathbf{h}(z_{i,j}, \mathbf{x})\right)}$$

where λ is the parameter vector, $\mathbf{h}(z_{i,j}, \mathbf{x})$ the feature vector, and $z_{i,j} \in \{0, 1\}$. Recall $\mathbf{x} = (\mathbf{e}, \mathbf{f})$, $I = |\mathbf{e}|$, $J = |\mathbf{f}|$ and $\mathbf{z} \subset [I] \times [J]$. Here, word alignment is cast as a binary classification problem, for which training and inference are very efficient. The MaxEnt model obtains directly link posterior probabilities, which can be used in several subsequent applications, e.g. confidence estimation and phrase pair extraction.

For link-level models, using all links in the alignment matrix as training instances would lead to a data imbalance problem, as there are roughly quadratically many negative examples, and only linearly many positive ones. To avoid this problem, Tomeh et al. [2014] propose to first run automatic aligners on the corpus. Only links appearing in the union of automatic directional alignments are then used as training examples. The same technique is applied during decoding.

3.4.2 Sequence Models

Blunsom and Cohn [2006] use first-order linear chain Conditional Random Fields (CRFs) to model directional word alignments. The alignment $\mathbf{a}$ from $\mathbf{e}$ to $\mathbf{f}$ is:

$$p(\mathbf{a}|\mathbf{x}) = \frac{1}{Z(\mathbf{x})} \exp\left(\sum_i \lambda^\top \mathbf{h}(a_{i-1}, a_i, \mathbf{x})\right)$$

where $Z(\mathbf{x})$ is the partition function, $a_i \in [J]_0$. In a sense, this is the conditional version of the HMM alignment model. We can use modified versions of forward-backward and Viterbi algorithms to perform training and inference. Compared to the HMM, CRFs incorporate a rich set of features, even including alignment scores of complicated generative models such as IBM 4. Compared to link models, CRFs can make use of Markov features, such as the `jump_width` feature in [Blunsom and Cohn, 2006]:

$$\texttt{jump\_width}(t-1, t) = abs(a_i - a_{i-1} - 1)$$

which helps to model alignment positions.

As the HMM, the sequence CRFs alignment model encodes directional word alignments. We can use standard heuristics to perform the symmetrization, if necessary.

3.4.3 Global Models

Liu et al. [2005] present a log-linear framework for symmetric word alignment:

$$p(\mathbf{z}|\mathbf{x}) = \frac{\exp\left(\lambda^\top \mathbf{h}(\mathbf{z}, \mathbf{x})\right)}{\sum_{\mathbf{z}} \exp\left(\lambda^\top \mathbf{h}(\mathbf{z}, \mathbf{x})\right)}$$

Liu et al. [2005] consider three types of features: IBM 3 scores, cross-lingual POS transition scores and dictionary-based word match scores. Since the model does not encode any structural assumption, inference requires an enumeration of the $2^{I \times J}$ possible $\mathbf{z}$s, and is intractable. Liu et al. [2005] resolve to greedy search for decoding, and use N-best lists for computing marginals in training. Moore [2005] proposes a similar model, with features relying heavily on word cooccurrence and alignment link frequencies. Decoding is performed using beam search, and learning uses a modified version of averaged perceptron.

The models of Liu et al. [2005] and Moore [2005] operate at the alignment level, and make no structural assumptions, thus they both face difficult inference problems. Another global model is proposed by Niehues and Vogel [2008], in which the whole alignment matrix is represented by a two-dimensional CRF:

$$p(\mathbf{z}|\mathbf{x}) = \frac{1}{Z(\mathbf{x})} \prod_{c \in \mathcal{C}} \Phi(\mathbf{z}_c, \mathbf{x})$$

where $\mathbf{z} = \{z_{ij} \in \{0, 1\} | i \in [I], j \in [J]\}$, $\mathcal{C}$ are cliques in the graphical structure, and $\Phi(\mathbf{z}_c, \mathbf{x})$ are clique potentials (generally instantiated as a log-linear combination of feature functions). Niehues and Vogel [2008] define a very rich set of clique potentials, some with complicated structures, e.g. the fertility feature for word e_i couples all variables $\{z_{ij} | j \in [J]\}$ together. The resulting model is highly complex, and inference is intractable. Niehues and Vogel [2008] use the LBP algorithm to perform approximate inferences, as in the work of Cromières and Kurohashi [2009].

3.5 Conclusions

In this chapter, we have very briefly reviewed the literature on word alignment. Our interest in such techniques originates from our study of confidence estimation for alignment links, thus we have only covered a small set of methods, preferring those with clear probabilistic interpretations, and leaving many successful ones undiscussed, such as large-margin trained discriminative models [Taskar et al., 2005; Lacoste-Julien et al., 2006] and neural network based word alignment [Yang et al., 2013; Tamura et al., 2014] (for lack of systematic analysis frameworks).

It is very difficult to make a comprehensive evaluation of alignment methods, due to several reasons. First, many word alignment methods are highly technical and hard to reimplement. Second, intrinsic evaluation faces the scarce resource problem: gold word

alignments exist in few language pairs, and producing gold word alignment data is hard in itself. Third, extrinsic evaluation can be slow, and difficult to analyze. In consequence, researchers quite often run one or several popular word alignment toolkits, then choose the results using various heuristics. In our opinion, this situation calls for more efforts on evaluation methodologies and resources, which would help to understand strengths and drawbacks of word alignment methods, as well as on confidence estimation, such that applying alignments could be easier and/or more effective. In Chapter 7, we will present our work on confidence measures for word alignment.

Chapter 4

Manual Alignments

Bitext alignment is an important ingredient in many modern natural language processing (NLP) systems. For instance, sentence alignment has been applied in translator training [Simard et al., 1993b], translation checking [Macklovitch, 1994b], language learning [Nerbonne, 2000; Kraif and Tutin, 2011], and bilingual reading [Pillias and Cubaud, 2015; Yvon et al., 2016]. Sentence alignment typically precedes word alignment, which is a critical component of most statistical machine translation (SMT) systems [Brown et al., 1993; Koehn et al., 2003],[1] as well as in bilingual lexica extraction [Smadja et al., 1996], word sense disambiguation [Diab and Resnik, 2002], cross lingual transfer [Täckström et al., 2013; Aufrant et al., 2016], medical information collection [Prud'hommeaux and Roark, 2015], etc. Thanks to a sustained research effort, many alignment methods have been developed. Two recent reviews of bitext alignment methods are in [Wu, 2010; Tiedemann, 2011].

Different applications might favor alignments computed in different ways. For instance, Lambert et al. [2005] pointed out that bilingual lexica extraction requires high precision single-word links, while machine translation (MT) prefers high recall alignments. Therefore, the evaluation of bitext alignment methods, in particular sub-sentential alignments, is often carried out in an *extrinsic* way, by comparing the performance of the targeted application (e.g. a SMT system) before and after using the alignment method, keeping other components unchanged. However, extrinsic evaluations have several pitfalls. First, such evaluations can be computationally expensive. For large scale parallel corpora that are used nowadays, such as the Europal [Koehn, 2005], the word alignment process can take up to days. Second and perhaps more importantly, it might be hard to understand the performance of alignment methods using extrinsic evaluations. Applications can be highly complex systems containing many subcomponents, involving complex interactions. Even if an alignment method leads to better end-to-end performance, we might not be able to tell which characteristics of the new alignments cause the improvement, since analyzing the inner mechanism of the application system is too hard. For example, in phrase-based SMT,

[1] With the exception of the recently introduced NMT [Sutskever et al., 2014].

bitext alignment lies at the very beginning of the process, and is followed by many steps, some of which are heuristic and difficult to analyze quantitatively (e.g. the generation of phrase pairs). Despite much effort [Fraser and Marcu, 2007; Ganchev et al., 2008], the MT community still struggles to understand the relation between word alignment quality and translation quality [Wang et al., 2015]. For these reasons, it is also necessary to perform *intrinsic* evaluations for bitext alignment methods, in hope that they would provide more direct, easy-to-understand reflections of the alignment quality, and would permit finer grain analyses of the produced alignments.

Intrinsic evaluations of a bitext alignment method require three types of resources: an annotation scheme for alignments, a reference alignment set (the *gold standard*), and an evaluation metric. The annotation scheme specifies the formal representation of alignments. Crucially, the reference alignment, the metric and the automatic alignment should all use the same annotation scheme; otherwise, the evaluation would be inaccurate, or even impossible. Given the scheme, we collect gold standard alignments by manually annotating certain bitexts; then use the alignment method to produce automatic alignments for the same bitexts; finally we compute an evaluation score according to the metric by comparing the reference and the computed alignments. Another important use of gold alignments is to serve as supervision instances to train alignment models.

Constructing reference alignment annotations, however, can be a challenging task. In some cases, this can be due to a lack of a clear annotation scheme. In others, annotation schemes can vary a lot, depending on the targeted applications, language pairs, etc. In this chapter, we review existing resources created for intrinsic bitext alignment evaluation, focusing on sentence-level and word-level alignments. For sentence alignment, there exist well accepted annotation schemes and evaluation measures; the creation of reference alignments is relatively easy. Evaluating word alignment is more complicated. We present the annotation schemes and evaluation metrics that are used in the literature, analyzing the challenges faced in the creation of reference alignments, and briefly reviewing existing data sets. Other types of alignments also exist, for instance, phrasal alignments [Volk et al., 2006; Li et al., 2012]. We do not discuss them here, as our study has not included these levels. But they are certainly useful resources for bitext alignment studies.

4.1 Resources for Sentence Alignment Evaluation

We have described the sentence alignment task in § 2.1, and have illustrated its general conventions. These conventions play the role of guidelines for manual sentence alignment. For convenience, we repeat them here:

– Each side of an alignment link should be a *flat contiguous segment*, that is, a consecutive group of sentences, unless it is empty. If E_i and E_{i+2} are both inside a link, then so must be E_{i+1}.

– Alignment links are *monotonic*. If $[E_i; F_j]$ is a link, then no source sentences *following* E_i (e.g. E_{i+1}) can link to target sentences *preceding* F_j (e.g. F_{j-1}).

– Links must be *minimal*, meaning that they cannot be decomposed into strictly smaller links that do not violate previous constraints. For example, if both $[E_i; F_j]$ and $[E_{i+1}; F_{j+1}]$ are good alignment links, then it is incorrect to form a larger link $[\mathbf{E}_i^{i+1}; \mathbf{F}_j^{j+1}]$. Further, a sentence belongs to exactly one (null or non-null) alignment link.

4.1.1 Annotation Schemes

Tiedemann [2003b] developed **Uplug**, a web based toolkit for performing manual alignments, which includes an interface for sentence alignment. Figure 4.1 gives a glimpse of the manual annotation process.

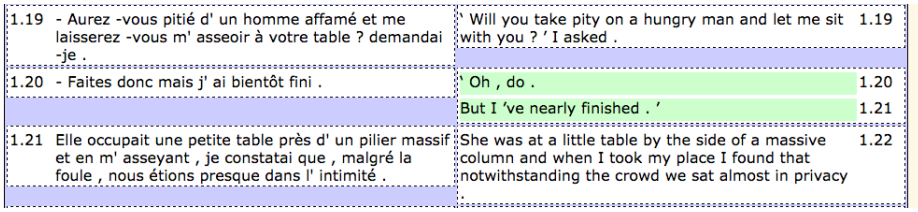


Figure 4.1: The Uplug toolkit for manual sentence alignment.

The representation of sentence alignment is relatively easy. Perhaps the most lightweight representation is the one adopted to distribute bitexts for training MT systems: each side of the bitext is in a separate text file, one sentence per line, with the convention that sentences having the same line number in the source and target file are mutual translations. For sentence alignment evaluation purposes, this representation is however problematic: sentences lack clear, easy-to-reference identifiers, which makes the comparison between gold and hypothesis alignments difficult. Another light-weight format is COAL, used in BAF [Simard, 1998], where sentence alignment for a bitext is represented in three files: two plain text files, each containing one text, and a third alignment file, in the form of a series of pairs $([s_1; t_1], .., [s_n; t_n])$. A pair $[s_i; t_i]$ contains two *character* offsets, each indicating the beginning character of the ith link in the respective side. Simard [1998] also encoded the alignments in the more complex, SGML-based format, the *Corpus Encoding Standard* (CES),[2] which enables to explicitly represent supplementary information. In the CES, alignments are represented by three files. Both sides of the bitext take the form of an XML document, where each paragraph, each sentence and each word is referenced with a unique identifier. A third XML document contains a list of `<link>` elements. Each `<link>` element specifies an alignment link by matching sentence identifiers. CES is supported by Uplug

[2] `http://www.xces.org/`

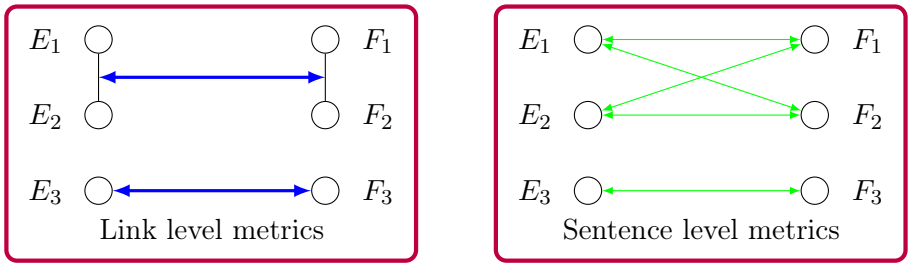


Figure 4.2: The left figure shows one 2:2 alignment link and one 1:1 link. The right figure shows the Cartesian products, with five sentence pairs in total. Sentence level metrics use all five sentence pairs.

and some other automatic sentence alignment tools, such as Yasa [Lamraoui and Langlais, 2013].

4.1.2 Evaluation Metrics

The standard evaluation metrics for sentence alignment are *precision* (P), *recall* (R) and *F-score* (F). We denote the reference sentence alignment as $\mathcal{L}$, and the computed sentence alignment as $\mathcal{A}$. A link $a \in \mathcal{A}$ is correct if it is also an element of $\mathcal{L}$, that is, there exists a link $b \in \mathcal{L}$, such that the two sides of a are exactly the same as the two sides of b. P, R and F defined using this notion of correctness are called *link level metrics* for sentence alignment, since comparisons are made between links:

$$P_l = \frac{|\mathcal{A} \cap \mathcal{L}|}{|\mathcal{A}|} \qquad R_l = \frac{|\mathcal{A} \cap \mathcal{L}|}{|\mathcal{L}|} \qquad F_l = \frac{2P_l R_l}{P_l + R_l}$$

Langlais et al. [1998b] deem link level metrics to be too severe, since they do not take into account that some links are *partially* correct. That is, even though some link in $\mathcal{A}$ is not in $\mathcal{L}$, it might contain many correct sentence pairs. To alleviate this problem, they also define P, R and F *at the sentence level*. Each link in $\mathcal{A}$ is converted into a set of sentence pairs, by taking the Cartesian product of its two sides. This is illustrated in Figure 4.2. The new computed set $\mathcal{A}'$ is the union of all sets of sentence pairs derived from links. In the same way, $\mathcal{L}$ is converted into $\mathcal{L}'$. We can then compute sentence level metrics:

$$P_s = \frac{|\mathcal{A}' \cap \mathcal{L}'|}{|\mathcal{A}'|} \qquad R_s = \frac{|\mathcal{A}' \cap \mathcal{L}'|}{|\mathcal{L}'|} \qquad F_s = \frac{2P_s R_s}{P_s + R_s}$$

Both link level and sentence level metrics are frequently used in subsequent studies. We should note that character level alignment metrics also exist. Link level and sentence level metrics heavily depend on a pre-existing segmentation in sentences, which might be

problematic, for instance, to evaluate segment alignments of transcribed speech. Véronis and Langlais [2000] thus suggested to compute precision and recall scores at the word and character level to enable a fair comparison of alignment techniques that would use different sentence segmentation schemes. Since all our studies use the same segmentation method, we do not use character level metrics in this thesis.

4.1.3 Gold Standard Sentence Alignment Sets

Publicly available gold standard sentence alignment data sets turn out to be scarce. Large collections of parallel sentence pairs, such as the Europarl corpus [Koehn, 2005], are generated automatically, and are not suited for evaluation purpose. Another concern is that certain text genres, such as technical manuals, are usually translated very literally, thus might not be good evaluation materials since all aligners can achieve excellent scores.[3] For such reasons, manual sentence alignments of harder bitexts, such as literary books, constitute valuable resources. In the following, we briefly review publicly available resources. This list is not exhaustive. We only list the datasets that we managed to locate. There must be other valuable ones.

The BAF corpus The purpose of the ARCADE project [Langlais et al., 1998b] was to evaluate bitext alignment techniques. During the project, the BAF corpus was created, which contained English-French bitexts with manual sentence alignments. The data set is available at `http://rali.iro.umontreal.ca/rali/?q=en/node/71`.

The BAF contained four genres of bitexts:

- Institutional (4 institutional reports, proceedings, etc): 14,457 English sentences, 14,725 French sentences, 14,124 links;
- Scientific (5 scientific papers): 3,118 English sentences, 3,134 French sentences, 2,800 links;
- Technical (1 user manual): 3,766 English sentences, 3,871 French sentences, 3,454 links;
- Literary (1 novel): 2,554 English sentences, 3,319 French sentences, 2,521 links.

In total, the BAF corpus contained 23,895 English sentences, 25,049 French sentences, 22,899 links. Note for the last text genre, the literary bitext,[4] the ratio of numbers of English and French sentences (0.770) is much smaller than 1, unlike other text genres.[5] This makes this bitext particularly interesting, as such a ratio suggests stronger translational irregularities, making this bitext more challenging for automatic sentence aligners.

[3] Of course, manual alignment of these easy bitexts are not useless. For example, they can serve as bottomline for alignment methods.

[4] "De la Terre à la Lune" by J. Verne.

[5] For individual bitexts of other genres, the ratio most far away from 1 is 1.12.

The Multext-East Corpora By the joint effort of the MULTEXT-East project [Dimitrova et al., 1998] and the EU Concerted Action TELRI (Trans-European Language Resources Infrastructure), nine versions of "1984" by G. Orwell were aligned, each time between two of the 9 languages studied in MULTEXT-East and TELRI.[6] Alignments are encoded in the CES format. This data set is available at `http://nl.ijs.si/ME/V4/`.

In total, 36 bilingual sentence alignments have been created. Remarkably, a 9-way alignment exists, where all 9 versions of the book participate in a single sentence alignment. That is, every link of this alignment involves 9 languages. This can be useful for multi-lingual research, as well as for the study of multi-sequence alignment problems. The sentence alignments in this data set are relatively regular. For each of the 36 pairs of languages, we have computed the ratio between the numbers of sentences of the two sides. All such ratios lie in the window $[0.95, 1.05]$, the smallest being 0.957 (Estonian/Hungarian). We have also computed the percentage of 1:1 links in the reference alignments. For 19 pairs, the percentage of 1:1 links lies in $[0.95, 1]$; for 35 pairs, the percentage lies in $[0.9, 1]$; the smallest percentage is 0.890 (Estonian/Romanian).

The JOC and MD corpus The ARCADE II project [Chiao et al., 2006] aimed to enrich the resources produced by the ARCADE I project, by promoting the multilingualism. Two corpora were created: the JOC corpus and the MD corpus.

The JOC corpus contained a subset of the Official Journal of the European Community, published in the 9 official languages of the European Community in 1993. In ARCADE II, a portion of the French version was aligned to its corresponding portions in four other languages: English, German, Italian, Spanish:

- French-English: 41,901 links, 43,043 French sentences, 42,986 English sentences;
- French-German: 42,213 links, 43,043 French sentences, 44,023 German sentences;
- French-Italian: 41,600 links, 43,043 French sentences, 42,372 Italian sentences;
- French-Spanish: 41,940 links, 43,043 French sentences, 42,797 Spanish sentences;

The MD corpus contained news articles from the French monthly newspaper "Le Monde Diplomatique", published in multiple languages, including non-European ones. This corpus involved another set of languages:

- French-Arabic: 10,998 links, 13,849 French sentences, 10,844 Arabic sentences;
- French-Greek: 4,242 links, 4,198 French sentences, 4,290 Greek sentences;
- French-Persian: 4,609 links, 5,168 French sentences, 5,263 Persian sentences;
- French-Japanese: 5,069 links, 5,299 French sentences, 5,546 Japanese sentences;

[6] The data set contains the Russian version of the book. But it is not aligned to any other version.

– French-Russian: 4,003 links, 4,231 French sentences, 4,188 Russian sentences;

– French-Chinese: 4,452 links, 5,163 French sentences, 5,523 Chinese sentences;

The sentence alignments in both corpora are relatively regular. Nevertheless, to our knowledge, for many language pairs (such as French-Persian), the ARCADE II project was among the first to produce parallel corpora. The two corpora are both valuable resources for the study of sentence alignment techniques.

Farkas' alignments of literary bitexts The web site of A. Farkas[7] contains a large set of alignments of literary bitexts, which are available for research purposes. For each bitext, an automatic sentence alignment is produced using the Hunalign software [Varga et al., 2005]. Farkas then manually checked some of these alignments. This data set is at `http://www.farkastranslations.com/bilingual_books.php`.

To use this data set as resource for sentence alignment evaluation, we have to resolve two issues: first, the alignment is not defined at the sentence level, rather, small groups of sentences are aligned. In consequence, these alignments are not compatible with the conventional annotation schemes presented in § 2.1.1; second, some of the alignments are not revised manually, so the correctness is not guaranteed for all books. Still, this resource is extremely valuable. Its large amount of content and multilinguality make it a good starting point for large scale evaluations of sentence alignment methods. In this thesis, we will present our effort to make use of this data set in Chapter 6.

4.2 Resources for Word Alignment Evaluation

Compared to sentence alignment, the creation of gold standard word alignments turns out to be a much more challenging task. Most difficulties trace back to the fuzziness in the definition of parallelism. Generally, two lexical units are parallel if they roughly express the same meaning. We handle the sentence alignment task well with this definition, while for word alignment a lot of problems arise.

The first difference lies in the different roles of sentences and words in the expression of sense. In principle, a sentence conveys a complete thought and implies (perhaps implicitly) a clause. Thus, telling whether two sentences express (approximately) the same meaning is generally an easy exercise for a human, and annotators almost always reach the same judgment. On the contrary, a written word's meaning often can only be discovered in conjunction with its neighbours, or the discourse context. Words such as articles ("a", "the", etc) even make no sense by themselves. When aligning words, in many circumstances we have to resort to linking *word groups*, rather than single words. However, deciding the extent of groups can again become a tricky matter. Different languages may use dramatically

[7] © 2014 FarkasTranslations.com.

different constructions for the same meaning, where the individual words are completely unrelated in themselves.

The second difference comes from the impact of surface forms. When aligning sentences, our consideration falls totally on the semantic level, ignoring their inner construction. We deem two sentences as being parallel as long as they give the same understanding, even if the structures have been substantially modified (e.g. deletions/insertions, abstraction/developments). However, whether or not we can ignore linguistic features of words is unclear. For example, how should we deal with unmatched tenses? Can we consider the English word "doing" in the present progressive tense and the French word "faire" (the translation of "do") in the present tense as being parallel (there is not an exact counterpart of the present progressive tense in French)? The answer to such a question depends on the interpretation of parallelism, and different annotators might hold different views. There are many other problems of this kind. Further, such grammatical subtleties vary from language pairs to language pairs.

In consequence, when preparing manual word alignments, a detailed guideline is always designed prior to the annotation, which describes how to handle such complex situations. Still, even with best guidelines, one must expect that some word matchings will remain hard to decide unambiguously, if decidable at all. The research community has long observed and recognized this characteristic of manual word alignments, and has designed annotation schemes and evaluation metrics accordingly. In the following, we first review the annotation schemes (§ 4.2.1) and evaluation metrics (§ 4.2.2). Then we describe the challenges to overcome in the creation of reference alignment sets (Section 4.2.3). In Appendix B, we review projects conducted for the purpose of creating reference word alignments, by briefly describing the particularities of each one, and the resources created.

4.2.1 Annotation Schemes

The original definition of word alignment, as proposed by Brown et al. [1993], is as follows: for a parallel sentence $\mathbf{e}_1^I = (e_1, ..., e_I)$ and $\mathbf{f}_1^J = (f_1, ..., f_J)$ where each e_i $(1 \leq i \leq I)$ or f_j $(1 \leq j \leq J)$ stands for a token, a word alignment is a function which maps a word index of $\mathbf{e}_1^I$ (resp. $\mathbf{f}_1^J$) to a single word index of $\mathbf{f}_1^J$ (resp. $\mathbf{e}_1^I$). Such a pair of indices is called a *word alignment link*. Thus, the following scheme gives a natural representation of word alignment:

- use one line for one alignment link;
- each line takes the form:

`sentenceID` i j

  where `sentenceID` is the identifier of the sentence (usually its index in the parallel corpus).

Since word alignment is defined as a function, it is *directional*: the alignment from e to f satisfies the condition that one word of e (resp. f) is mapped to at most one word of f (resp. e). However, the representation does not enforce such a limitation, i.e. we can encode alignments between groups of words using this scheme.[8]

The representation makes no distinction between links. However, as we have discussed, word alignment is a complicated task. While some word pairs can be easily determined as a good link, others are less obvious and the existence of a link between them is ambiguous. To distinguish between these two cases, Och and Ney [2000] proposed to annotate each link with one of the two tags: **S**ure (S) and **P**ossible (P). For a pair of words, if the annotator considers the link between them is unambiguous, then the S tag is used, and the link is called an S-link; otherwise the P tag is chosen. Och and Ney [2000] further assume that every S-link is also a P-link.[9] Hence, after an annotator finishes processing one bitext, we can collect two sets of links: $\mathcal{S}$ which contains only the S-links, and $\mathcal{P}$ which contains all S-links and P-links. It is guaranteed that $\mathcal{S} \subseteq \mathcal{P}$. Och and Ney [2000] added a field for the S/P tag into the annotation of one link:

sentenceID	i	j	S/P

Mihalcea and Pedersen [2003] further augmented the scheme by adding another field for the confidence score of the link:

sentenceID	i	j	S/P	score

This scheme is very popular. We will, in the remainder of the thesis, refer to this scheme as the "standard annotation scheme" for manual word alignment. It is simple yet expressive. One can convert any kind of alignment annotation into this format. The two tags, S and P, roughly meet the need to capture the difference between unambiguous and ambiguous links. Finally, Och and Ney [2000] designed a word alignment evaluation metric, the Alignment Error Rate (AER), which relies on this annotation scheme. AER has become a very widely-used intrinsic evaluation metric in the MT community. For all these reasons, this annotation scheme for word alignment has been used in many subsequent studies, although modifications are sometimes made. For instance, Graça et al. [2008] used the same annotation scheme, but with a different interpretation. They used S to tag a link when the concerned pair of words constitute a translation in every context, and used P to tag pairs where a translation is possible in certain contexts. With this definition, Graça et al. [2008] narrowed the usage of the S tag to very sure, context-independent cases.

Li et al. [2009] use a matrix-based scheme for word alignment. The word alignments of one sentence pair are represented as one matrix $\mathbf{M}$. The rows of $\mathbf{M}$ contain the indices of target tokens, and the columns contain the indices of source tokens. The value of one

[8] Anyway, enforcing such limitations in the representation may be unnecessary, since directional alignments are typically made symmetric in downstream components of application systems.

[9] To avoid confusion, we will use the term "P-tagged word pair" to refer to a P-link that is not an S-link.

entry (i, j) of $\mathbf{M}$ can be 0, 1 or 2. The first row (resp. column) $\mathbf{M}[i, 0]$ (resp. $\mathbf{M}[0, j]$) represents whether or not the target (resp. source) tokens are translated. For the first row and the first column, the value 0 means the token has alignments, 1 means the token has no explicit correspondence but the meaning is conveyed in the translation, 2 means the token has no explicit correspondence and the meaning is missing/inserted. The other entries are treated differently. The value of an entry $\mathbf{M}[i, j]$ with $i \neq 0, j \neq 0$ indicates the relation between the i-th target and the j-th source tokens. For these entries ($i \neq 0, j \neq 0$), the value 0 means the two tokens are not aligned, 1 means the two tokens are correspondent and the translation is correct, 2 means they are correspondent but the translation is incorrect. Hence, Li et al. [2009] also impose a classification of alignment links, although their criteria are quite different from that of Och and Ney [2000].

The Yawat word alignment tool of Germann [2008] uses a representation scheme in which the basic alignment units are *word groups*, similar to that of Li et al. [2010b]. This is different from the scheme of Och and Ney [2000], where the basic units are word pairs. In [Germann, 2008; Li et al., 2010b], the alignment links of one sentence pair are all stored in one line of the alignment file. One alignment link is represented by a sequence of source indices and a sequence of target indices, potentially with tags. This is illustrated in the example below.

English:	`I want to eat .`
French:	`Je veux manger .`
Alignment:	1:1:S 2,3:2:P 4:3:S 5:4:S

4.2.2 Evaluation Metrics

Precision, Recall and F-score As in sentence alignment, we can use Precision (P), Recall (R) and F-score (F) as the metrics for evaluating an automatic word alignment against a reference alignment, especially when links are not distinguished by tags such as S and P. Melamed [1998a] used these metrics in the Blinker project.

Suppose the set of reference alignment links is $\mathcal{L}$, and the set of the computed alignment links is $\mathcal{A}$, then the definition of P, R and F is:

$$P = \frac{|\mathcal{A} \cap \mathcal{L}|}{|\mathcal{A}|} \qquad R = \frac{|\mathcal{A} \cap \mathcal{L}|}{|\mathcal{L}|} \qquad F = \frac{2PR}{P + R}$$

By using these metrics, we are treating all word alignment links equally. But, since word alignment is ambiguous, some reference links are actually more reliable than others, and errors on more reliable ones should be penalized more severely by the evaluation metric. Precision, recall and F-score ignore this difference, which is somewhat inappropriate.

Refined Precision, Recall and F-score If both $\mathcal{L}$ and $\mathcal{A}$ are tagged using the S/P scheme, we can compute a refined version of P, R and F. Note the set of S-tagged pairs in $\mathcal{L}$ (reps. $\mathcal{A}$) as $\mathcal{L}_s$ (resp. $\mathcal{A}_s$), the set of P-tagged pairs in $\mathcal{L}$ (reps. $\mathcal{A}$) as $\mathcal{L}_p$ (resp. $\mathcal{A}_p$), then we compute P, R and F once for S-links and once for P-links:

$$P_s = \frac{|\mathcal{A}_s \cap \mathcal{L}_s|}{|\mathcal{A}_s|} \qquad R_s = \frac{|\mathcal{A}_s \cap \mathcal{L}_s|}{|\mathcal{L}_s|} \qquad F_s = \frac{2P_sR_s}{P_s + R_s}$$
$$P_p = \frac{|\mathcal{A}_p \cap \mathcal{L}_p|}{|\mathcal{A}_p|} \qquad R_p = \frac{|\mathcal{A}_p \cap \mathcal{L}_p|}{|\mathcal{L}_p|} \qquad F_p = \frac{2P_pR_p}{P_p + R_p}$$

AER The Alignment Error Rate metric goes hand in hand with the standard annotation scheme presented in § 4.2.1. Suppose two annotators annotate the same bitext with this scheme. The first annotator produces a set of S-links $\mathcal{S}_1$ and a set of P-links $\mathcal{P}_1$, the second produces $\mathcal{S}_2$ and $\mathcal{P}_2$. We generate two reference alignment sets: $\mathcal{S} = \mathcal{S}_1 \cap \mathcal{S}_2$ and $\mathcal{P} = \mathcal{P}_1 \cup \mathcal{P}_2$. The AER of a computed alignment $\mathcal{A}$ is:

$$AER(\mathcal{S}, \mathcal{P}; \mathcal{A}) = 1 - \frac{|\mathcal{A} \cap \mathcal{S}| + |\mathcal{A} \cap \mathcal{P}|}{|\mathcal{A}| + |\mathcal{S}|}$$

Note AER can also be used in cases of more than two annotators, by taking the intersection of S-links in all annotations to form $\mathcal{S}$, and taking the union of all P-links in all annotations to form $\mathcal{P}$. $\mathcal{S}$ contains links that all annotators consider as being unambiguous, thus more likely an obvious good link. $\mathcal{P}$ contains links that at least one annotator considers as being possible. We expect $\mathcal{S}$ to be highly precise, and $\mathcal{P}$ to give a wide coverage of potentially useful links.

We can derive the following properties of AER [Och and Ney, 2000]:

1. $0 \leq AER \leq 1$, since $|\mathcal{A} \cap \mathcal{P}| \leq |\mathcal{A}|$ and $|\mathcal{A} \cap \mathcal{S}| \leq |\mathcal{S}|$;
2. if $\mathcal{S} \subseteq \mathcal{A} \subseteq \mathcal{P}$, then $\mathcal{A} \cap \mathcal{S} = \mathcal{S}$ and $\mathcal{A} \cap \mathcal{P} = \mathcal{A}$, leading to $AER = 0$.

Another appealing characteristic of the AER is that it emphasizes the importance for the computed alignment $\mathcal{A}$ of being correct for links in $\mathcal{S}$: if we change $\mathcal{A}$ to $\mathcal{A}'$ by replacing one $\mathcal{S}$ link in $\mathcal{A}$ by one P-tagged pair, then $|\mathcal{A} \cap \mathcal{S}| + |\mathcal{A} \cap \mathcal{P}| - |\mathcal{A}' \cap \mathcal{S}| - |\mathcal{A}' \cap \mathcal{P}| = 1$, thus $\mathcal{A}$ would have a better AER score than $\mathcal{A}'$. So, AER gives more weight to more reliable reference links, which is a desired feature.

The AER is sensitive about the ratio between the S and P tags. Lambert et al. [2005] argued that, when using the annotation scheme of Och and Ney [2000] to create reference alignments and using the AER as the metric, the ratio between S- and P- tagged word pairs impacts two aspects of the evaluation: first, reference alignments with a high proportion of P-tagged pairs would favor high precision automatic alignments, while reference sets with a high proportion of S-tagged pairs would favor high recall automatic alignments; second, if the reference alignment set contains a large proportion of P-tagged pairs, very different

automatic alignments can achieve the same AER score,[10] i.e. large proportion of P-tagged pairs in the reference set would reduce the ability of the metric to discriminate between alternative alignment techniques. Lambert et al. [2005] concluded that reference alignments with a large proportion of S-tagged pairs are preferred for MT. The correlation between the AER and translation quality metrics such as the BLEU has been studied in [Fraser and Marcu, 2007; Ganchev et al., 2008]. Nevertheless, the AER has become a very widely-used intrinsic evaluation metric in the MT community (e.g. [Och and Ney, 2003; Liang et al., 2006; Cromières and Kurohashi, 2009; Dyer et al., 2013a; Liu et al., 2015], to name a few).

Null links The treatments of null links vary across studies. It is common practice to P-tag null links [Och and Ney, 2000; Lambert et al., 2005]. However, in some works null links are counted in the evaluation, in others they are not. As Lambert et al. [2005] pointed out, evaluation results are greatly affected by the treatment of null links. One should always be clear on this issue when reporting/analyzing word alignment evaluation results.

Link weights Melamed [1998a] discussed the weighting of reference alignment links from another perspective. Melamed [1998a] argues that, since one word can be linked to any number of words of the other side, an evaluation metric that treats all links equally would place too much importance on words that are included in more than one link. He proposed to attach the following weight to a link (i, j) (where i and j are word indices from different sides):

$$w(i,j) = \frac{1}{\texttt{max}(\texttt{out}(i), \texttt{out}(j))}$$

where the function $\texttt{out}(i)$ gives the number of links that include the word at index i, and the function $\texttt{max}()$ returns its largest argument. Melamed [1998a] used this weighting scheme to compute generalized versions of P, R and F. It can also be incorporated into other metrics. Lambert et al. [2005] pointed out that if this weighting method is used, then it must be applied for both reference and computed links. Nonetheless, this weighting scheme does not seem widely accepted.

4.2.3 The Creation of Reference Word Alignments

Preprocessing The setup of a manual word alignment annotation process consists of the following preprocessing steps:

– Bitexts: Generally the bitext is composed of a set of parallel sentences, extracted from some parallel corpus. English is often one of the involved languages. The Bible (e.g. in [Melamed, 1998a]), the Europarl (in [Graça et al., 2008]) and the Hansard (in [Och and Ney, 2000]) are popular sources of parallel sentences. Li et al. [2009] explore

[10] With this kind of reference sets, a computed set $\mathcal{A}$ has better chance to get $\mathcal{S} \subseteq \mathcal{A} \subseteq P$, thus a zero AER.

several text genres: newswire, broadcast news, broadcast conversation, newsgroups ans weblogs.

- Tokenization: The tokenization step is important for making annotations useful. Manual alignments and automatic alignments should stick to the same tokenization. As well as for sentence segmentation, tokenization rules should be made explicit and reproducible, if the reference is to be used for evaluation purposes.

- Guidelines: Manually annotating word alignment links requires to deal with many ambiguous cases, which are often language-dependent. To ensure the consistency of annotations, the annotators are generally provided with precise annotation guidelines. To the best of our knowledge, [Melamed, 1998b] was the first published guidelines for bilingual word alignment annotation (for English-French). Subsequent works often contain their own guidelines, adapted to specific language pairs. Typically, the draft version of the guideline is used to annotate a sample of sentence pairs, then it is revised according to the feedbacks and suggestions of the annotators. Finally, the revised version is applied to the annotation.

Manual alignment tools As for the software to perform the annotation, there are many choices. In all cases, the annotators are provided with a graphical user interface (GUI). We list a few below:

- Blinker: Melamed [1998a] developed his own annotation software (also called Blinker), in which a word alignment link is represented by drawing a line between the two words. This tool does not support the distinction between S- and P-links.

- Uplug: The Uplug toolkit [Tiedemann, 2003b] contains a web-based tool to interactively modify word alignment results of an automatic word aligner. It uses a matrix representation for word alignments. Annotators can add or remove a one-to-one word alignment link each time. Uplug is available at `https://bitbucket.org/tiedemann/uplug/wiki/Home`.

- Alpaco: The Alpaco toolkit is a manual word alignment tool written in Perl, whose functionalities are similar to Blinker, with additional coloring schemes. This toolkit is at `http://www.d.umn.edu/~tpederse/parallel.html`.

- UMIACS: The UMIACS word alignment tool is written in Java and has similar functionalities with Blinker. It it available at `http://www.umiacs.umd.edu/~nmadnani/alignment/forclip.htm`

- HandAlign: HandAlign is a Java-based tool. Importantly, it supports one-to-many links and phrase-level alignments. This is different from Alpaco and UMIACS, where only one-to-one links are allowed. Alignment links are again represented by lines

between the two sides. HandAlign distinguishes sure links from possible links by different link colors. It is available at `http://www.umiacs.umd.edu/~hal/HandAlign/index.html`.

- AltAligner: The AltAligner [Gao and Vogel, 2010] is a word alignment interface on top of Google Web Toolkit (GWT). This tool is web-based, hence it can be deployed on the Internet, and allows alignment annotations at distance. AltAligner also represents links by drawing lines between words. It is available at `https://code.google.com/archive/p/alt-aligner/`.

- Yawat: In Yawat [Germann, 2008], a link represents the relation between *groups of words*. A word in one link cannot be further linked to words outside the group. Links are no longer represented by lines. Germann [2008] uses both a highlighting scheme and the matrix representation for links. Yawat supports link tagging with pre-defined or user-customized labels. It is also web-based, so it can be deployed on the Internet. The Yawat package can be obtained by request from its developer.

These are only representative tools that allow manual alignments. We have not included tools only supporting word alignment visualization, such as Cairo [Smith and Jahr, 2000]. Importantly, the software functionality has a direct impact on resulting annotations. In order to obtain annotations in the desired form, we must take care to choose the right tool.

Merging annotations Despite the existence of guidelines, annotator disagreements arise frequently in the annotation process. There are multiple ways to merge annotations of various annotators. We list some typical solutions:

- the final reference alignment set only contains links on which all annotators agree (the intersection);

- alternatively, the reference set contains the union of all annotations;

- alternatively, the reference set first contains the intersection of annotations. Then, all disagreements are processed, typically by a vote of annotators, or by an adjudication process. The final decisions are added into the reference set.

The first heuristic emphasizes the importance of very sure links. The second approach maximally captures potentially useful ones. The third one can be viewed as a trade-off. [Melamed, 1998a] and the Romanian-English part of [Mihalcea and Pedersen, 2003] have adopted the last methodology to resolve ambiguities.

High-level rules Among the various annotations rules, some are more general and are language-independent. These rules typically concern the high-level annotation style.

– *Semantic omissions*: by this term we refer to the situation where a piece of information on one side is lost on the other side. This is to be distinguished from small omissions due to grammatical reasons. For semantic omissions, the words involved are all linked to NULL. In the example in Figure 4.3, if we delete the group of English tokens "And he said ,", the two sentences would become more parallel. Thus all these four tokens should be aligned to NULL.

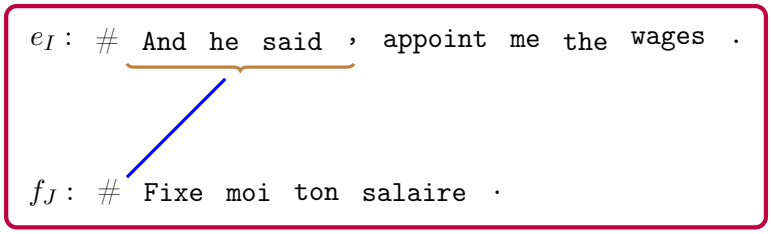


Figure 4.3: An example of the rule related to semantic omission.

– *Maximal decomposition*: the lexical units involved in a link should be as small as possible, but contain as many words as necessary. In other words, a link is not acceptable if it can be decomposed into two valid links. In Figrue 4.4, the French expression "ronger son frein" is translated as "champ at the bit". Since we can establish a fine grain correspondence between the two expressions at the word level, it is not correct to just link the two expressions all together.

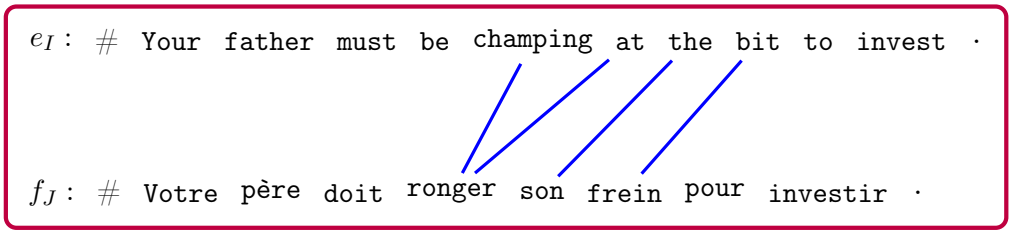


Figure 4.4: An example of the rule of maximal decomposition.

– *Phrasal correspondences*: if a source word group and a target word group express the same meaning, but cannot be decomposed into smaller links, then they should be linked as a whole. In Figure 4.5, the French expression "ronger son frein" is translated as "seal my lips'. Unlike the preceding example, here no word-to-word correspondence exists. We have to link the two expressions as a whole.

In the standard annotation scheme of § 4.2.1, phrasal correspondences are represented by taking all possible word level links, that is, the Cartesian product of the two sides.

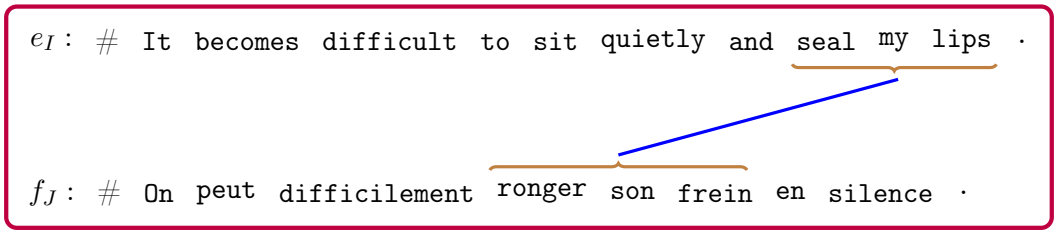


Figure 4.5: An example of the rule of phrasal correspondence.

Usually these word-level links are P-tagged, except for those word links where the two words constitute a good word-level correspondence, which might be S-tagged.

– *Contextual correspondences*: sometimes the source and target sides are semantically equivalent only in the specific context. In most projects, these word pairs are marked as a P-link, as illustrated in Figure 4.6. But in some others (e.g. [Graça et al., 2008]), such word pairs are left unaligned.

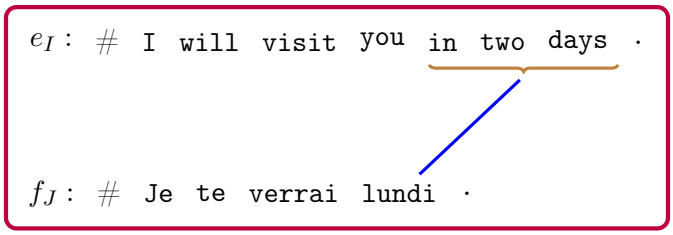
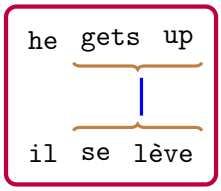


Figure 4.6: An example of contextual correspondences.

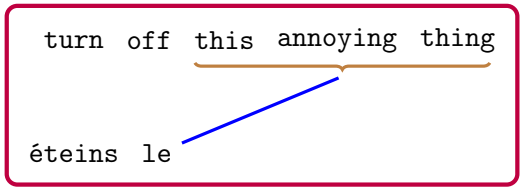
Rules of this kind are observed in many studies. Nonetheless, the way to handle these cases ultimately depends on the goal of the annotation task.

Common grammatically ambiguous cases Some recurring ambiguous cases in manual word alignment are caused by surface structure changes across languages, typically due to syntactic divergences. Mainly following [Melamed, 1998b], we list several patterns that seem to arise in several language pairs, as well as typical ways of handling them:

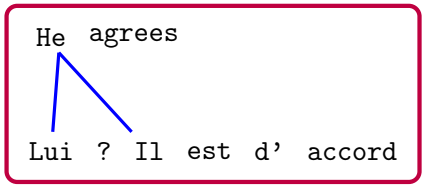
– idioms and near idioms are linked as a whole with their correspondences



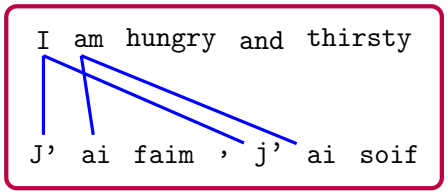
– descriptions of the same thing are linked as wholes;



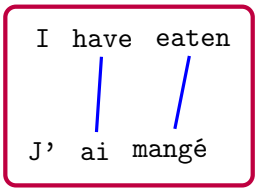
– resumptive pronouns, if not translated, are linked to the translation of their antecedents;

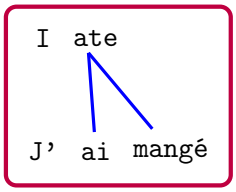


– repetitions, if not all translated, are all linked to the unique counterpart;

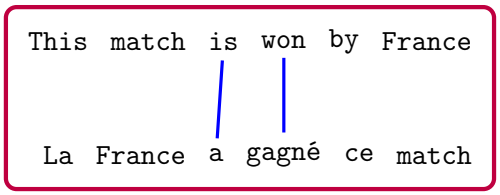


– an auxiliary verb is linked to the auxiliary verb on the other side, if it is present; otherwise, it should be linked to the main verb on the other side;

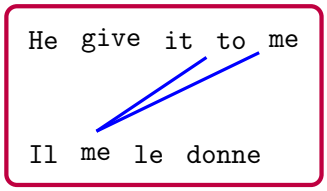




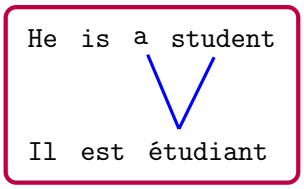
– upon change of voices, if both sides have auxiliary and main verbs, then they are separately linked to their counterparts; otherwise, the verb structures are linked as wholes;



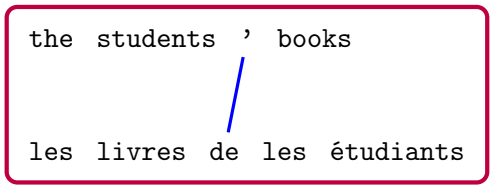
– an extra preposition is linked to the translation of its object, not to its subject;



– an extra determiner is linked to the translation of the head noun;



– possessive markers are linked separately from the nouns, even when the marker is just an apostrophe;



This list is by no means exhaustive. We have not distinguished the assignment of link tags such as S/P, which varies a lot from projects to projects. The bottom line is, in a word alignment annotation task, the preparation of the guidelines often requires careful analyses of the languages involved and of the aims of the task.

Evaluation of gold alignments Unlike sentence alignment, where we can achieve agreements on (almost) all links, manual word alignment data sets often contain divergent annotations. As other annotation tasks which involve subjective judgments, the *inter-annotator agreement rate* (IAA) is often computed to measure the reliability of annotations. A commonly used metric for categorical annotation tasks is Cohen's kappa coefficient [Cohen, 1960], where observed agreement counts are balanced by chance agreement counts.

However, Cohen's kappa is not a proper inter-annotator agreement rate for manual word alignment [Graça et al., 2008]. To use Cohen's kappa, we need to cast word alignment into a categorical annotation task, which consists of viewing each word pair as an annotation instance. For a source sentence with I words and a target one with J words, there are $I \times J$ such instances. There are at least two candidate categories for each instance: *true* if the annotator deems there is a link between the pair of words, *false* if not.[11] Approximately linearly many pairs would be linked. That is, linearly many instances would be tagged as true, while quadratically many instances as false. Among the quadratically many false instances, most of them would be tagged the same by all annotators, since they are clearly wrong. This leads to a dominating number of observed agreements in the computation of Cohen's kappa. In consequence, Cohen's kappa gives overly optimistic estimations for word alignment annotation.

The reason why Cohen's kappa is not a good metric for this task is that it treats all instances equally, while the large majority of instances are not informative. To correct this effect, we can ignore the instances (i.e. word pairs) that no annotator labels as true. Melamed [1998a] proposed the following inter-annotator agreement rate:

$$\texttt{IAA}(\mathcal{X}, \mathcal{Y}) = \frac{2 * |\mathcal{X} \cap \mathcal{Y}|}{|\mathcal{X}| + |\mathcal{Y}|}$$

where $\mathcal{X}$ is the set of alignment links annotated by one annotator, $\mathcal{Y}$ is the set of another annotator. The form of this metric is similar to Dice's coefficient. The metric is widely used in manual word alignment tasks. Melamed [1998a] also proposed to weight individual links, using the method discussed in § 4.2.2 and some extensions. Graça et al. [2008] further incorporated the link types (S and P) into the IAA metric, by varying the importance of different types of disagreements.

Note, however, that a low agreement score can be caused by several reasons:

– the guidelines are not sufficiently detailed;

– the annotators do not make consistent annotations;

– the bitext contains a large amount of ambiguous cases.

[11] If we distinguish link types, as in the S/P scheme, we will further increase the number of categories, but the false category is always present.

A manual, case-by-case analysis is often necessary to investigate agreement ratios.

Only after all these details are settled can the manual alignment process begin. There exist many published manual word alignment resources. We list and discuss some datasets in Appendix B. Our discussion, of course, cannot be exhaustive. We only present those that we are aware of and were able to locate. There must be other valuable resources. We will keep searching.

4.3 Conclusions

In this chapter, we have briefly reviewed resources for evaluating sentential and subsentential bitext alignments. For both levels, we discussed three types of resources: annotation schemes, evaluation metrics, and gold data sets.

For sentence alignment, the task of creating manual alignments is well defined, and it is possible to achieve a very high inter-annotator agreement ratio in practice. However, most existing resources are based on relatively easy types of bitexts, which might not be sufficiently challenging for automatic aligners. We will present, in Chapter 5, our contributions on manual word alignments of literary bitexts, which, we hope, will enrich the set of resources for sentence alignment evaluation.

For word alignment, the task is much more complicated. Detailed, task-dependent guidelines are often required. The majority of presented studies focus on close language-pairs, except [Mihalcea and Pedersen, 2003; Li et al., 2009]. For some European languages, the resources are relatively rich, and several annotation guidelines have been proposed. Yet, in each task, researchers make a lot of effort to achieve a satisfactory inter-annotator agreement ratio. For other language pairs, e.g. across European and Asian languages, manual word alignment should be more difficult, and available resources are scarce. Works in this direction would be very valuable.

In Chapter 5, we will present two datasets of manual word alignments, one for $1:1$ alignment links, the other one being an innovative way to produce many-to-many links, which is an easier alternative for generating such links that are usually difficult to produce reliably.

Part II

Contributions

Chapter 5

Novel Annotation Schemes for Bitext Alignment

Annotated reference alignment datasets are valuable resources for the development of alignment techniques. On the one hand, they can be used as supervision examples for supervised methods [Blunsom and Cohn, 2006; Mújdricza-Maydt et al., 2013]; on the other hand, they provide ways to directly evaluate automatic alignment quality, and warrant the investigation of error patterns. However, constructing manually annotated alignment datasets can be challenging. For some tasks, this can be due to a lack of a clear annotation scheme. For others, annotation protocols can vary a lot, depending on the targeted applications, language pairs, etc.

Existing resources for evaluating sentence-level and word-level alignments, reviewed in Chapter 4, can still be improved in various ways. Regarding sentence alignments, the existing data is too scarce, especially when it comes to difficult bitexts, containing instances of non-literal translations. Regarding word-level alignments, most available hand-aligned data provide a complete annotation at the level of words that is difficult to exploit, for lack of clear semantics of alignment links.

In this chapter, we propose new methodologies for collecting human judgments on alignment links, which have been used to annotate four new datasets, at the sentence and at the word level. In particular, our annotation protocol has been designed to promote studies of confidence estimation of alignment links. These data are released online at `https://transread.limsi.fr/resources.html`, in order to enrich publicly-available resources for evaluating alignment software and confidence estimation tools for automatic alignment.

This chapter is organized as follows. We describe our contribution to manual sentence-level alignment annotations in Section 5.1, followed by word-level alignment annotations in Section 5.2. For sentence alignment, the research community has reached a consensus on the annotation scheme [Tiedemann, 2011]. But the resource is quite scarce for certain types

of bitexts. We report, in § 5.1.1, our collection of manual sentence alignments for literary bitexts, a challenging usecase for alignment techniques. Next, in § 5.1.2, we propose a new scheme for annotating parallel fragments, which has been used to label datasets of possible parallel sentences. These resources might prove useful for tasks such as confidence estimation, or for filtering incorrect pairs in a translation memory. Regarding word alignments, our view is to consider one-to-one and many-to-many links separately. We present a novel set of annotation labels for one-to-one links in § 5.2.2, and a collection of annotations using these tags. We then describe an innovative methodology for collecting many-to-many word alignment links in § 5.2.3, as well as the corresponding dataset.

All the datasets described in this chapter, except the first one, were created by three annotators pursuing a Master level degree in translation studies, who were retributed for this work. Two of them are native French speakers with advanced capacities in English and Spanish. The other annotator is a native Greek speaker, fluent in English and French. For each task, the annotators were given guidelines, and applied them to annotate a small amount of sandbox instances (which are not included in the final datasets). Potential ambiguities regarding the task and the guidelines were then discussed and resolved, in order to a) ensure a shared understanding of the principles and details, and b) if necessary, improve the guidelines. In a second step, the actual datasets were annotated.

Apart from annotation schemes and datasets, another contribution for bitext alignment data collection is a formalism for representing alignments (and other information related to electronic bitext reading). In particular, this format supports bitexts encoded in the EPUB format[1], a de facto standard for electronic books, making it possible to preserve all editorial (header, footer), typographic (bold, italic, font sizes and shapes) and dispositional information. This XML-based format is inspired by the XCES format,[2] already proposed in the early 2000s to represent alignments. It is generic enough to represent alignment links at various levels of granularity, as well as other arbitrary information, such as POS tags of words. A full description of this format is in Appendix D.

5.1 Sentence Alignment

5.1.1 Reference Alignments for Literary Works

The ARCADE evaluation campaigns [Véronis and Langlais, 2000; Chiao et al., 2006] have demonstrated that the quality of sentence aligners is variable, depending on the bitext genres and languages. For certain types of bitexts which are relatively regular, such as institutional bitexts, the task is easy and all systems tend to deliver good results (the basic system of Brown et al. [1991] already obtained above 95% precision on the Hansards). On the contrary, for literary bitexts, alignment quality could be much less satisfactory. Yu et al. [2012b] and Lamraoui and Langlais [2013] reported that the best link-level F-score obtained

[1] `http://idpf.org/epub`
[2] `http://www.xces.org/`

Book	Lang. pair	# Link	# Src.	# Trg.
Du Côté de chez Swann (M. Proust)	EN-FR	463	495	492
Emma (J. Austen)	EN-FR	164	216	160
Jane Eyre (C. Brontë)	EN-FR	174	205	229
La Faute de l'Abbé Mouret (E. Zola)	EN-FR	222	226	258
Les Confessions (J.-J Rousseau)	EN-FR	213	236	326
Les Travailleurs de la Mer (V. Hugo)	EN-FR	359	389	405
The Last of the Mohicans (F. Cooper)	EN-FR	197	205	232
* Alice's Adventures in Wonderland (L. Carroll)	EN-FR	746	836	941
* Candide (Voltaire)	EN-FR	1,230	1,524	1,346
* Hound of the Baskervilles (A. Conan Doyle)	EN-FR	822	862	893
* Vingt Mille Lieues sous les Mers (J. Verne)	EN-FR	778	820	781
* Voyage au Centre de la Terre (J. Verne)	EN-FR	714	821	754
* Candide (Voltaire)	EN-EL	1,247	1,524	1,585
* Candide (Voltaire)	EN-ES	1,113	1,524	1,196
Total 14 books		8,442	9,883	9,598

Table 5.1: Statistics of reference sentence alignments for literary works. All "Src" entries refer to English, and "Trg" refer to other languages. "# Src" and "# Trg" are numbers of sentences. Alignments marked with a * are refinements of A. Farkas' initial alignments. The others are revised version of the data presented in [Yu et al., 2012b].

for "De la Terre à La Lune" (J. Verne), a part of the BAF corpus [Simard, 1998], was only around 78%. Hence, literary bitexts, which typically include larger portions of non literal translations, would be very useful to evaluate the actual performance of state-of-the-art alignment systems. To our knowledge, however, there are few publicly available reference sentence alignments for literary works, the most used being the BAF corpus. The need for gold alignments for such materials has also been pointed out in [Yu et al., 2012b; Lamraoui and Langlais, 2013].

To alleviate this scarce resource problem, we have collected manual alignments for a small set of literary works. Our annotators have processed excerpts from 12 classical books for French-English. Smaller Greek-English and Spanish-English corpora have also been collected, notably resulting in a multiple sentence alignment of Voltaire's "Candide". The annotation was performed using the Uplug toolkit [Tiedemann, 2003b]. In order to make our annotations more suited to evaluate automatic alignment tools, the annotators have made sure that our manual alignments actually follow the conventions listed in § 2.1.1 (minimality, monotonicity, prohibition of gappy alignments), which are the most important annotation guidelines.

Table 5.1 summarizes the main statistics of the corpus. Note that for the books with

the * mark, alignment links were generated as refinements of existing reference paragraph alignments provided by A. Farkas.[3] The numbers of sentences of "Candide" are larger than other books, because for this book we have aligned the entire texts, while for others we have aligned typically several beginning chapters. Further exploitations of these sentence alignments will be presented in Chapter 6 and Chapter 7.

We do not report agreement numbers here because the task is relatively easy and well understood. In a sandbox experiment, the agreement rate between the annotators is as high as 99.8%.

5.1.2 Confidence in Sentence Alignment

Automatic alignments are mostly used for statistical machine translation [Koehn, 2005]. In this context, it is customary to filter out unreliable alignment links based on heuristic confidence measures, such as length ratios. Confidence estimation can also prove useful in other contexts, for instance in bilingual concordancers [Simard et al., 1993b; Bourdaillet et al., 2009], for translator training or in other language learning scenarios. This is even more necessary when alignments are extracted from noisy bitexts, e.g. bitexts collected from the internet [Tiedemann, 2011], or for crowd-sourced alignments.

Confidence Estimation (CE) for sentence alignments aims at evaluating the usability of alignment links. This is different from quality estimation for machine translation, where the quality of system outputs as valid sentences is not certain and plays an important role. In CE for sentence alignment, all sentences are deemed to be well formed, and the only thing that needs to be evaluated is the level of correspondence between the two sides of a link. However, the usability of a link depends on the targeted application. The canonical sentence alignment evaluation metric, the F-measure, distinguishes two classes (correct and wrong). The recently introduced task of translation memory (TM) checking considers three cases:[4] a pair of segments can be a) totally correct and need no editing at all, or b) need substantial editing, or c) can be mostly correct but need few simple edits. Similar categories have been used to assess the usefulness of automatic translations for post-edition [Wisniewski et al., 2013]. Note finally that for SMT, even partially correct alignments and very loose mutual translations can be useful as training data.

To better reflect this flexible notion of alignment link quality, we propose a new annotation scheme based on a 5-way categorization of sentence alignment links:

1. *sure*: the pair of sentences are (near) perfect mutual translations;
2. *partial*: one side contains some parts that are not translated on the other side.
3. *imperfect*: the pair constitutes a loose complete mutual translation, or a translation only in specific contexts;

[3] `http://FarkasTranslations.com`

[4] This scheme is used for the shared task on cleaning translation memories of the NLP4TM'16 workshop. See `http://rgcl.wlv.ac.uk/nlp4tm2016/shared-task/`.

4. *erroneous*: the pair has no correspondence relation at all (i.e. the pair is not correct);

5. *undecidable*: none of the previous four cases, which corresponds to cases where the two sides of a link are highly context-dependent and cannot be annotated in isolation. In practice, this class is quite rarely used.

Note that upon choosing the label "partial", our annotators were also asked to mark explicitly the untranslated part (see Figure 5.1).

We took the automatically sentence-aligned English-French OpenSubtitle Corpus,[5] randomly picked 1,800 alignment links, and used the proposed tags to annotate them. Each alignment link was annotated twice. The first annotator annotated links 3000 $\sim$ 4199 (link ID range), the second annotated 3600 $\sim$ 4799, and the third annotated 4200 $\sim$ 4799 and 3000 $\sim$ 3599. We used an adapted version of the Yawat tool [Germann, 2008] to perform this task. Figure 5.1 displays the annotation interface. We found that the inter-annotator agreement for this task was very high (the average $\kappa \approx 0.85$), showing that our annotation scheme was sensible. Among the 1,663 links that the annotators agreed on the labels, 1,002 were tagged as "sure" (62.25%), 252 "partial" (15.15%), 163 "imperfect" (9.80%), 244 "erroneous" (14.67%), and 2 "undecidable" (0.12%). This dataset can be used, for example, to train confidence estimation tools for sentence alignment links, or in translation memory cleaning tasks.

Figure 5.1: Sentence alignment confidence annotation. For each alignment link, the color of the special symbol $$$ encodes its label: green for "sure", violet for "partial", etc. Note the untranslated part of the pair 3003 (labelled "partial") appears in gray. Further note that the pair 3003 is in fact a wrong 1:2 link, but not a case of sentence segmentation problem.

[5] Downloadable from `http://opus.lingfil.uu.se/`.

5.2 Collecting Sub-sentential Alignments: Two New Proposals

5.2.1 Evaluating Word Alignments with Gold References

Bilingual word alignments constitute an important resource for many downstream applications in multilingual NLP. Some rely on 1:1 alignment links, e.g. in cross-lingual transfer of Part-of-Speech labels [Täckström et al., 2013; Wisniewski et al., 2014] or of other kinds of information; others use many-to-many alignments, e.g. phrase-based SMT [Koehn et al., 2003]. Most applications perform better when alignment quality is improved [Lambert et al., 2005]. Because word alignment is both important and challenging, it has received a sustained attention of the research community since the introduction of IBM Models by Brown et al. [1993]. Numerous approaches have been since proposed to improve alignment quality ([Liang et al., 2006; Dyer et al., 2011; Wang et al., 2015], to name a few).

Metrics We have extensively discussed the evaluation metrics of word alignment in Chapter 4. The evaluation of word alignments is a tricky question [Tiedemann, 2011]. The most commonly used intrinsic evaluation metric is the Alignment Error Rate (AER) proposed by Och and Ney [2000], which relies on the **S**/**P** annotation scheme. This metric and the corresponding annotation scheme have been criticized in many subsequent studies [Fraser and Marcu, 2007], notably due to the lack of clear semantics of **P**-links, which tend to be used in too many situations (non-literal translation, many-to-many alignments, etc.). Regarding extrinsic metrics, a widely used approach is to consider SMT output quality measured by automatic scores such as BLEU. As repeatedly noted [Lopez and Resnik, 2006; Fraser and Marcu, 2007; Lambert et al., 2010], AER poorly correlates with translation quality, especially for large corpora, which makes the direct comparison of alignment systems more difficult.

Building reference alignments The construction of gold word alignments is a complicated task: their specification must address deep linguistic issues (which are often specific to language pairs), but also take into account the intended use of these alignments, notwithstanding more concrete issues such as interface design and disagreement resolution procedures. Melamed [1998b] was the first to propose a complete annotation guideline for the Blinker project, which was used to align 250 verse pairs of the Bible (English-French) with a binary annotation scheme. Och and Ney [2000] used the Blinker guidelines to align 484 sentence pairs of the Hansard corpus (English-French), further introducing the Sure/Possible distinction. Please refer to § 4.2.3 for a thorough discussion on reference alignment creation.

We propose new methodologies to collect evaluation data for word alignment. Our proposal relies on two distinct protocols: the first focuses on 1:1 alignments and proposes a much clarified version of the S/P distinction (see § 5.2.2); the second specifically targets many-to-many alignments, and is based on a divisive annotation strategy which proceeds

iteratively (see § 5.2.3). For both tasks, the annotations are carried out with adapted versions of Yawat.

5.2.2 A New Annotation Scheme for 1-to-1 Alignments

One major problem with the S/P annotation scheme, which is widely-used for 1:1 alignment links, is the vagueness of this distinction, yielding annotations that are highly subjective. In [Och and Ney, 2000], it is stated that:

> a S (sure) alignment which is used for alignments which are unambiguously and a P (possible) alignment which is used for alignments which might or might not exist.

Yet, for some annotators, an unambiguous link might imply a context-independent word pair; for others, if a source word A is in the context of a particular sentence pair the best match for target word B, and vice-versa, then the link is unambiguous. Many-to-one alignments are also often difficult to annotate. Second, the vagueness of **P** links makes their systematic exploitation difficult: for instance, when a multiword expression is paraphrased, it is common practice to **P**-tag all individual word links in the corresponding block [Lambert et al., 2005; Graça et al., 2008]. This block of **P** links would be helpful for a multiword expression extractor; however, some other **P** links are made of word pairs that share the same meaning in a particular context and that would be irrelevant for such an application. Lambert et al. [2005] further pointed out that reference alignments having a large majority of **P** links would harm the usefulness of the AER metric, as automatic alignments of very different underlying quality might achieve the same AER score with respect to such a reference dataset.[6]

We hold the view that, for annotations to be maximally useful, the **S** tag should indicate word pairs that can reliably be used in any application, thus it should be reserved for word pairs that share the same meaning in most contexts (a similar semantics for the **S** tag is used in [Graça et al., 2008]). As for **P** links, we find that the majority of them fall into two categories: some are contextual, while others are part of a larger correspondence between *groups of words*. We thus propose to define the following annotation tags for 1:1 word alignment links:

- *sure*: the pair of words express the same meaning, e.g. "dog – chien";
- *contextual*: the pair of words express the same meaning only in the specific context, e.g. "tomorrow – samedi" (French for "Saturday");
- *partial*: the pair of words do not constitute a good link by themselves, but they should be included in a larger link (group of words), e.g. "(make) use (of) – (se) servir (de)";

[6] When a group of source words are aligned to a group of target words, it is customary to **P**-tag all resulting 1:1 links in the Cartesian product. Unfortunately, this can easily lead to a large number of 1:1 **P** links in the reference.

– *wrong*: the corresponding pair of words should not be aligned.

This annotation scheme has been tested using high-confidence 1:1 links produced automatically. This set of alignments was prepared as follows. For each language pair, we first combined the sentence-aligned "Candide" and the Europarl data for this language pair [Koehn, 2005] into a parallel corpus, which was word-aligned by running MGIZA [Gao and Vogel, 2008] in both directions. We then formed a small candidate corpus, by taking all sentence pairs of "Candide" and a few hundreds of the Europarl.[7] Finally, for each sentence pair in the candidate corpus, we have selected at most five 1:1 links in the intersection of the directional alignments, thereby ensuring that the potential alignment points were sensible choices. However, in order to avoid that only very sure links were given to annotators, we have decided to pick links with low probability scores from the intersection.

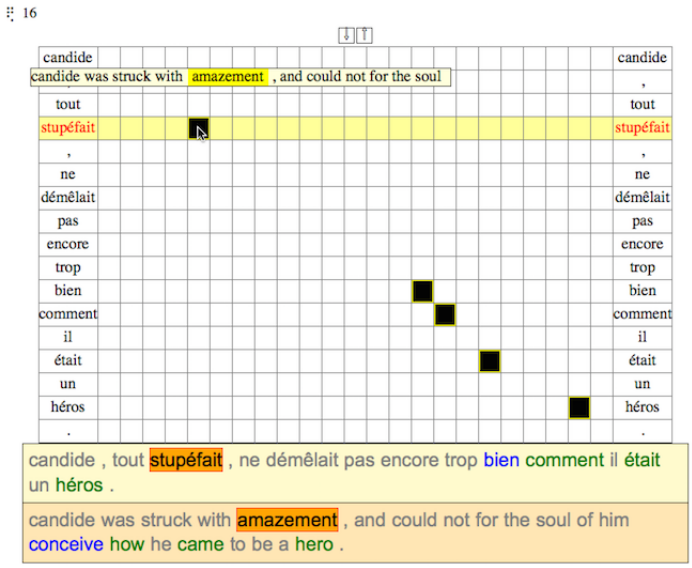


Figure 5.2: 1:1 word alignment confidence annotation for a parallel sentence from "Candide". A black cell in the alignment matrix represents a potential alignment link. For each link, the color of the word pair corresponds to its label.

Each link was then manually annotated with one of the four labels described above. Figure 5.2 illustrates the annotation process for one parallel sentence from "Candide" (French-English). Using this methodology, we were able to collect 2,691 link annotations for English-French, 3,118 for English-Spanish, 2,996 for Spanish-French, 2,204 for Greek-English, and 527 for Greek-French, totaling 11,536 word-level annotations. On the English-French subset of links that were hand-annotated more than once, the inter-annotator agreement rate is

[7] We ran MGIZA on the combined large corpus instead of just "Candide" to maximize the quality of automatic word alignments.

around 0.75. Figure 5.3 shows the distribution of labels per language pair.[8] We observe that each of the labels "partial" and "contextual", though less frequently used than "sure" and "wrong" in general, represents a non-negligible, sometimes even important, portion. This observation confirms our belief that a finer categorization than **S**ure and **P**ossible is sensible. The distribution of labels varies for each language pair. The most remarkable situation is perhaps the large proportion of links tagged as "contextual" and the relatively low portion of "sure" links in the Spanish-French data. Our study of this dataset reveals two potential reasons. First, the translation between the Spanish and French texts are not very literal. There are many cases where quite different expressions are used to express the same thing. For example, the Spanish word "párrafo" ("paragraph") was used to translate the French word "article", to indicate un passage of some law; also, the Spanish word "entrar" ("enter") was used to translate the French word "circuler" ("circulate"). Second, the annotator who has annotated the majority of Spanish-French links has applied a very strict standard for "sure" links. She has tagged, for example, the word pair "positivo" ("positive") and "bon" ("good") as "contextual". For her, words with relatively quite close meaning can still be regarded as being "contextual".

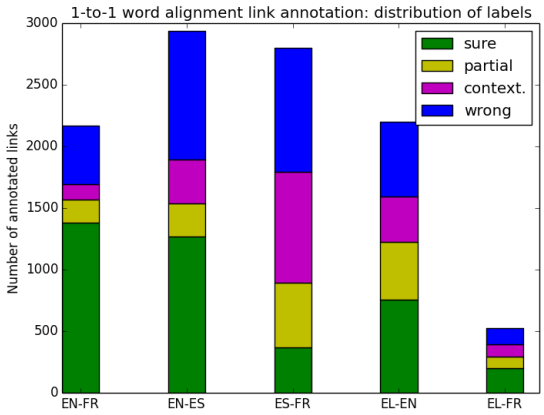


Figure 5.3: The distribution of 1-to-1 word alignment annotation labels per language pair.

5.2.3 Collecting Reference Many-to-Many Alignments

We further propose a novel method to obtain reference many-to-many alignments. The protocol is based on recursive divisions of parallel sentence pairs. Given a pair of parallel segments (we call such a pair of segments a *bi-segment*) E_1^I with I words and F_1^J with J words, the annotators iterated the following process:

[8] For some subsets annotated by more than one annotator, we have taken the intersection. So the numbers of links in Figure 5.2 are slightly different from those reported in the text.

1. If the bi-segment cannot be further divided, terminate;
2. Else, the annotators pick an index i for E, an index j for F, such that the four segments $E_1^i, E_{i+1}^I, F_1^j, F_{j+1}^J$ can form two bi-segments. One possibility is that E_1^i is parallel with F_1^j and E_{i+1}^I is parallel with F_{j+1}^J; another is that E_1^i is parallel with F_{j+1}^J and E_{i+1}^I is parallel with F_1^j (the indices i and j define *splitting points*);
3. For each bi-segment produced in step 2, go to step 1.

In essence, the idea behind this procedure is quite similar to the automatic alignment systems of Simard and Langlais [2003]; Lardilleux et al. [2012].

We believe that this protocol is much simpler than annotating the full alignment matrix, since at each step there is only one single decision to make. Two heuristics are used to guide the annotation process: (a) when many segmentation index pairs (i and j in step 2) are acceptable splitting points, choose one such that (i) the segmentation is as balanced as possible (in terms of segment lengths), (ii) the linguistic structures are preserved as much as possible; (b) the process terminates when either the bi-segment in step 1 is a 1:1 word pair, or when the segment is not strictly compositional and thus cannot be split.

Figure 5.4 illustrates the first iteration of splitting a sentence pair. Note many splitting points other than the chosen one are possible: for example, the pair of words "authority" and "autorité". But the resulting two bi-segments would be less balanced. We may even choose "on" and "en", but this would destroy the pair of expressions (though compositional) "on the subject" and "en la matière", which we prefer to keep together. Figure 5.5 displays the second iteration, during which we split the two resulting bi-segments of the first pass. It is obvious that we can continue to proceed in this way, for instance, by processing the bi-segment "the hon. member for Don Valley" and "le député de Don Valley".

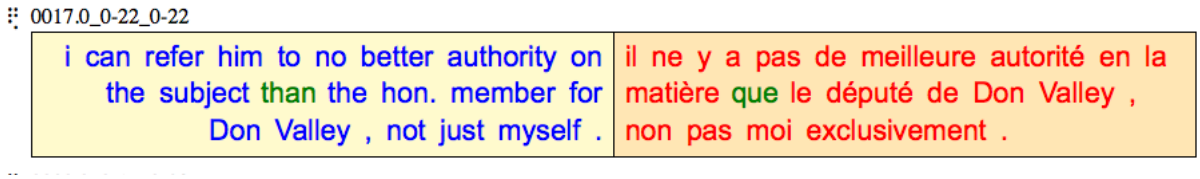


Figure 5.4: The first pass, splitting sentence pair 0017.

We have recorded all the bi-segments generated during the whole process, resulting in a hierarchical alignment structure between the original sentences. Ideally, for a parallel sentence, all annotators would arrive at the same set of final bi-segments (albeit with different choices of splitting points). This is hard to achieve in practice, since annotators might differ on the notion of linguistic structures and compositionality. Still, the bi-segment sets can help to estimate our confidence for many-to-many alignments. If a computed many-to-many alignment can be decomposed into a combination of several bi-segments, then it

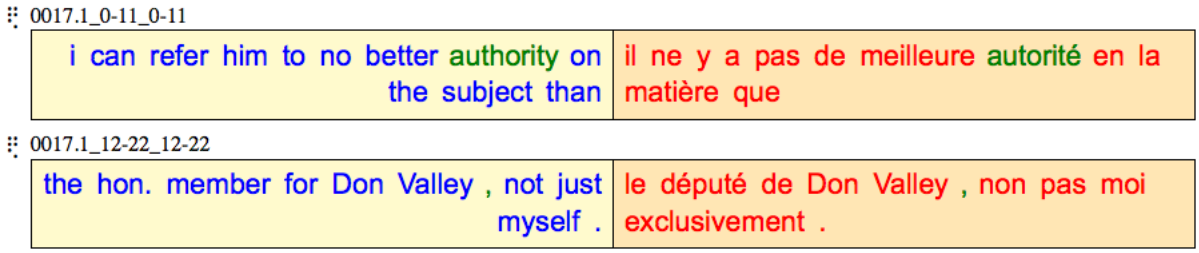


Figure 5.5: The second pass, splitting the two bi-segments of sentence pair 0017.

is reasonable to suggest that it is a good link. Furthermore, the hierarchical nature of our annotations makes it possible to design metrics for word alignments that could explicitly depend on their compatibility with our bi-segmentation.

For this dataset, the annotators were presented with 1,086 sentence pairs from Europarl, 220 from the Hansard, and 290 from Jules Verne's "Vingt Mille Lieues sous les Mers" and Sir Arthur Conan Doyle's "The Great Shadow". The final set contains approximately 10,000 bi-segments.

The 220 Hansard sentence pairs were chosen from the trial and test set of the NAACL 2003 workshop on word alignment [Mihalcea and Pedersen, 2003], where reference word alignment had been provided by Och and Ney [2000]. This subset enables us to compare our minimal bi-segments with this reference alignment. For these 220 sentences, our recursive segmentation method gave rise to 3,971 final bi-segments, which contained 2,540 (64.0%) one-to-one links, 451 (11.4%) two-to-one links, 133 (3.3%) three-to-one, and 335 (8.4%) links whose both sides had more than 2 words (many-to-many). The reference alignment of these 220 sentence pairs contains 2,720 **S** one-to-one links, among which only 37 are not included in any of our bi-segments. In other words, our bi-segmentations contain a large majority of the **S** one-to-one links of Och and Ney [2000]. We believe this partially confirms the value of our annotation scheme. Comparatively, the analysis of **P** one-to-one links is much less satisfactory, since only 4,328 (out of 9,915) **P** one-to-one links in the reference of 2003 are actually included in one of our bi-segments, demonstrating again the uncertainty of these correspondences.

5.3 Conclusions

In this chapter, we have described several datasets, all designed for the purpose of evaluating bitext alignment software with a special attention to their possible use for confidence estimation. We have analyzed the alignment annotation tasks and discussed the weaker points of existing annotation schemes. Based on this analysis, we have proposed new annotation schemes for both sentence and word level alignments. We contribute also a method for collecting reference many-to-many alignments, which, we believe, is an innovative at-

tempt for direct evaluation of this kind of alignments. The resources and corresponding annotation guidelines are publicly available.[9]

We will present the use of these datasets in follow-up chapters. In Chapter 6, we use the sentence alignments for literary bitexts as both evaluation and training corpus for two new sentence alignment methods. In Chapter 7, we employ the sentence alignment confidence annotation and word-level annotations to evaluate confidence estimation measures, e.g. based on posterior link probabilities [Huang, 2009].

Another lesson learned in this annotation process is that sentence-level and word-level alignments are quite sensitive to pre-processing, e.g. sentence segmentation, tokenization of words, etc. It might be beneficial to investigate new ways to overcome these man-made noises so as to produce gold annotations that would be less dependent on these early steps.

[9] `https://transread.limsi.fr/resources.html`.

Chapter 6

Towards Improving Sentence Alignment: State-of-the-Art and Beyond

Sentence alignment often lies at the beginning of the bitext processing pipeline. For applications that require high-confidence alignment links (at the sentential or sub-sentential level), such as bitext reading, high-quality sentence alignment forms the foundation: most existing automatic sub-sentential alignment tools assume parallel sentences as inputs. Thus, improving sentence alignment quality is an important subject in cross-lingual NLP studies.

The sentence alignment task has received much attention in the early days of word-based SMT, driven by the need to obtain large amounts of parallel sentences to train translation models. Most approaches to sentence alignment are unsupervised and share two important assumptions which help make the problem computationally tractable: (a) a restricted number of *link types* suffice to capture most alignment patterns, the most common types being 1:1 links, then 1:2 or 2:1, etc.; (b) the relative order of sentences is the same on the two sides of the bitext. From a bird's eye view, alignment techniques can be grouped into two main families: on the one hand, *length-based approaches* [Gale and Church, 1991; Brown et al., 1991] exploit the fact that the translation of a short (resp. long) sentence is short (resp. long). On the other hand, *lexical matching approaches* [Kay and Röscheisen, 1993; Chen, 1993; Simard et al., 1993a; Melamed, 1999; Ma, 2006] identify sure anchor points for the alignment using bilingual dictionaries or crude surface similarities between word forms. Length-based approaches are fast but error-prone, while lexical matching approaches seem to deliver more reliable results but at higher computational costs. The majority of the state-of-the-art approaches to the problem [Langlais, 1998; Simard and Plamondon, 1998; Moore, 2002; Varga et al., 2005; Braune and Fraser, 2010; Lamraoui and Langlais, 2013] combine both types of information. See Chapter 2 for a detailed review on sentence alignment techniques.

The difficulty of sentence alignment depends on various factors, the first one being the targeted application, and the way in which alignment links are used. In SMT, sentence alignment mostly aims at extracting parallel sentence pairs from large-scale corpora (e.g. bilingual parliament proceedings, web-crawled multilingual materials) to fuel downstream statistical processing. For such use, the alignment problem is considered to be solved: on the one hand, it is possible to discard unreliable alignments or difficult pairs (although, as pointed out by Uszkoreit et al. [2010], this might lead to a waste of training material); on the other hand, Goutte et al. [2012] showed that the translation quality of SMT (as measured by BLEU and METEOR) is robust to noise levels of $\approx 30\%$ in sentence alignments. For other applications, the situation is quite different. A requirement may be to align the full bitext, for instance in translation checking [Macklovitch, 1994a] or bilingual reading [Pillias and Cubaud, 2015; Yvon et al., 2016].

A second factor that impacts the difficulty of sentence alignment is the text type. Certain types of corpora exhibit important translational irregularities, making high precision alignment difficult. In particular, Yu et al. [2012b]; Lamraoui and Langlais [2013] showed the link-level F-score of state-of-the-art sentence aligners on bilingual fictions remains unsatisfactory. It was for instance found that the best link-level F-score obtained for "De la Terre à La Lune" (J. Verne), a subpart of the BAF corpus [Simard, 1998], was only around 78%.

In this chapter, we report our contribution on sentence alignment. We start, in Section 6.1, by re-evaluating the actual performance of state-of-the-art methods, both in terms of their precision and recall, using large collections of publicly available novels, the goal being to clarify to which extent the sentence alignment problem is solved. Section 6.2 presents analyses of state-of-the-art tools, based on which we propose two novel methods: a MaxEnt-based model (Section 6.3) and a two-dimensional Conditional Random Fields (2D CRF) (Section 6.4). Section 6.5 contains experimental results and analyses. We conclude and discuss future works in Section 6.6.

All our experiments are conducted on bilingual literary works, making good use of the first dataset presented in Chapter 5. Literary bitexts are particularly interesting for bitext alignment studies. On the one hand, as Yu et al. [2012b]; Lamraoui and Langlais [2013] have shown, state-of-the art sentence aligners find more challenges dealing with literary bitexts than other text genres. Thus they constitute good evaluation materials. On the other hand, alignments for literary bitexts have interesting social-economical impacts. In the digital era, more and more books are becoming available in electronic form. Works of fiction account for a major part of the e-book market.[1] Global economical and cultural exchanges also facilitate the dissemination of literary works, and many works of fiction nowadays target an international audience and successful books are pre-sold and translated very rapidly to reach the largest possible readership.[2] Multiple versions of e-books constitute a highly

[1] About 70 % of the top 50,000 bestselling e-books on Amazon are in the "fiction" category (source: `http://authorearnings.com/report/the-50k-report/`).

[2] For instance, J. K. Rowling's *Harry Potter* has already been translated into over 70 languages.

valuable resource for a number of uses, such as language learning [Kraif and Tutin, 2011] or translation studies. While reading a novel in the original language often helps to better appreciate its content and spirit, a non-native reader may come across many obstacles, e.g. unfamiliar jargon, complex sentences, etc. Under those circumstances, an alignment between two versions of the same book could prove very helpful [Pillias and Cubaud, 2015]: A reader, upon encountering a difficult fragment in the original language, would then be able to refer to its translation in a more familiar language.[3] Employing such alignments of literary bitexts under human eyes, however, poses new challenges for automatic alignment techniques [Yvon et al., 2016]: the entire bitext must be aligned, and alignments should be as confident as possible. Sentence alignment is a good starting point of bitext alignment research in this direction.

6.1 Aligning Literary Texts: Solved or Unsolved ?

Commenting the unsatisfactory results achieved by all sentence alignment systems during the Arcade evaluation campaign [Véronis and Langlais, 2000] on the single test book (Jules Verne's *De la terre à la lune*), Langlais et al. [1998a] hint that:

> these poor results are linked to the literary nature of the corpus, where translation is freer and more interpretative,

expressing a general feeling that literary texts should be more difficult to align than, say, technical documents. However, assessing the real difficulty of the task is in itself challenging, for lack of a large set of books annotated with a reference (gold) alignment: for instance, the recent study of Mújdricza-Maydt et al. [2013] on English-German alignment used only three books for evaluation. In this section, we aim to provide a more precise answer to this question, using a large collection of *partially aligned* books in two language pairs.

6.1.1 The State-of-the-art

To evaluate state-of-the-art performance, we first need to identify baseline tools, appropriate evaluation metrics and representative test sets.

Baseline tools

Baseline alignments are computed using several open-source sentence alignment packages, described respectively in [Melamed, 1999][4] (GMA), in [Moore, 2002][5] (BMA), in [Varga et al., 2005][6] (Hunalign), in [Braune and Fraser, 2010][7] (Gargantua, or Garg for short), and

[3] An example implementation is at `http://www.doppeltext.com/`.

[4] `http://nlp.cs.nyu.edu/GMA/`

[5] `http://research.microsoft.com/en-us/downloads/aafd5dcf-4dcc-49b2-8a22-f7055113e656/`

[6] `http://mokk.bme.hu/en/resources/hunalign/`

[7] `http://sourceforge.net/projects/gargantua/`

in [Lamraoui and Langlais, 2013][8] (Yasa). These tools constitute, we believe, a representative panel of the current state-of-the-art in sentence alignment. Note that we leave aside here approaches inspired by Information Retrieval techniques such as [Bisson and Fluhr, 2000], which models sentence alignment as a cross-language information retrieval task, as well as the MT-based approach of Sennrich and Volk [2010], also based on some automatic translation of the "source" text, followed by a monolingual matching step.

GMA, introduced in [Melamed, 1999], is the oldest approach included in this panel, and yet one of the most effective: assuming "sure" lexical anchor points in the bitext map, obtained e.g. using bilingual dictionaries or cognate-based heuristics, GMA greedily builds a so-called "sentence map" of the bitext, trying to include as many anchors as possible, while also remaining close to the bitext diagonal. A post-processing step will take sentence boundaries into account to deliver the final sentence alignment. Note that GMA uses no length cues, and also that it has been shown to perform particularly well at spotting large omissions in a bitext [Melamed, 1996].

The approach of Moore [2002] implements a two-pass, coarse-to-fine, strategy: a first pass, based on sentence length cues, computes a first alignment according to the principles of length-based approaches [Brown et al., 1991; Gale and Church, 1991]. This initial alignment is used to train a simplified version of IBM model 1 [Brown et al., 1993], which provides the alignment system with lexical association scores. These scores are then used to refine the measure of association between sentences. This approach is primarily aimed at delivering high-confidence, 1:1 sentence alignments to be used as training material for data-intensive MT. Sentences that cannot be reliably aligned are discarded from the resulting alignment.

Hunalign is described in [Varga et al., 2005]. It also implements a two-pass strategy which resembles Moore's approach. Their main difference is that Hunalign also produces many-to-one and one-to-many alignment links, which are needed to ensure that all the input sentences are actually aligned.

Gargantua [Braune and Fraser, 2010] is, similarly to the approach of Deng et al. [2007] and our own approach, an attempt to improve the final steps of Moore's algorithm. The authors propose a two-pass unsupervised approach that works along the following lines: (a) search for an optimal alignment considering only links made of at most one sentence in each language (including null links); (b) heuristically improve this initial solution by merging adjacent links. A key observation in this work is that step (a) can be very fast. Like most works in this vein, this approach requires to explicitly model null links, and is prone to miss large untranslated portions on one side of the bitext.

The work of Lamraoui and Langlais [2013] also performs multiple passes over the data, but proceeds in the reverse order: predefined lexical associations (from a bilingual dictionary or from so-called *cognates*) are first used to prune the alignment search space to a restricted number of near-diagonal alignments (the diagonal is defined with respect to these sure *anchor* alignment points.). The second pass will then perform dynamic programming

[8] `http://rali.iro.umontreal.ca/rali/?q=en/yasa`

search with the additional help of a length-based model. In spite of its simplicity, this computationally lightweight approach is reported to perform remarkably well by Lamraoui and Langlais [2013].

Evaluation metrics

Sentence alignment tools are usually evaluated using standard *recall* (R) and *precision* (P) measures, combined in the *F-measure* (F), with respect to some manually defined gold alignment [Véronis and Langlais, 2000]. These measures can be computed at various levels of granularity: at the level of alignment links, of sentences, of words, and of characters. As gold references only specify alignment links, the other references are automatically derived in the most inclusive way. As a side effect, all metrics but the link-level ones *ignore null alignments*.[9] Our results are therefore based solely on the link-level F-measure, so as to reflect the importance of correctly predicting unaligned sentences in our targeted applicative scenario.

Evaluation corpora

The performance of sentence alignment algorithms is typically evaluated on reference corpora for which a gold alignment is provided. Manual alignments constitute the most reliable references, but are quite rare. In this work, we have used two sets of manual alignments of literary works: one is an extract of the BAF corpus [Simard, 1998], consisting of one book by Jules Verne (*De la Terre à la Lune*); the other corpus, a subset of the literary dataset presented in § 5.1.1, has been developed for a preliminary study described in [Yu et al., 2012a], and is made up of 4 novels translated from French into English and 3 from English into French. These two sets are both relatively small, and only contain bitexts in English and French. As these corpora were manually aligned, they may contain links of arbitrary types. These gold references constitute our main source of evidence for comparing the various algorithms used in this study.

For the purpose of a larger scale evaluation, we have also made use of two much larger, multi-parallel, corpora of publicly available books that are available on the Internet.[10] The corpus `auto en-fr` contains novels in English and French, and the corpus `auto en-es` contains novels in English and Spanish. These corpora are only imperfectly aligned, and are used to provide approximations of the actual alignment quality. Table 6.1 contains basic statistics for all these evaluation sets.

[9] Assume, for instance, that the reference alignment links two Foreign sentences F_1, F_2 to the single English sentence E: reference *sentence-level alignments* will contain both $[E; F_1]$ and $[E; F_2]$; likewise, reference *word-level alignments* will contain all the possible word alignments between tokens in the source and the target side, etc. For such metrics, missing the alignment of a large "block" of sentences is more harmful than missing a small one; likewise, misaligning short sentences is less penalized than misaligning longer ones.

[10] See `http://www.farkastranslations.com/bilingual_books.php`

	# books	lang.	# links	# sent. en	# sent. fr or es
BAF	1	en-fr	2,520	2,554	3,319
`manual en-fr`	7	en-fr	1,790	1,970	2,100
`auto en-fr`	24	en-fr	75,731	129,022	126,561
`auto en-es`	17	en-es	61,181	102,545	104,216

Table 6.1: Corpus statistics

Note that there is an obvious discrepancy in the BAF corpus between the number of sentences on the two sides of the bitext, which makes the automatic alignment of this book especially challenging. Also note that in the larger corpora, `auto en-fr` and `auto en-es`, manual alignment links are defined at the level of *paragraphs*, rather than at the level of sentences.

6.1.2 Baseline Evaluations

We evaluate the performance of the baseline sentence alignment tools on these four corpora, using the standard link-level F-measure. As explained above, the two larger corpora are aligned *at the paragraph level*, meaning that such resources cannot be readily used to compute alignment quality scores. Our solution has been to refine this coarse alignment by running the Gale and Church [1991] alignment program to compute within-paragraph sentence alignments, keeping the paragraph alignments unchanged from the reference. This approach is similar to the procedure used to align the Europarl corpus at the sentence level [Koehn, 2005], where reliable paragraph boundaries are readily derived from speaker turns or session changes. As a result, these semi-automatic references only contain a restricted number of link types as computed by Gale and Church [1991]'s program: 1:0, 0:1, 1:1, 1:2, and 2:1. We then take these partially correct alignments as pseudo-references for the purpose of evaluating alignment tools, while keeping in mind that the corresponding results will only be approximate. Our main evaluation results are in Table 6.2. For more details, see Tables C.1, C.2 and C.3 in Appendix C.

Regarding the gold corpus (`manual en-fr`), the numbers in the top part of Table 6.2 show that the alignment problem is far from solved, with an average F-score around 80 for the three best systems (Garg, GMA and Yasa).

On the two larger corpora, a legitimate question concerns the reliability of numbers computed using semi-automatic references. To this end, we manually aligned excerpts of three more books: Lewis Carroll's *Alice's Adventures in Wonderland*, Arthur Conan Doyle's *The Hound of the Baskervilles*, and Edgar Allan Poe's *The Fall of the House of Usher*, for a total of 1,965 sentences on the English side. We have then computed the difference in performance observed when replacing the gold alignments with semi-automatic ones. For these three books, the average difference between the evaluation on pseudo-references and on actual references is less than 2 points in F-measure; furthermore, these differences are

		GMA	BMA	Hun	Garg	Yasa
BAF		61.4	73.6	71.2	65.6	75.7
manual en-fr	min	53.5	57.4	54.3	51.7	59.9
	max	92.8	91.5	92.6	97.1	95.6
	mean	79.6	74.9	74.5	80.2	79.1
auto en-fr	min	62.1	47.1	56.6	56.4	62.3
	max	99.5	98.4	99.5	98.1	98.8
	mean	88.7	84.0	87.9	88.7	89.6
auto en-es	min	60.3	48.8	43.7	60.9	58.3
	max	96.5	98	96.4	98.8	98.4
	mean	82.8	78.4	81.0	80.5	82.7

Table 6.2: Baseline evaluation results, using link-level F-scores.

consistent across algorithms.

A second comforting observation is that the same ranking of baseline tools is observed across the board: Gargantua, GMA and Yasa tend to produce comparable alignments, outperforming Hunalign and BMA by approximately 2 to 3 F-measure points. It is also worth pointing out that on the two large datasets, less than 3% of the sentence links computed by BMA actually cross the reference paragraph boundaries, which warrants our assumption that BMA actually computes sure 1:1 links. Note that even for "easy" books, the performance falls short of what is typically observed for technical documents, with a F-measure hardly reaching 0.95; for difficult ones (such as Jane Austen's *Pride and Prejudice*), the best F-measure can be as low as 0.62. From this large-scale experiment, we can conclude that sentence alignment for literary texts remains challenging, even for relatively easy language pairs such as English-French or English-Spanish. It is expected that sentence alignment can only be more difficult when involving languages that are historically or typologically unrelated, or that use different scripts.

6.2 Methodological Analyses of Existing Methods

Most state-of-the-art aligners use a two-step approach. A first, relatively coarse decoding pass extracts a set of parallel word or sentence pairs that the system deems reliable (for instance using length-based information). These pairs serve as either anchor points to reduce the search space of subsequent steps, or as seeds to obtain better parallelism estimation tools (for instance a classifier or a bilingual lexicon), or both. A second decoding pass, using the information gathered during the first step, realigns the bitext. Most of these alignment tools are unsupervised, so that the system has to collect information from the sole bitext(s) that need to be aligned. In decoding, aligners often make the following assumptions: (a) alignment links lie around the bitext diagonal; (b) there exists a limited number

of link types. These two assumptions, together with the convention that alignment links are monotone and associate continuous spans,[11] warrant the use of dynamic programming (DP) techniques to perform the search. The resulting alignment tools are often light-weight and efficient, a major requirement if one wishes to process very large bitexts.

Despite the fact that existing sentence aligners achieve efficiency and good empirical performance on many corpora, we can identify the following issues, and (possible) ways of improvement:

- probabilistic alignment models typically assume a fixed prior distribution over link types, as well as specific choices for length distributions (e.g. Gaussian or Poisson). However, Wu [1994] demonstrated that these assumptions could be inaccurate, especially for language pairs that are not closely related;

- as shown in [Yu et al., 2012b], DP-based methods often give poor results for *null links*, i.e. links for which one side is empty. Among the five methods compared in this study, only Melamed [1999] predicted a similar number of null links as the reference, while others tended to miss a significant portion of them. A possible reason for this problem is the lack of a coherent scoring mechanism which would allow to fairly compare null and non-null links; this especially applies to methods using lexical clues;

- large-scale parallel corpora widely exist nowadays, and have proven very useful in many NLP tasks. Sentence alignment might also benefit from using such external resources, given that the efficiency concern can be less relevant in some scenarios. Parallel corpora can, for instance, be used as training examples for a parallelism classifier.

- probabilistic alignment models rely on *local* features, and ignore contextual evidences. It might be beneficial to explore structural dependencies in the training;

- the limitation on link types is also overly restrictive. Six main link types are used in most studies: 0:1, 1:0, 1:1, 2:1, 1:2, 2:2, and it is a fact that these types rassemble a large majority of links for most text genres. However, we will show that, in the `manual en-fr` corpus, the other types account for approximately 5% of the total number of links, a non-negligible portion for full-text alignment tasks. Besides, such intrinsic model errors can propagate during the DP process.

Based on these observations, we propose two models for sentence alignment. The first one uses a Maximum Entropy (MaxEnt) model as the alignment link scoring function, relying on external parallel corpora as training sets. The second method introduces a two-dimensional Conditional Random Fields (2D CRF) to enrich the structural modeling of sentence alignment, attempting to make use of the dependencies between alignment links and better represent null links.

[11] If a group of sentences on one side has no correspondence on the other side, they form a null link.

6.3 A Maxent-Based Algorithm

We present in this section our approach to obtain high-quality alignments. We borrow from [Yu et al., 2012a] the idea of a two-pass alignment process: the first pass computes high-confidence 1:1 links and outputs a partially aligned bitext containing sure links and residual *gaps*, i.e. parallel blocks of non-aligned sentences. These small blocks are then searched using a more computationally costly,[12] but also more precise, model, so as to recover the missing links: we propose, following again previous work, to evaluate possible intra-block alignments using a MaxEnt model, which is trained here on a large external corpus using a methodology and features similar to [Munteanu and Marcu, 2005; Smith et al., 2010; Mújdricza-Maydt et al., 2013]. These steps are detailed below.

6.3.1 A MaxEnt Model for Parallel Sentences

Any sentence alignment method needs, at some point, to assess the level of parallelism of a sentence pair, based on a surface description of these sentences. As discussed above, two kinds of clues are widely employed in existing systems to perform such assessment: sentence lengths and lexical information. Most dynamic programming-based approaches further impose a prior probability distribution on link types [Gale and Church, 1993].

Our system combines all the available clues in a principled, rather than heuristic, way, using a MaxEnt model.[13] For any pair of sentences $\mathbf{l} = [E; F]$, the model computes a link posterior probability $p(Y = y|E, F)$, where Y is a binary variable representing the existence of an alignment link. The rationale for using MaxEnt is (a) that it is possible to efficiently integrate as many features as desired into the model, and (b) that we expect that the resulting posterior probabilities will be less peaked towards extreme values than what we have observed with generative alignment models such as Moore's model. We detail the features used in our model in § 6.5.2. Note that the posterior probabilities computed by a MaxEnt model can also be used as confidence scores. That is, a trained MaxEnt model can be used as a confidence estimation tool for sentence alignment links.

A second major difference with existing approaches is our use of a very large set of high-confidence alignment links to train our model. Indeed, most sentence alignment systems are *endogenous*, meaning that they only rely on information extracted from the bitext under consideration. While this design choice was probably legitimate in the early 1990's, it is much more difficult to justify it now, given the wide availability of sentence-aligned parallel data, such as the Europarl corpus [Koehn, 2005], which can help improve alignment systems.

To better match our main focus, which is the processing of literary texts, we collected positive instances from the same publicly available source, extracting all 1-sentence paragraphs as reliable alignment pairs. This resulted in a gold set of approximately 18,000 sentences pairs for French/English (out of a grand total of 125,000 sentence pairs). Negative

[12] Too costly, in fact, to be used in the first pass over the full bitext.

[13] We used the implementation from `http://homepages.inf.ed.ac.uk/lzhang10/maxenttoolkit.html`.

examples are more difficult to obtain and are generated artificially as explained in § 6.5.3.

Finally, note that our model, even though it is trained on 1:1 sentence pairs, can in fact evaluate any pair of segments. We make use of this property in our implementation.[14]

6.3.2 Computing Sure One-to-one links

As in many existing tools implementing a multi-step strategy, the main purpose of the first step is to provide a coarse alignment, in order to restrict the search space of the subsequent steps. In our approach, the links computed in the first step are mainly used as anchor points, which makes the more costly search procedure used in the second step feasible. Since we do not reevaluate these anchor links, they should be as reliable as possible.

Our current implementation uses the algorithm of Moore [2002] to identify these sure anchors: as explained in § 6.1.1, this algorithm tends to obtain a very good precision, at the expense of a less satisfactory recall. Furthermore, Moore's algorithm also computes posterior probabilities for every possible link, which are then used as confidence scores. The good precision of this system is due to its discarding all links with a posterior probability smaller than 0.5. As explained below, such confidence measures can be used to control the quality of the anchor points that are used downstream. The result of this first step is illustrated in Table 6.3.

en_1	Poor Alice!	Pauvre Alice !	fr_1
en_2	It was as much as she could do, lying down on one side, to look through into the garden with one eye; but to get through was more hopeless than ever : she sat down and began to cry again.	C'est tout ce qu'elle put faire, après s'être étendue de tout son long sur le côté, que de regarder du coin de l'oeil dans le jardin.	fr_2
		Quant à traverser le passage, il n'y fallait plus songer.	fr_3
		Elle s'assit donc, et se remit à pleurer.	fr_4
en_3	"You ought to be ashamed of yourself," said Alice,"a great girl like you," (she might well say this), "to go on crying in this way!	«Quelle honte !» dit Alice.	fr_5
		«Une grande fille comme vous» («grande» était bien le mot) «pleurer de la sorte !	fr_6
en_4	Stop this moment, I tell you!"	Allons, finissez, vous dis-je ! »	fr_7
en_5	But she went on all the same, shedding gallons of tears, until there was a large pool all round her, about four inches deep and reaching half down the hall.	Mais elle continue de pleurer, versant des torrents de larmes, si bien qu'elle se vit à la fin entourée d'une grande mare, profonde d'environ quatre pouces et s'étendant jusqu'au milieu de la salle.	fr_8

Table 6.3: An example alignment computed by Moore's algorithm for *Alice's Adventures in Wonderland*. The first and third anchor links delineate a 2×5 gap containing 2 English and 5 French sentences.

[14] Training the model on 1:1 links, and using it to assess multi-sentence links creates a small methodological bias. Our attempts to include other types of links during training showed insignificant variance in the performance.

6.3.3 Closing Alignment Gaps

The job of the second step is to complete the alignment by filling in first-pass gaps. Assume that a gap begins at index o and ends at index q ($o \leq q$) on the English side, and begins at index r and ends at index t ($r \leq t$) on the Foreign side. This step aims at refining the alignment of sentences $\mathbf{E}_o^q$ and $\mathbf{F}_r^t$, assuming that these blocks are already (correctly) aligned as a whole.

If one side of the gap is empty,[15] then nothing is to be done, and the block is left as is. In all other cases, the block alignment will be improved by finding a set of n links $\{\mathbf{l}_1, \ldots, \mathbf{l}_n\}$ maximizing the following score:

$$\prod_{i=1}^{n} \frac{p(1|\mathbf{l}_i = [\mathbf{E}_i; \mathbf{F}_i])}{\alpha \times \texttt{size}(\mathbf{l}_i)}, \tag{6.1}$$

where $\texttt{size}(\mathbf{l})$ is the size of link $\mathbf{l}$, defined as the product of the number of sentences on the source and target sides and α is a hyper-parameter of the model. $\mathbf{E}_i$ and $\mathbf{F}_i$ are the two sentence sequences of $\mathbf{l}$ on the two sides. $p(1|\mathbf{l}_i = [\mathbf{E}_i; \mathbf{F}_i])$ is the probability score that $[\mathbf{E}_i; \mathbf{F}_i]$ is a good link, computed by the MaxEnt model.

This score computes the probability of an alignment as the product of the probabilities of individual links. The size-based penalty is intended to prevent the model from preferring large blocks over small ones: this is because the scores of alignments made of large blocks contain fewer factors in Equation (6.1); dividing by $\alpha \times \texttt{size}(\mathbf{l}_i)$ mitigates this effect and makes scores more comparable.

In general, the number of sentences in a gap makes it possible to consider all the sub-blocks within a gap. In some rare cases, however, the number of sub-blocks was too large to enumerate them and we had to impose an additional limitation on the number of sentences on both sides of a link. Our inspection of several manually annotated books revealed that links are seldom composed of more than 4 sentences on either side and we have used this limit in our experiments (this is parameter δ in the algorithms below). Note that this is consistent with previous works such as [Gale and Church, 1993; Moore, 2002] where only small links (the largest alignment links are 2:1) are considered. Our two pass approach allows us to explore more alignment types, the largest link type being 4:4. We compare below two algorithms for finding the optimal set of links: a greedy search presented in [Yu et al., 2012b], and a novel (exact) algorithm based on dynamic programming.

Greedy search

Greedy search is described in Algorithm 1: it simply processes the possible links in decreasing probability order and inserts them into the final alignment unless they overlap with an existing alignment link. For a gap with M English sentences and N Foreign sentences and

[15] This happens when the algorithm detects two consecutive anchors $[E_o; F_r]$ and $[E_q; F_{r+1}]$ where $q > o+1$ (and similarly in the other direction).

fixed δ, the worst-case complexity of the search is $O(M \times N)$, which, given the typical small size of the gaps (see Figure 6.3), can be computed very quickly.

Algorithm 1 Greedy search

Input: block = $[\mathbf{E}_o^q; \mathbf{F}_r^t]$, priority list L, max gap size δ
Output: result list R

```
Generate the set of all possible links S between sub-blocks in [E_o^q; F_r^t]
for all l in S do
    insert l into L, with score(l) defined by (6.1)
end for
while L not empty do
    pop top link l* from L
    insert l* into R
    remove any link that intersects or crosses l* from L
end while
complete R with null links
```

The result of Algorithm 1 is a collection of links which do not overlap with each other and respect the monotonicity constraint. Note that even though the original list L does not contain any null link, the resulting alignment R may contain links having an empty source or target side. For instance, if links $[\mathbf{E}_b^{b+1}; \mathbf{F}_{d-1}^d]$ and $[\mathbf{E}_{b+2}^{b+3}; \mathbf{F}_{d+2}^{d+3}]$ are selected, then the null link $[; \mathbf{F}_{d+1}^{d+1}]$ will also be added to R.

Dynamic programming search

The other search algorithm considered in this study is based on dynamic programming (DP). Given a series of English sentences $\mathbf{E}_o^q$ and the corresponding Foreign sentences $\mathbf{F}_r^t$, DP tries to find the set of links yielding maximal global score. Our DP search procedure is described in Algorithm 2: as typical in DP approaches, the algorithm merely amounts to filling a table D containing the score of the best alignments of sub-blocks of increasing size. The search complexity for a gap containing M English and N Foreign sentences is $O(M \times N)$. The constant term in the complexity analysis depends on the types of links DP has to consider: as explained above, we only consider here links having less than 4 sentences on each side.

An important issue for DP search is that the probability of null links must be estimated, which is difficult for MaxEnt, as no such information can be found in the training corpus. In greedy search, which only considers non-null links, this problem does not exist. In DP, however, null links appear in all backtraces. We have adopted here a simple method, which is to estimate the score of a null link $\mathbf{l} = [; E]$ or $\mathbf{l} = [E;]$ as:

$$\texttt{score}(E,) = \texttt{score}(, E) = \exp(-\beta|E|) \tag{6.2}$$

Algorithm 2 The dynamic programming search

Input: Gap=$[\mathbf{E}_o^q; \mathbf{F}_r^t]$), empty tables D and B, max gap size δ
Output: link list R

```
for p ← o to q do
  for s ← r to t do
    max ← 0
    for m ← 0 to min(p, δ) do
      for n ← 0 to min(s, δ) do
        cur ← D(p − m, s − n) + score(E^p_{p−m}, F^s_{s−n})
        if cur ≥ max then
          max ← cur
          D(p, s) ← max
          B(p, s) ← (p − m, s − n)
        end if
      end for
    end for
  end for
end for
Back trace on B to find R
```

where $|u|$ returns the number of tokens in u and $\beta > 0$ is a hyper-parameter of the method. The intuition is that long null links should be less probable than shorter ones.

6.4 The 2D CRF Model

While the MaxEnt-based method follows the common pattern of sentence alignment techniques, our second method makes substantial new design choices. Inspired by the model of Mújdricza-Maydt et al. [2013], we propose a two-dimensional CRF model for sentence alignment. We use a binary variable to model the existence of the parallelism relation between one source-to-target sentence pair, and include contextual information in our predictions. Decoding consists of classifying each variable as negative or positive. Furthermore, the model structure is richer than that of Mújdricza-Maydt et al. [2013] and includes an explicit representation of null links.

6.4.1 The Model

Given a sequence of source language sentences $\mathbf{E}_1^I = (E_1, ..., E_I)$ and a sequence of target sentences $\mathbf{F}_1^J = (F_1, ..., F_J)$, we propose a 2D CRF model to predict the presence of links between any pair of sentences $[E_i; F_j]$, where $1 \leq i \leq I, 1 \leq j \leq J$. Note that similar models have also been developed for sub-sentential alignments [Niehues and Vogel, 2008; Cromières

and Kurohashi, 2009; Burkett and Klein, 2012]. Each pair $[E_i; F_j]$ gives rise to a binary variable $\mathbf{y}_{i,j}$, whose value is 1 (*positive*) if E_i is aligned to F_j, and 0 (*negative*) otherwise. For the sequence pair $\mathbf{E}_1^I$ and $\mathbf{F}_1^J$, there are $I \times J$ such variables, collectively denoted as $\mathbf{y}$. Dependencies between links are modeled as follows. For each pair $[E_i; F_j]$, we assume that the associated variable $\mathbf{y}_{i,j}$ depends on $\mathbf{y}_{i-1,j}$, $\mathbf{y}_{i+1,j}$, $\mathbf{y}_{i,j-1}$, $\mathbf{y}_{i,j+1}$, $\mathbf{y}_{i-1,j-1}$ and $\mathbf{y}_{i+1,j+1}$. In other words, it depends on the presence of links $[E_{i-1}; F_j]$, $[E_{i+1}; F_j]$, $[E_i; F_{j-1}]$, $[E_i; F_{j+1}]$, $[E_{i-1}; F_{j-1}]$ and $[E_{i+1}; F_{j+1}]$. Figure 6.1 displays a graphical representation of the model.

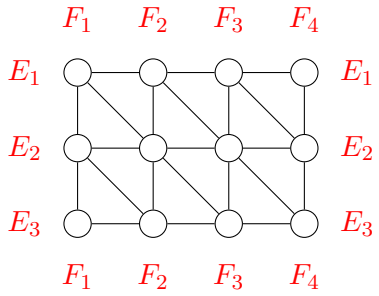


Figure 6.1: The 2D CRF model, for a bitext of 3 source $E_1 - E_3$ and 4 target sentences $F_1 - F_4$.

The topology of our model differs from the proposal of Mújdricza-Maydt et al. [2013], where each diagonal of the alignment matrix was modeled as a linear chain CRF. This topology captured the important diagonal direction dependency, but did not encode the horizontal or vertical dependencies. Another important difference lies on the generation of final outputs. In the model of Mújdricza-Maydt et al. [2013], variables encode labels corresponding to link types (e.g. 1:1, 2:1). Note that this encoding makes it impossible to include all link types, and has also a bearing on the computational cost, since the inference complexity of a linear chain CRF is quadratic in the number of labels. As a result, these authors only considered 6 link types (1:1, 1:2, 2:1, 1:3, 3:1, F).[16] In our model, all prediction variables are binary. We generate final links using the transitive closure operation according to sentence alignment conventions, which can theoretically lead to any possible link type. For instance, an all zero-valued j^{th} column indicates an unaligned target sentence F_j; if $\mathbf{y}_{p,q}$ is the only positive value in the p^{th} row and q^{th} column, then there is a 1:1 link $[E_p; F_q]$, etc. In fact, the model can express finer correspondences than conventional alignment link representations. For example, if both E_{u-1} and E_u are aligned to F_{v-1}, E_u is further aligned to F_v, our formalism can represent exactly the relation, while the alignment link representation would contain a coarser 2:2 link $[E_{u-1}, E_u; F_{v-1}, F_v]$.

We use two kinds of clique potentials in our model: *node potentials* and *edge potentials*. We impose that all single node cliques use the same clique template, i.e. they share the

[16] F stands for all other link types or unaligned sentences.

same set of feature functions and corresponding weights. For edge potentials, we use distinct clique templates for vertical, horizontal and diagonal edges. One main limitation of this model is that it does not include long distance dependencies, which makes it difficult to encode certain types of constraints (e.g. that alignment links should not cross). The model for a pair of sentence sequences $[\mathbf{E};\mathbf{F}]$ (as a shorthand for $[\mathbf{E}_1^I;\mathbf{F}_1^J]$) can be written as:

$$p(\mathbf{y}|\mathbf{E},\mathbf{F}) = \frac{1}{Z(\mathbf{E},\mathbf{F})}\prod_{\nu}\Phi_n(\mathbf{y}_\nu)\Phi_v(\mathbf{y}_\nu)\Phi_h(\mathbf{y}_\nu)\Phi_d(\mathbf{y}_\nu)$$

where $\nu \in \{(i,j) : 1 \le i \le I, 1 \le j \le J\}$, $\Phi_n(\mathbf{y}_\nu)$ stands for the single node potential at ν, $\Phi_v(\mathbf{y}_\nu)$ represents the potential on the vertical edge connecting ν and the node just below it:

$$\forall j, \Phi_v(\mathbf{y}_{i,j}) = \begin{cases} 1 & \text{if} \quad i = I \\ \Phi_v(\mathbf{y}_{i,j},\mathbf{y}_{i+1,j}) & \text{if} \quad 1 \le i < I \end{cases}$$

$\Phi_h(\mathbf{y}_\nu)$ (the horizontal potential) and $\Phi_d(\mathbf{y}_\nu)$ (the diagonal potential) are defined similarly. $Z(\mathbf{E},\mathbf{F}) = \sum_{\mathbf{y}'}\prod_\nu \Phi_n(\mathbf{y}'_\nu)\Phi_v(\mathbf{y}'_\nu)\Phi_h(\mathbf{y}'_\nu)\Phi_d(\mathbf{y}'_\nu)$ is the normalization factor (*the partition function*) of the CRF. All potentials take the generic form of a log-linear combination of feature functions:

$$\Phi_\nu(\mathbf{y}_\nu) = \exp\{\boldsymbol{\theta}^\top \mathbf{G}_\nu(\mathbf{y}_\nu)\},$$

where $\mathbf{G}_\nu$ and $\boldsymbol{\theta}$ are the feature and weight vectors. We also use ℓ_2 regularization with scaling parameter $\alpha > 0$.[17]

6.4.2 Learning the 2D CRF Model

The conventional learning criteria for CRF is the Maximum Likelihood Estimation (MLE). For a set of fully observed training instances $\mathcal{A} = \{(\mathbf{E}^{(s)},\mathbf{F}^{(s)},\mathbf{y}^{(s)})\}$, MLE consists of maximizing the log-likelihood of the training set with respect to model parameters $\boldsymbol{\Theta} = \{\boldsymbol{\theta}, \alpha\}$. The log-likelihood is concave with respect to the weight vector, which warrants the use of convex optimization techniques to obtain parameter estimates. In order to do this, we need the gradient of the likelihood function with respect to the weight vector.

Computing the gradients requires two kinds of marginal probabilities: single node marginals $p(\mathbf{y}_{i,j}|\mathbf{E}^{(s)},\mathbf{F}^{(s)})$ and edge marginals $p(\mathbf{y}_{i,j},\mathbf{y}_{i+1,j}|\mathbf{E}^{(s)},\mathbf{F}^{(s)})$, $p(\mathbf{y}_{i,j},\mathbf{y}_{i,j+1}|\mathbf{E}^{(s)},\mathbf{F}^{(s)})$, and $p(\mathbf{y}_{i,j},\mathbf{y}_{i+1,j+1}|\mathbf{E}^{(s)},\mathbf{F}^{(s)})$. We need to perform inference to compute these marginals. Since the topology of our model contains loops, we use the *Loopy Belief Propagation* (LBP) inference algorithm. Even though LBP is an *approximate* inference algorithm with no convergence guarantee, Murphy et al. [1999] observe that it often gives reasonable estimates (assuming it converges).

[17] In the experiments, α is tuned on a development set, and takes the value 0.1.

For a tree-structured undirected graphical model, the message from a node $\mathbf{y}_\mu$ to a neighboring node $\mathbf{y}_\nu$ takes the following form [Wainwright and Jordan, 2008]:

$$m_{\mu\nu}(\mathbf{y}_\nu) \propto \sum_{\mathbf{y}_\mu} \Phi(\mathbf{y}_\mu)\Phi(\mathbf{y}_\nu, \mathbf{y}_\mu) \prod_{\gamma \in N(\mu)\backslash\nu} m_{\gamma\mu}(\mathbf{y}_\mu)$$

where $N(\mu)$ denotes the set of neighbors of μ. LBP is performing such message passing procedure on a cyclic graph. Once message passing has converged, the single node and edge marginals (a.k.a. "beliefs") are expressed as:

$$b_\nu(\mathbf{y}_\nu) \propto \Phi(\mathbf{y}_\nu) \prod_{\gamma \in N(\nu)} m_{\gamma\nu}(\mathbf{y}_\nu)$$

$$b_{\mu\nu}(\mathbf{y}_\mu, \mathbf{y}_\nu) \propto \Phi(\mathbf{y}_\mu)\Phi(\mathbf{y}_\nu)\Phi(\mathbf{y}_\nu, \mathbf{y}_\mu) \prod_{\delta \in N(\mu)\backslash\nu} m_{\delta\mu}(\mathbf{y}_\mu) \prod_{\gamma \in N(\nu)\backslash\mu} m_{\gamma\nu}(\mathbf{y}_\nu)$$

In practice, it is possible that LBP does not converge for certain training instances. In this case, we simply stop it after 100 iterations. Convex optimization routines also require to compute the log-partition function $\log Z(E, F)$, as a part of the likelihood function. LBP approximates this quantity with the Bethe Free Energy [Yedidia et al., 2001].

In learning, we first train the CRF without any edge potential (thus making the model similar to the simpler MaxEnt model), and use it to initialize the parameter vector of node potentials. We then randomly initialize other parameters,[18] and use the L-BFGS algorithm [Liu and Nocedal, 1989] implemented in the SciPy package to perform parameter learning, this time with all potentials.

6.4.3 Search in the 2D CRF Model

For the 2D CRF model, we perform the search in multiple steps. First, we run the BMA algorithm [Moore, 2002] to extract high-confidence 1:1 links. This algorithm first extracts reliable 1:1 sentence pairs from a bitext, using only length information, then trains a small IBM Model 1 based on these links, finally realigns the bitext using both length and lexical information. It returns a set of 1:1 sentence pairs. According to our evaluation results, BMA tends to obtain a very good precision, at the expense of a less satisfactory recall. Furthermore, BMA computes posterior probabilities for every possible link, which are then used as confidence scores. We filter the result links with a very high posterior probability threshold (≥ 0.99999) (this threshold is much higher than BMA's default choice). These links segment the entire search space into sub-blocks. For each sub-block, we construct a 2D CRF model, and perform decoding. As exact Maximum A Posteriori decoding is

[18] See [Sutton, 2008] pages 88–89 for a discussion on parameter initialization of general CRFs trained using LBP.

intractable, instead, we run max-product LBP, and pick the *local best* label for each node. The label assigned to a variable $\mathbf{y}_{i,j}$ is

$$\arg\max_{l\in\{0,1\}} b_{i,j}(l)$$

This procedure returns a set of *sentence-level* links. Since the sizes of the sub-blocks are often small (generally smaller than 10×10), decoding is very fast in practice.

Figure 6.2 displays an *alignment prediction matrix*.[19] It contains four types of cells, corresponding to four types of predictions: true positive (red, with underlined score), true negative (white, with normal score), false positive (yellow, with overlined score), false negative (cyan, with hatted score). The score in each cell is the marginal probability of the pair being positive, as computed by the CRF. A red or yellow cell indicates a sentence-level link predicted by the model.

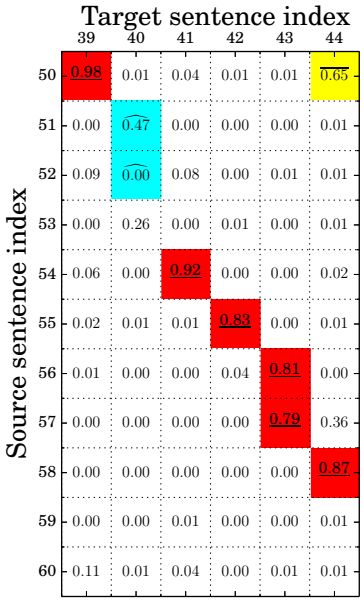


Figure 6.2: An alignment prediction matrix.

Two types of errors exist in the alignment prediction matrix: false negatives (cyan cells with hatted scores) and false positives (yellow cells with overlined scores). We cannot easily deal with false negatives. False positives introduce noise, for example, the pair $[50; 44]$ in

[19] Note these matrices are drawn just after the CRF decoding, before the post-processing described below.

Figure 6.2 (the upper right corner). The two positive pairs [50; 39] and [50; 44] lead to two separate links involving the same source sentence, which violates the general convention of sentence alignment. In fact, the pair [50; 44] is clearly wrong: it links the first source sentence with the last target one, thus overlapping with all other positive sentence-level links.[20] In our experiments, in all sub-blocks, true positive sentence-level links always lie around the main diagonals. We have used the following heuristics to smooth the alignment prediction matrix:

1. perform a linear regression on all predicted positive sentence-level links, then take a band of fixed width around the regression line, and drop positive links that lie outside of this band.[21]

2. if after this step, there are still separate links involving the same sentence, we take the positive sentence-level links in the surrounding window with width 5, and discard the ones which are inconsistent with the surrounding links;

3. if it is still undecidable, we perform again a linear regression of positive sentence-level links in the surrounding window, and discard the link that is farthest away from the regression line.

In practice, step 3 was hardly performed.

Finally, to turn sentence-level links into *alignment-level* links, we apply the following rules:

1. consecutive sentence-level links in the horizontal or vertical directions are combined into a large alignment-level link;

2. a sentence-level link without horizontal or vertical neighbors becomes a 1 : 1 type alignment-level link.

These rules follow from the interpretation of our model, where an *n:m* type alignment-level link decomposes into $n * m$ sentence-level links.

6.5 Experiments and Analyses

We conducted experiments on the same corpora as in Section 6.1. We first discuss the impact of various hyper-parameters of the system on its overall performance. Then we report results of our alignment algorithm on these corpora, compared to those of other methods.

[20] Note that this particular matrix was computed by an early version of the 2D CRF model. We show it here for illustration purpose. In later versions, the model is augmented with features capturing the relative position information, which effectively prevents this kind of errors.

[21] The band width is taken to be half of the number of sentences of the shorter one of the two sides.

6.5.1 A Study of Moore's Alignments (BMA)

Our methods heavily rely on the information computed by Moore's algorithm in the first step, since we use those links as anchor points to prune the search space of the second step. The number of anchor points has an effect on the computational burden of the search algorithm. Their quality is even more important as incorrect anchors hurt the performance in two ways: they count as errors in the final result, and they propagate erroneous block alignments for the second step, thereby generating additional alignment errors. It is then natural to investigate the quality of BMA's results from several perspectives.

On the BAF corpus, which contains a complete reference alignment, Moore's algorithm returns 1,944 1:1 links, among which only 1,577 are correct (P=0.81). The 1,944 links define 445 gaps to be aligned by the second alignment pass. The quality of these gaps is also relevant. We define a gap as correct if it can be fully decomposed into links that appear in the reference set. Among the 445 gaps to be aligned, 180 are incorrect. Finally note that the noise in each incorrect gap also negatively impacts the search.

Moore's algorithm associates a confidence score with each link. As shown in Figure 6.3, using a tighter threshold to select the anchor links significantly improves the precision, and also reduces the number of wrong gaps, at the expense of creating larger blocks.

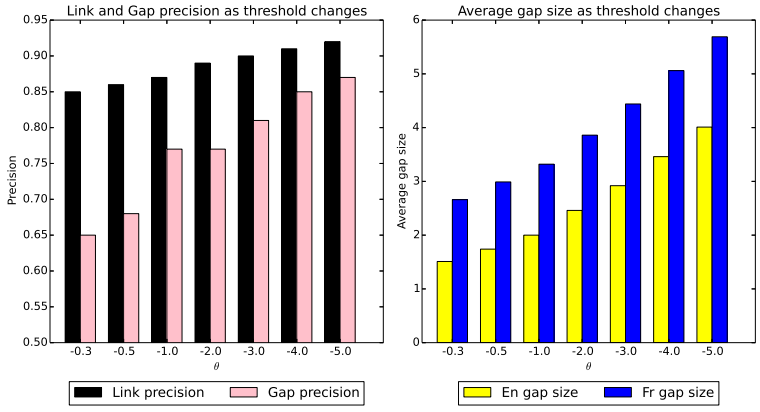


Figure 6.3: Varying confidence threshold causes link precision, gap precision (left) and gap size (right) to change. Numbers are computed over all the manually aligned data (BAF and `manual en-fr`). θ is a transformation of the actual threshold t: $\theta = \lg(1-t)$.

In Figure 6.3, we plot precision as a function of θ, where the actual threshold is equal to $1 - 10^{\theta}$: for instance, $\theta = -0.3$ corresponds to a threshold of 0.5, and $\theta = -5$ corresponds to a threshold of 0.99999.[22] On BAF, using a very high confidence threshold of 0.99999

[22]Posterior link probabilities computed by generative models such as those used by Moore's algorithm tend to be very peaked, which explains the large number of very confident links.

improves the precision of anchor points from 0.81 to 0.89, and the ratio of correct gaps rises from 0.6 to 0.82. On the `manual en-fr` corpus, threshold 0.99999 yields an anchor point precision of 0.96 and a correct gap ratio of 0.94. In both cases, a high confidence threshold significantly reduces the number of wrong gaps. In our implementation, we set the threshold to 0.9999 to reach an acceptable trade-off between correct gap ratio and gap sizes.

6.5.2 Feature Engineering

Features of the MaxEnt Model

The core component of the MaxEnt-based system is a classifier, which, given a pair $[E; F]$ of source and target sentences, evaluates the level of parallelism between them. By allowing ourselves to also consider external resources, we can use more complex and effective feature families. We have designed the following list of features for MaxEnt:

1. The *length* (in character) of E and F, and the length ratio, discretized into 10 intervals, so this family contains a total of 12 features.

2. The *number of identical tokens* in E and F . We define 5 features for values (0, 1, 2, 3, 4+).

3. The *number of cognates*[23] in E and F, also defining 5 features for values (0, 1, 2, 3, 4+).

4. The *word pair lexical features*, one for each pair of words co-occurring at least once in a parallel sentence. For example, if the first token in E is "Monday" and the first token in F is "Lundi", then the pair "Monday-Lundi" defines a word pair lexical feature.

5. *Sentence translation* score features. For a pair of sentences $E = \mathbf{e}_1^I$ and $F = \mathbf{f}_1^J$, we use the IBM Model 1 score [Brown et al., 1993]:

$$T_1(E,F) = \frac{1}{J}\sum_{j=1}^{J}\log(\frac{1}{I} * \sum_{i=1}^{I} p(f_j|e_i))$$

$$T_2(E,F) = \frac{1}{I}\sum_{i=1}^{I}\log(\frac{1}{J} * \sum_{j=1}^{J} p(e_i|f_j))$$

After discretizing these values, we obtain 10 features for each direction.

[23] Following Simard et al. [1993a], we call a pair of words "cognates" if they share a prefix of at least 4 characters.

6. *Longest continuous covered span* features. A word e_i is said to be covered if there exists one word f_j such that the translation probability $t(e_i|f_j)$ in the IBM Model 1 table is larger than a threshold (1e−6 in our experiments). A long span of covered words is an indicator of parallelism. We compute the length of the longest covered spans on both sides, and normalize them by their respective sentence length. This family contains 20 features.

7. *Uncovered words*: The notion of coverage is defined as above. We count the number of uncovered words on both sides and normalize by the sentence length. This family contains 20 features, 10 on each side.

8. *Unlinked words* in the IBM 1 alignment. A word e_i is said to be linked if in an alignment $\mathcal{Z}$, there exists some index j such that $z_j = i$.[24] Large portions of consecutive unlinked words is a sign of non-parallelism. These counts are normalized by the sentence length, and yield 20 additional features.

9. *Fertility* features : The fertility of a word e_i is the number of indices j that satisfy $z_j = i$. Large fertility values indicate non-parallelism. We take, on each side, the three largest fertility values, and normalize them with respect to the sentence lengths. This yields 60 supplementary features.

The feature families (1–4) are borrowed from [Yu et al., 2012a], and are used in several other studies on supervised sentence alignment, e.g. [Munteanu and Marcu, 2005; Smith et al., 2010]. All other features rely on IBM Model 1 scores and can only be computed reliably on large (external) sources of data.

	Model accuracy		
	~110K tokens	~1M tokens	~5M tokens
Family 1	0.778	0.873	0.859
+Family 2 and 3	0.888	0.869	0.879
+Family 4	0.957	0.976	0.977
+Family 5	0.943	0.985	0.987
+Family 6	0.912	0.979	0.986
+Family 7	0.913	0.975	0.986
+Family 8	0.913	0.979	0.988
+Family 9	0.913	0.981	0.988

Table 6.4: Evaluation of MaxEnt with varying feature families and training data

[24] This notion is different from coverage and assumes that an optimal 1 : 1 word alignment has been computed based on IBM 1 model scores. Words can be covered, yet unlinked, when all their possible matches are linked to other words.

To evaluate the usefulness of these various features, we performed an incremental feature selection procedure. We first train the model with only one feature family, then add the other families one by one, monitoring the performance of the MaxEnt model as more features are included. For this study, model performance is measured by the prediction accuracy, that is the ratio of examples (positive and negative) for which the model makes the right classification. Because the new features are all based on IBM Model 1, the size of the training corpus also has an important impact. We have thus set up three datasets of increasing sizes: the first one contains around 110,000 tokens, which is the typical amount of data that Moore's algorithm would return upon aligning one single book; the second one contains 1,000,000 tokens; the third one includes all the parallel literary data collected for this study and totals more than 5,000,000 tokens. Each data set is split into a training set, a development set and a test set, using 80% for training, 10% for tuning, and 10% for testing. For these experiments, the model is trained with 30 iterations of the L-BFGS algorithm with a Gaussian prior, which is tuned on the development set.

Table 6.4 gives the performance of the MaxEnt model on the test set as more features are included. Note that families 2 and 3 are added together.

As expected, the new families of features (5-9) are hardly helping when trained on a small data set; as more training data are included in the model, the accuracy increases, allowing the system to divide the error rate by more than 2 in comparison to the best small data condition.

In our applicative scenario, not only do we want the model to make the right alignment decisions, but also expect that it can do so with a large confidence. To check that this is actually the case, we plot the ROC (Receiver Operating Characteristic) curve in Figure 6.4.[25] We only display the ROC curves for the medium and large data sets.

From the two ROC curves, we can observe that on the medium size data set, the model achieves very good performance in all settings, with large areas under curve (AUC). In both figures, we can see that the use of feature families 6 to 9 hardly helps to improve the confidence of the model over the results of the first five. In the experiments reported below, we only use the feature families 1 to 5 for the MaxEnt model.

Features of the 2D CRF Model

In the 2D CRF model, feature functions take the form $f(\mathbf{E}_1^I, \mathbf{F}_1^J, i_1, j_1, i_2, j_2, \mathbf{y}_{i_1,j_1}, \mathbf{y}_{i_2,j_2})$, where $\mathbf{E}_1^I$ is the source sequence of I sentences, $\mathbf{F}_1^J$ the target sequence of J sentences, (i_1, j_1) and (i_2, j_2) neighboring source-target indices, $\mathbf{y}_{i_1,j_1}$ and $\mathbf{y}_{i_2,j_2}$ respectively corresponding labels (0 or 1). Thus, the 2D CRF model can use all features of the MaxEnt model in its single-node potentials, and those involving link-dependencies in the edge potentials. For each pair $[E_i, F_j]$, we compute the following families of features:

– Families 1, 2, 3, 4, 5 of the MaxEnt model;

[25] See § 7.1.3 for a detailed introduction of ROC curves.

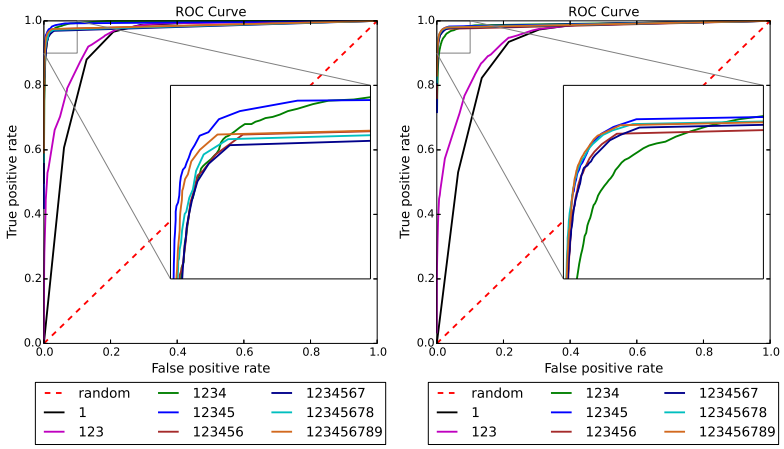


Figure 6.4: ROC curves on the medium size (left) and large (right) data sets. The embedded box is a zoom over the top-left corner.

– (Family 6) The *span coverage*. We split a string into several **spans** by segmenting on punctuations (except for the quotation marks). For each source span $span\_e$, we compute the translation score $T_2(span\_e, F_j)$. If the score is larger than a threshold,[26] we consider $span\_e$ as being **covered**. We then compute the ratio of covered source spans and the ratio of covered target spans, and discretize each into 10 features.

– (Family 7) The *label transition*. These features capture the regularity of the transition of labels from one node (E_i, F_j) to one of its neighbors (e.g. (E_{i+1}, F_j)). For each of the three types of neighbors (vertical, horizontal, diagonal), we define four label transition features (because our prediction variables are binary). For example, for the vertical template, we define

$$g_{00}(i,j) = \delta\{\mathbf{y}_{i,j} = 0 \wedge \mathbf{y}_{i+1,j} = 0\}$$
$$g_{01}(i,j) = \delta\{\mathbf{y}_{i,j} = 0 \wedge \mathbf{y}_{i+1,j} = 1\}$$
$$g_{10}(i,j) = \delta\{\mathbf{y}_{i,j} = 1 \wedge \mathbf{y}_{i+1,j} = 0\}$$
$$g_{11}(i,j) = \delta\{\mathbf{y}_{i,j} = 1 \wedge \mathbf{y}_{i+1,j} = 1\}$$

where δ is the Kronecker delta function. We have similar features for horizontal and diagonal transitions. In total, this family contains 12 features.

– (Family 8) The *augmented length difference ratio*. This family only applies to the vertical and horizontal edge potentials, under the condition that the two neighboring

[26] In our experiments, the threshold is set to $\log(1e{-}3)$

pairs are both positive. In the vertical (resp. horizontal) case, we combine the two consecutive source (resp. target) sentences E_i, E_{i+1} (resp. F_j, F_{j+1}) into one new sentence E' (resp. F'), then apply the computations carried out for feature family 1 for the pair (E', F_j) (resp. (E_i, F')). This evaluates the impact of keeping two sentences separated or merging them together.

– (Family 9) The *augmented translation score*. This family only applies to vertical and horizontal edge potentials, under the condition that the two neighboring pairs are both positive. We construct E' (resp. F') as in the previous feature family. We then compute the augmented translation score $T_1(E', F_j) - T_1(E_i, F_j)$ (resp. $T_2(E_i, F') - T_2(E_i, F_j)$). The intuition is that a longer partial translation is better than a shorter one. Each score is discretized into 10 features.

Note feature families 7, 8 and 9 are computed only when possible. Feature families 6, 8 and 9 are new in our model. Others have been used in previous methods, for instance, [Munteanu and Marcu, 2005; Yu et al., 2012b; Tillmann and Hewavitharana, 2013; Mújdricza-Maydt et al., 2013].

6.5.3 Performance of the MaxEnt Model

The final MaxEnt-based system is constructed as follows (the procedure is identical for both English-French and English-Spanish). Each test document is processed independently from the others, in a leaving-one-out fashion. We use 80% of the sure parallel sentence pairs from the other 23 books for `auto en-fr` (resp. 16 books for `auto en-es`) corpus as positive examples; we also include the 1:1 links computed by BMA in the left-out corpus. For every pair of parallel sentences E_i and F_j, we randomly sample 3 target (source) sentences other than F_j (resp. E_i) to construct a negative pair with E_i (resp. F_j). Each positive example thus gives rise to 6 negative ones. We train a distinct MaxEnt model for each test book, using 80% of these book-specific data, again using 20% of examples as a held-out data set. Only the feature families 1 to 5 are included in the model, yielding an average test accuracy of 0.988. The ME+DP and ME+gr procedures are then finally used to complete BMA's links according respectively to algorithms 2 and 1. Results are summarized in Table 6.5 (see also Tables C.2 and C.3 in the Appendix C for a full listing). For corpora containing multiple books, the average, minimum and maximum scores are reported.

Our system obtains the best overall results for the manually aligned `manual en-fr` corpus. All the average differences between ME+DP and the other algorithms are significant at the 0.01 level, except for Gargantua and GMA, where the difference is only significant at the 0.05 level. On the large approximate reference sets, ME+DP achieves comparable results with Gargantua and GMA, slightly worse than Yasa. Comparing greedy with DP search, the mean performance on the `manual en-fr` has been increased by 3 points in F-measure and the differences on the two large corpora, even though they are only approximations, are also important. Surprisingly, this does not reflect on BAF, where the heuristic search is

		GMA	Hun	Garg	Yasa	ME+gr	ME+DP
BAF		61.4	71.2	65.6	75.7	76.3	66.5
`manual en-fr`	min	53.5	54.3	51.7	59.9	61.4	51.0
	max	92.8	92.6	97.1	95.6	95.3	98.0
	mean	79.6	74.5	80.2	79.1	78.3	**81.5**
`auto en-fr`	min	62.1	56.6	56.4	62.3	56.7	57.7
	max	99.5	99.5	98.1	98.8	97.5	97.9
	mean	88.7	87.9	88.7	89.6	85.7	88.9
`auto en-es`	min	60.3	43.7	58.4	64.3	60.4	65.2
	max	96.5	96.4	96.8	100	97.7	98.0
	mean	82.8	81.0	82.6	84.6	80.5	82.7

Table 6.5: Performance of the MaxEnt approach with greedy search (ME+gr) and dynamic programming search (ME+DP) and of baseline alignment tools, using link-level F-measure.

better than the DP algorithm - again, this might be because of the peculiar trait of BAF, which contains on average larger gaps than the other books (and crucially has an average gap size greater than 4 in one dimension).

We finally performed an error analysis on the manual alignment data set (`manual en-fr`). Table 6.6 lists the link types in the reference, along with the numbers of reference links that greedy search or DP fails to find.

Link type	in Ref.	ME+gr	ME+DP
0:1	20	13	18
1:0	21	12	18
1:1	1364	68	105
1:2	179	60	36
1:3	32	17	9
2:1	96	54	32
2:2	24	22	19
others	27	22	15
total	1,763	268	252

Table 6.6: Analyses of the errors of greedy search (ME+gr) and DP search (ME+DP) by link type, relative to the number of reference links (in Ref.), for the `manual en-fr` corpus. Only the link types occurring more than 5 times are reported. This filters out 27 links out of 1,790.

The numbers in Table 6.6 suggest that null links remain difficult, especially for the DP algorithm, reflecting the fact that estimating the scores of these links is a tricky issue. This problem arises for all systems whose search is based on DP: for instance, Yasa makes

a comparable number of errors for null links (16 errors for type 0:1, 17 for type 1:0), Hunalign's results are worse (20 errors for type 0:1, 19 for type 1:0), while Gargantua does not return any null link at all. DP tends to be more precise for larger blocks such as 1:2 or 2:1. Table 6.7 illustrates this property of DP search: this excerpt from Jean-Jacques Rousseau's *Les Confessions* is difficult because of the presence of consecutive 1-to-many links. ME+DP is the only algorithm which correctly aligns the full passage.

en_1	My mother had a defence more powerful even than her virtue; she tenderly loved my father, and conjured him to return; his inclination seconding his request, he gave up every prospect of emolument, and hastened to Geneva.	Ma mère avait plus que la vertu pour s'en défendre; elle aimait tendrement son mari. Elle le pressa de revenir : il quitta tout, et revint.	fr_1 fr_2
en_2	I was the unfortunate fruit of this return, being born ten months after, in a very weakly and infirm state; my birth cost my mother her life, and was the first of my misfortunes.	Je fus le triste fruit de ce retour. Dix mois après, je naquis infirme et malade. Je coûtai la vie à ma mère, et ma naissance fut le premier de mes malheurs.	fr_3 fr_4 fr_5
en_3	I am ignorant how my father supported her loss at that time, but I know he was ever after inconsolable.	Je n'ai pas su comment mon père supporta cette perte, mais je sais qu'il ne s'en consola jamais.	fr_6

Table 6.7: A passage of a reference alignment Jean-Jacques Rousseau's *Les confessions*. MaxEnt with DP finds all three links.

The gap size also has an impact on the performance of DP. Since in DP-search, we constrain alignment links to contain at most 4 sentences on each side, if at least one side of an actual alignment link exceeds this limit, our algorithm is not guaranteed to find the correct solution. Table 6.8 displays the average gap size[27] inside each book of the `manual en-fr` corpus, along with the F-score of greedy search and DP search.

	Ave. gap size	ME+gr	ME+DP
Du Côté de chez Swann	2.62×2.54	89.4	93.3
Emma	10.25×6.75	61.4	51.0
Jane Eyre	4×4.8	67.4	78.9
La Faute de l'Abbé Mouret	1.85×2.79	95.3	98.0
Les Confessions	2.89×4.7	67.8	74.0
Les Travailleurs de la Mer	3.37×3.74	80.8	85.3
The Last of the Mohicans	2.14×3.07	85.8	90.1

Table 6.8: Gap size and performance of MaxEnt on `manual en-fr`.

We can see that DP works better when gap sizes are smaller than 4 on each side. When this is not the case, the results tend to decrease significantly, as for instance for Jane

[27] A $N \times M$ gap contains N sentences on the source side and M sentences on the target side.

Austen's *Emma*. Greedy search, while generally outperformed by DP, is significantly better for this book. This outlines the need to also improve the anchor detection algorithm in our future work, in order to make sure that gaps are both as correct and as small as possible.

6.5.4 Performance of the 2D CRF Model

Learning corpus

The training of the 2D CRF model requires full reference alignments. This is in contrast with MaxEnt, which only requires independent instances of parallel and non-parallel sentences. We have used the reference sentence alignments presented in Chapter 5. The training corpus contains alignments for four books: "Alice's Adventures in Wonderland" (L. Carroll), "Candide" (Voltaire), "Vingt Mille Lieues sous les Mers", and "Voyage au Centre de la Terre" (both by J. Verne). Table 6.9 displays the statistics of the training corpus.

Book	# Links	# Sent_EN	# Sent_FR
Alice's Adventures in Wonderland	746	836	941
Candide	1,230	1,524	1,346
Vingt Mille Lieues sous les Mers	778	820	781
Voyage au Centre de la Terre	714	821	754
Total	3,468	4,001	3,822

Table 6.9: The training corpus of the 2D CRF model.

We have to convert the training corpus into a training set. A training instance is a fully observed sentence-level alignment matrix. In order to make train conditions as close as possible to test conditions, each fully aligned book was segmented into sub-blocks, again using high confidence 1 : 1 links computed by BMA as anchor points. Each sub-block, annotated with reference alignments, is then turned into one training instance. This strategy has the additional benefit to greatly reduce the total number of prediction variables, hence make the training less memory consuming. Besides, the training can enjoy better parallelization. There is potentially another advantage of using smaller training instances. Since our model contains one predictive variable for each pair of source-target sentences, there are roughly quadratically many negative examples, and linearly many positive ones. This data imbalance problem becomes more severe as the size of the prediction matrix grows larger. Using smaller training instances helps alleviate this problem.

Using this strategy, we obtain 450 fully observed alignment matrices. We use 360 for the training set, 90 as the development set. Among the 7,095 labeled sentence pairs, approximately 77% are negative.

For the test, we use the fully aligned novel "De la Terre à la Lune" in the BAF corpus and the `manual en-fr` corpus composed of 7 partial alignments of literary bitexts.

Results of the 2D CRF Model

We evaluate alignment results at two levels of granularity: the alignment level and the sentence level. At the alignment level, a link in the output alignment is considered correct if exactly the same link is also in the reference alignment. At the sentence level, we decompose a $m{:}n$ type link in the reference alignment into $m \times n$ sentence pairs, all considered as correct. The same decomposition applies to computed links. We summarize precision and recall ratios into F-scores.

Since the 2D CRF model is intrinsically trained to optimize **sentence-level** metrics, we first look at its sentence-level performance, summarized in Table 6.10. For the sake of comparison, we also display the performance of six other state-of-the-art aligners: GMA, BMA, Hunalign, Garg (as shorthand for Gargantua), Yasa, MaxEnt with DP. The CRF model achieves great improvements over BMA and Hunalign. Its average score on the `manual en-fr` corpus is slightly inferior to other systems, but it obtains the second best F-measure on the large bi-text "De la Terre à la Lune". We note that Yasa, perhaps the most lightweight tool, is very robust with respect to the sentence-level measure.

	Sentence level F-score						
	GMA	BMA	Hunalign	Garg	Yasa	MaxEnt	CRF
De la Terre à la Lune (BAF)	72.9	77.3	81.9	77.3	86.2	76.6	84.0
Du Côté de chez Swann	95.4	88.9	89.4	95.0	95.2	96.0	94.3
Emma	73.8	52.1	62.8	61.2	73.8	71.2	69.4
Jane Eyre	88.0	54.6	59.4	84.2	82.5	88.0	77.2
La Faute de l'Abbé Mouret	94.8	83.8	82.8	98.7	97.7	98.9	90.8
Les Confessions	82.8	49.9	48.5	80.5	82.8	86.1	76.6
Les Travailleurs de la Mer	87.8	79.6	78.8	91.5	90.4	91.9	89.1
The Last of the Mohicans	94.9	76.0	77.0	95.6	94.5	95.0	91.1
Average on `manual en-fr`	88.2	69.3	71.2	86.7	88.1	89.6	84.1

Table 6.10: Sentence level F-scores of the 2D CRF method on the test corpus, compared with state-of-the-art methods.

The first decoding step of both MaxEnt and CRF uses a subset of BMA's results as anchors to segment the bi-text space. Table 6.11 compares in more detail the performance of these three methods. Yu et al. [2012b]; Lamraoui and Langlais [2013] have reported that BMA usually delivers very high precision 1:1 links. We observe the 2D CRF model preserves a high sentence level precision, and greatly increases the recall. Thus, the 2D CRF model manages to extract true positive sentence pairs from the gaps defined by BMA's links with a very high accuracy. The behavior of MaxEnt varies on different corpora. On the `manual en-fr` corpus, while it slightly decreases the precision, it obtains the best recall, leading to the best overall F-score. However, on "De la Terre à la Lune", its precision is too low

compared to BMA and CRF, thus its F-score is worse.

	BMA			MaxEnt			CRF		
	P	R	F	P	R	F	P	R	F
De la Terre à la Lune (BAF)	97.2	64.1	77.3	72.0	81.8	76.6	95.5	74.9	84.0
Du Côté de chez Swann	99.5	80.3	88.9	97.1	94.9	96.0	96.3	92.5	94.3
Emma	89.8	36.7	52.1	62.8	82.3	71.2	76.1	63.7	69.4
Jane Eyre	93.7	38.5	54.6	86.7	89.3	88.0	86.6	69.6	77.2
La Faute de l'Abbé Mouret	99.5	72.3	83.8	98.9	98.9	98.9	98.2	84.5	90.8
Les Confessions	98.4	33.4	49.9	89.3	83.2	86.1	92.6	65.4	76.6
Les Travailleurs de la Mer	97.7	67.2	79.6	90.8	93.0	91.9	97.1	82.2	89.1
The Last of the Mohicans	98.7	61.8	76.0	94.2	95.8	95.0	97.1	85.7	91.1
Average on `manual en-fr`	96.8	55.7	69.3	88.5	91.1	89.6	92.0	77.7	84.1

Table 6.11: The comparison of BMA, MaxEnt and the 2D CRF model, using sentence-level measures. P stands for Precision, R is Recall, and F is F-score.

	Alignment level F-score		
	BMA	MaxEnt	CRF
De la Terre à la Lune (BAF)	73.6	66.5	73.3
Du Côté de chez Swann	91.5	93.3	90.9
Emma	57.4	51.0	55.4
Jane Eyre	61.1	78.9	63.2
La Faute de l'Abbé Mouret	88.4	98.0	82.8
Les Confessions	59.6	74.0	58.1
Les Travailleurs de la Mer	83.4	85.3	83.0
The Last of the Mohicans	82.7	90.1	84.3
Average on `manual en-fr`	74.9	81.5	74.0

Table 6.12: Link level F-scores of the 2D CRF model, compared with BMA and MaxEnt.

The link level F-scores of the CRF model are in Table 6.12.[28] The CRF achieves comparable alignment level F-scores to BMA on both sub-corpora. Although their average scores on `manual en-fr` are worse than MaxEnt, they outperform it considerably on those more difficult bitexts: "De la Terre à la Lune" and "Emma". In our opinion, this calls for further analyses for the *deployment* of alignment methods: for sentence alignment, it might be beneficial to investigate which types of methods tend to perform well for which types of

[28] We only show BMA, MaxEnt and CRF in this table, since MaxEnt obtained the best average alignment F-score on the `manual en-fr` corpus.

bitexts, identify indicative characteristics (of methods and bitexts), and deduce operational guidelines.[29]

Table 6.11 and Table 6.12 together show that, while the 2D CRF model obtains much higher sentence level F-scores than BMA (approximately 15 points on average on `manual en-fr`), their alignment level F-scores are actually comparable. In other words, the CRF does find more true positive sentence pairs, but not all of them contribute to form true links. Take for instance the 2:2 link $[14, 15; 24, 25]$ in Figure 6.5. To correctly recover this link, it is necessary to find at least three among the four cells. Even though the CRF finds one cell $[15; 25]$, this only yields a wrong 1:1 link, which, for the alignment level F-score metric, is no better than not finding any pair. While this imbalance between the alignment level and sentence level F-scores can seem surprising, it is by no means uncommon. In fact, this phenomenon was the reason that sentence-level F-score was proposed as an evaluation metric for sentence alignment in [Langlais et al., 1998a]. Nonetheless, this reinforces our belief that the deployment strategy of alignment methods, as well as evaluation metrics, need further study.

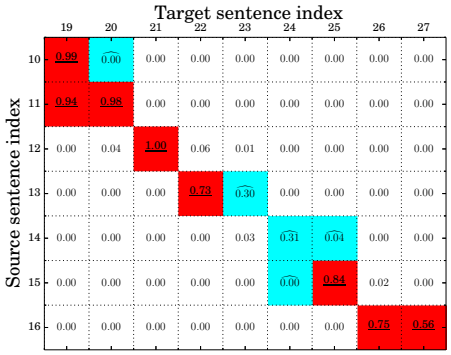


Figure 6.5: An alignment prediction matrix for a passage of "Les Confessions".

Analyses of Results

Error distribution by link type To better understand the behavior of the 2D CRF model, we perform an error analysis of its results on the `manual en-fr` corpus, with respect to link types. The corresponding statistics are in Table 6.13. We compare CRF with the MaxEnt approach, which gives the best average score on this corpus.

[29] This is in line with the views of Deng et al. [2007] and Lamraoui and Langlais [2013], who suggested to model sentence alignment as part of the target application, so that it can benefit the optimization conducted toward the task.

Link type	in Ref.	Error MaxEnt	Error CRF
0:1	20	18	15
1:0	21	18	15
1:1	1,366	105	64
1:2	179	36	**98**
1:3	32	9	**29**
2:1	96	32	**54**
2:2	24	19	20
others	27	15	26
Total	1,765	252	321

Table 6.13: Analyses of the errors of the MaxEnt and the CRF by link type, relative to the number of reference links (in Ref.), for the `manual en-fr` corpus. For example, 20 0:1 links are in the reference, and MaxEnt missed 18 of them. Only the link types occurring more than 5 times are reported. This filters out 27 links out of 1,792.

Compared to the MaxEnt method, CRF has a higher recall on null and 1:1 links. Its main weakness lies in the prediction of $1:n$ and $n:1$ links. After a closer study of the erroneous instances, we find a common pattern of error: when predicting a m:n link with $m * n > 1$ (that is, a 1-to-many or many-to-many link), the CRF often correctly labels some sentence pairs as positive, while leaving others as negative. Figure 6.5 displays an alignment prediction matrix for a passage of Jean-Jacques Rousseau's "Les Confessions". The corresponding text (correctly aligned) is displayed in Table 6.14. The CRF fails to predict the 1:2 link $(13; 22, 23)$, only labelling $(13; 22)$ as positive; nor does it find the 2:2 link $(14, 15; 24, 25)$.

The failures of the 2D CRF model on 1-to-many and many-to-many links makes it necessary to study edge potentials. One of the reasons of using a CRF model is its ability to encode the dependencies between neighboring links, with which we expect to better predict non $1:1$ links. An obvious direction to investigate is to add more edge features. Current edge features (families 6, 7 and 8) are quite general. It might be helpful to add features that encode finer level clues to edge potentials, e.g. word alignment information.

Besides features of edge potentials, it might also be possible to consider other alignment matrix decoding algorithms. Compared to the 2D CRF, MaxEnt has the advantage of directly scoring alignment-level links, rather than doing it obliquely through sentence-level ones. This is also possible in the 2D CRF model, since LBP can readily compute marginals over edges, or even larger factors. We might use such marginals to improve our post-processing routines.

Null sentences Another motivation for the 2D CRF model is that it provides a mechanism where null and non-null links are handled coherently. We summarize its performance

en_{10}	My mother's circumstances were more affluent; she was daughter of a Mons.	Ma mère, fille du ministre Bernard, était plus riche: elle avait de la sagesse et de la beauté.	fr_{19}
en_{11}	Bernard, minister, and possessed a considerable share of modesty and beauty; indeed, my father found some difficulty in obtaining her hand.	Ce n'était pas sans peine que mon père l'avait obtenue.	fr_{20}
en_{12}	The affection they entertained for each other was almost as early as their existence; at eight or nine years old they walked together every evening on the banks of the Treille, and before they were ten, could not support the idea of separation.	Leurs amours avaient commencé presque avec leur vie; dès l'âge de huit à neuf ans ils se promenaient ensemble tous les soirs sur la Treille; à dix ans ils ne pouvaient plus se quitter.	fr_{21}
en_{13}	A natural sympathy of soul confined those sentiments of predilection which habit at first produced; born with minds susceptible of the most exquisite sensibility and tenderness, it was only necessary to encounter similar dispositions; that moment fortunately presented itself, and each surrendered a willing heart.	La sympathie, l'accord des âmes, affermit en eux le sentiment qu'avait produit l'habitude.	fr_{22}
		Tous deux, nés tendres et sensibles, n'attendaient que le moment de trouver dans un autre la même disposition, ou plutôt ce moment les attendait eux-mêmes, et chacun d'eux jeta son coeur dans le premier qui s'ouvrit pour le recevoir.	fr_{23}
en_{14}	The obstacles that opposed served only to give a decree of vivacity to their affection, and the young lover, not being able to obtain his mistress, was overwhelmed with sorrow and despair.	Le sort, qui semblait contrarier leur passion, ne fit que l'animer .	fr_{24}
		Le jeune amant ne pouvant obtenir sa maîtresse se consumait de douleur: elle lui conseilla de voyager pour l'oublier .	fr_{25}
en_{15}	She advised him to travel – to forget her.		
en_{16}	He consented – he travelled, but returned more passionate than ever, and had the happiness to find her equally constant, equally tender.	Il voyagea sans fruit, et revint plus amoureux que jamais.	fr_{26}
		Il retrouva celle qu'il aimait tendre et fidèle.	fr_{27}

Table 6.14: The correct alignment of a passage of Jean-Jacques Rousseau's "Les Confessions", corresponding to the alignment prediction matrix in Figure 6.5.

for null sentences in Table 6.15, again, comparing it with the MaxEnt method.

Although the 2D CRF model incorrectly labels many sentences as unaligned, it is indeed able to find the majority of true null sentences, except for "The Last of the Mohicans". This is where our model seems to be improving, especially when compared to MaxEnt. It is interesting to observe that the two methods show very different tendencies on null links: MaxEnt with DP seems to be enforcing the coverage, and generates less null links, while the 2D CRF greatly over-generates null links. We hypothesize that this is due to the data imbalance problem in training: the majority (77%) of training instances of the CRF model are negative ones. This is an interesting question for future studies.

	#Null (in Ref.)	2D CRF		MaxEnt	
		#Null (in Hyp.)	#Correct	#Null (in Hyp.)	#Correct
De la Terre à la Lune (BAF)	714	1,311	672	150	91
Du Côté de chez Swann	9	27	8	5	3
Emma	41	85	28	2	2
Jane Eyre	10	77	7	0	0
La Faute de l'Abbé Mouret	2	52	2	1	1
Les Confessions	11	96	11	4	2
Les Travailleurs de la Mer	5	78	3	2	0
The Last of the Mohicans	12	37	3	2	2

Table 6.15: Performance of the 2D CRF model and the MaxEnt model on predicting null sentences. "#Null in Ref." is the number of unaligned sentences in the reference alignment; "#Null in Hyp." is the number of unaligned sentences in the hypothesis alignment computed by the model;"'#Correct" is the number of correctly predicted null sentences.

6.6 Conclusions

This chapter has presented a large-scale study of sentence alignment using a small corpus of reference alignments, and two large corpora containing dozens of coarsely aligned copyright-free novels for English-Spanish and English-French. We have shown that these coarse alignments, once refined, were good enough to compute approximate performance measures for the task at hand, and confirmed the general intuition that automatic sentence alignment for novels was still far from perfect; some translations appeared to be particularly difficult to align with the original text for all existing methods.

We have analyzed state-of-the-art sentence alignment methods, identified several recurring problems, and have accordingly proposed two methods for the full text sentence alignment task.

Borrowing ideas from previous studies on unsupervised and supervised sentence alignment, we have proposed and evaluated a new MaxEnt-based alignment algorithm, and showed that it performs better than several strong baselines, even if there remains a lot of room for improvement. There are still several obvious weaknesses in our current implementation that we intend to fix: first, it seems unnecessary to continue performing the second step of Moore's algorithm (which basically trains endogenously an IBM 1 Model) as the MaxEnt model also requires IBM 1 scores, which are computed on a large set of clean sentence alignments; second, the MaxEnt model is trained on isolated sentences and tested with blocks containing one or several sentences: it would be more natural to train the model in the same conditions as observed in testing. Another interesting problem is how to collect meaningful negative training examples for the MaxEnt model. Our current

approach gives in general easy negative pairs. It might be helpful to investigate more difficult examples, because the links encountered in the decoding, even true negatives, will mostly not be obviously wrong.

The two-dimensional Conditional Random Fields model is theoretically attractive, since it avoids several risky assumptions, computes posterior probabilities for all sentence alignment links, thereby explicitly representing null links, and warrants structured learning of parallelism scores. In the light of our experimental results and analyses, we conclude that there is clear room for improvement of our 2D CRF model. Currently, while the model is effective at identifying true 1:1 links with better recall than BMA's, its performance as measured by alignment level metric still needs to be improved. As perspectives for the 2D CRF model, we would like to study the following improvements:

- enhance edge features: current edge features do not seem to be strong enough to balance our rich set of node features. Including features informed with simple word alignment information, such as fertilities and linked regions, seems an obvious way to go;
- add node features that encode the decisions of other systems, e.g. BMA;
- explore ways to simulate a DP process using marginals of edges or larger factors, which might help improve our alignment matrix decoding algorithm.

In the long term, we would like to study ways to characterize tasks and alignment methods, such that it is possible to choose adequate alignment algorithms for specific task requirements.

Sentence alignment typically lies at the beginning of the bitext alignment pipeline. Hence, high-confidence sentence alignment would be very helpful for finer-grained alignments, and for downstream applications. We have made a substantial effort to design such systems, but there is still much room for improvement, especially for difficult bitexts such as literary ones. We can, on the one hand, continue to enhance the methods; on the other hand, employ confidence measures (CMs) to deal with imperfect automatic alignments, both at sentential and sub-sentential levels. Such CMs constitute our last part of contribution, and are presented in the next chapter.

Chapter 7

Confidence Measures for Bitext Alignment

Both sentential and sub-sentential bitext alignments are employed in many modern Natural Language Processing (NLP) applications. In general, system performance becomes better as alignment quality is improved [Lambert et al., 2005]. Typically, applications use automatic aligners to extract large amounts of alignments, which in most cases (if not all) contain noise. Noise can have more or less significant impacts on the performance. In some applications, a relatively high portion of noise can be tolerated, e.g., Goutte et al. [2012] showed that the translation quality of phrase-based SMT (as measured by BLEU and METEOR) is robust to noise levels of $\approx 30\%$ in sentence alignments. The relationship between the performance of SMT (as measured by BLEU) and word-level alignments (as measured by AER and/or F-score) has been studied in several works [Fraser and Marcu, 2007; Ganchev et al., 2008]. While no clear, quantitatively-defined relation has been established, various heuristics have been proposed to generate word alignments that lead to better translation quality. For some other applications, alignment quality influences the system performance more directly, such as translator training [Simard et al., 1993b] (sentence alignment), bilingual lexica extraction [Smadja et al., 1996] (word alignment). This is pushed to an extreme in bilingual reading [Yvon et al., 2016], where the system relies on bitext alignments to guide the reading process of a user accomplishing a cognitive task. Here, alignments must be as reliable as possible. Hence, in many contexts, it is very useful to be able to evaluate the quality of bitext alignments.

To automatically assess the quality of bitext alignments requires Confidence Measures (CM). Confidence Estimation (CE) has been applied in many NLP applications [Gandrabur et al., 2006]. In particular, CE has been extensively studied in Automatic Speech Recognition (ASR) [Jiang, 2005; Seigel, 2013]. CE is also useful in Information Extraction [Culotta and McCallum, 2004]. In SMT, CE has mainly been applied to evaluate translation quality [Blatz et al., 2004]: given the input source sentence and the translation output of the

MT system, CE consists of evaluating the quality of the translation, without reference translations or manual interventions. Ueffing and Ney [2007] proposed to compute the posterior probability *of each word* in the machine translation model. Blatz et al. [2004]; Raybaud et al. [2009] formulate CE as a binary classification problem, and investigated both sentence-level and word-level machine translation confidence measures. Applications of CE in SMT include error detection, post-editing, interactive machine translation [Gandrabur and Foster, 2003; Ueffing and Ney, 2007; Xiong et al., 2010], N-best list reranking [Bach et al., 2011; Luong et al., 2014], etc.

Confidence estimation for bitext alignment is different from MT quality estimation (QE), although the two problems are related. At the sentence level, CE for alignment evaluates whether the two sides of the sentence alignment link are parallel, while QE attempts to assess the quality of the output *as a translation*. Thus, a critical concern of QE is whether the output is a reasonable sentence in the target language, leading to the dominant role played by language models in QE systems [Wisniewski et al., 2013]. In CE for alignment, this question does not exist. At the word level, QE systems have to decide whether a word is part of a good translation. A word can be a legitimate part of a translation sentence, yet not be clearly related to any unit of the source side. CE for alignment again concentrates on the relation between the two sides of a word link. In other words, CE pays less attention to how well a word goes along with its neighbours. [1] In fact, for bitext alignment CE, the notions of the source and the target sides can be forgotten.

There exist relatively few works on CE for bitext alignment. At the sentence level, we are unaware of works explicitly studying sentence alignment confidence estimation. The work of Barbu [2015] on translation memory cleaning is related. By "sentence alignment confidence estimation", we refer to the following problem: given a source sentence group and a target sentence group (that some alignment system returns), assess whether the two sides are really parallel. Huang [2009] proposed a measure which, given a parallel sentence pair, evaluates the quality of a word alignment as a whole.[2] Perhaps confusingly, this CM was named "sentence alignment confidence measure".[3] But this is not the problem we are addressing. At the sub-sentential level, Huang [2009] studied confidence measures for word alignment, using link-level posterior probabilities under IBM Model 1. These CMs were employed to selectively combine word alignments generated by different aligners. Many word-level QE measures combine word-to-word information. For instance, Raybaud et al. [2009] combined word-to-word mutual information (MI) into word-level translation confidence measures. Such measures can also be inspiring for word alignment CE.

In this chapter, we report our work on confidence estimation for bitext alignment. We begin with a formal definition of CE for bitext alignment, as well as evaluation methodologies for CE. We then detail the CMs that we investigate at three levels: sentence align-

[1] However, this does not prevent a CE system from making use of contextual information of involved words.

[2] Recall a word alignment is a set of word-level links.

[3] In our opinion, this would better be called "sentence-level word alignment confidence measure".

ment links, one-to-one word alignment links, and many-to-many word alignment links. We present the performance of these CMs, using the data we presented in Chapter 5. Finally we present future works and perspectives.

7.1 Confidence Estimation for Bitext Alignment: Definitions

Confidence Estimation consists in predicting the quality of outputs of a specific application system *without using any reference*. It can be understood as a generic rescoring task: CE assigns a new *confidence score* to each output. In a very general way, denoting the input space as $\mathcal{X}$, the output space as $\mathcal{Y}$, CE can then be defined as a generic scoring function:

$$\texttt{CONFIDENCE\_SCORE} : \mathcal{X} \times \mathcal{Y} \longrightarrow \mathbb{R}$$

Typical usages of CE fall into two categories: filtering and reranking. Both rely on confidence scores. For filtering, CE can be formalized as a classification problem, mostly binary: given an output instance y, the CE system needs to label it as *positive* (often labeled as 1) or *negative* (often labeled as -1), typically using thresholds:

$$\texttt{class}(y) = \begin{cases} 1 & \text{if } \texttt{CONFIDENCE\_SCORE}(x, y) \geq \delta \\ -1 & \text{if } \texttt{CONFIDENCE\_SCORE}(x, y) < \delta \end{cases}$$

where δ is a predefined numerical threshold. Negative instances are to be rejected. This is the case of, for example, utterance rejection in ASR [Gandrabur et al., 2006]. A typical reranking scenario is as follows: one or several application systems propose several candidate outputs $\{y_1, ..., y_k\} \subseteq \mathcal{Y}$ for one input $x \in \mathcal{X}$. These outputs might have been ranked by the application. A CE system assigns a confidence score to each output, which is then used to rerank the candidates, potentially changing the 1-best output. For instance, Luong et al. [2014] use confidence scores to rerank the N-best lists in MT. For bitext alignment, CE is mainly used as a filtering tool, i.e. to filter out unreliable alignment links.

7.1.1 Alignment-level and Link-level Confidence Estimation

CE for bitext alignment can be used at the level of alignments or at the level of links. Alignment-level CE consists in assessing the quality of an entire alignment, that is, a set of links. We illustrate this notion in the context of word alignment, and similar definitions can be made for sentence alignment. For a pair of parallel sentences $E = \mathbf{e}_1^I$, $F = \mathbf{f}_1^J$, and a general word alignment $\mathcal{Z} = \{z_{ij} = 1 : 1 \leq i \leq I, 1 \leq j \leq J\}$, an alignment-level CM is a function

$$\texttt{WORD\_ALIGN\_CM}(E, F, \mathcal{Z}) \longrightarrow \mathbb{R}$$

For instance, Huang [2009] proposed the following alignment-level CM:

$$C(\mathcal{Z}|E, F) = \sqrt{P_{s2t}(\mathcal{Z}|E, F) P_{t2s}(\mathcal{Z}|F, E)}$$

where the alignment-level posterior probabilities are computed based on IBM Model 1:

$$\begin{aligned} P_{s2t}(\mathcal{Z}|E,F) &= \frac{P(\mathcal{Z},F|E)}{\sum_{\mathcal{Z}'} P(\mathcal{Z}',F|E)} \\ &= \frac{\prod_{j=1}^{J} p(f_j|e_i, z_{ij}=1)}{\prod_{j=1}^{J}\sum_{i=1}^{I} p(f_j|e_i)} \end{aligned}$$

and $P_{t2s}(\mathcal{Z}|E,F)$ is computed similarly.[4] Huang [2009] then defined the alignment-level confidence score to be equal to $-\log C(\mathcal{Z}|E,F)$. For each sentence pair, this confidence score is used to select the best alignment $\hat{\mathcal{Z}}$ among those generated by three different aligners $\mathcal{Z}_1$, $\mathcal{Z}_2$ and $\mathcal{Z}_3$. With this method, Huang [2009] was able to improve the overall F-score of the corpus by 0.8 over the best single word aligner. Similar techniques can be applied to sentence alignment.

Link-level CE consists in assigning confidence scores to individual alignment links. Thus, for word alignment, link-level CMs can be defined as:

$$\texttt{WORD\_ALIGN\_LINK\_CM}(\texttt{E},\texttt{F},\texttt{l}) \longrightarrow \mathbb{R}$$

where $l \in \mathcal{Z}$ is a link. Compared to alignment-level CE, link-level CE seems more appropriate for bitext alignment: we can examine each link instead of an alignment as a whole, and such finer-grain selection/combination is more promising to generate better overall alignments. Thus, in this chapter, we focus on link-level confidence measures.[5]

7.1.2 Unsupervised and Supervised Confidence Measures

In some cases, we can directly infer confidence measures from the application system or other knowledge domain, without the need of creating specific training examples for the CE system. In this chapter, we refer to such measures as *unsupervised confidence measures*, and refer to those trained on specific corpora as *supervised confidence measures*.

Unsupervised CMs can come from knowledge external to the task. For example, for sentence alignment CE, a very naive CM would be to compute the length ratio between the two sides of the link, which is very unlikely to be large. They can also be inferred from the systems. Some systems provide ways to compute posterior probabilities of outputs given the input $p(y|x)$, which can be used directly as confidence measures. The word alignment confidence measure proposed in [Huang, 2009] is an instance of unsupervised CMs.

[4] There is an undiscussed technical detail in [Huang, 2009]: this CM only works when $\mathcal{Z}$ contains exclusively 1:1 and null links. Otherwise, we cannot compute both P_{s2t} and P_{t2s}, since IBM Model 1 is a directional alignment model, as shown in the formulas. Exactly how to deal with non bijective alignments is not explained in this paper.

[5] This might not be the case for CE of other NLP applications. For example, in MT, it can be very useful to obtain the confidence score of the translation of one sentence, which makes it possible to reject very bad results as a whole and to proceed directly to human translation.

Supervised CMs require supervision examples. When such resources are available, we can train a discriminative classifier. In probabilistic terms, this means that it is possible to estimate the probability $p(1|x, y)$. Machine learning techniques such as logistic regression provide principled ways to encode various types of knowledge. In this chapter, we stick to the logistic regression model, for its simplicity and the good interpretability. We mainly consider binary classification tasks, thus a training paradigm for word alignment CE is:

$$p(1|E, F, z_{ij}) = \frac{1}{1 + \exp(-\theta^\top \mathbf{f}(E, F, z_{ij}))}$$

where $\mathbf{f}$ is the feature vector and θ is a trainable parameter vector.

Gandrabur et al. [2006] provided a detailed discussion on the advantages and disadvantages of unsupervised and supervised CMs. In that study, properly designed supervised CMs outperformed unsupervised ones, mainly due to their ability to incorporate sources of knowledge that might be absent in the unsupervised CMs. The most obvious limitation of using supervised CMs is the requirement of supervision examples, which might be difficult to obtain for some NLP applications. Our experiments on supervised CMs make use of the data presented in Chapter 5.

7.1.3 Evaluation Metrics of Confidence Measures

For a binary classification test set with reference labels, we call the test instances with positive reference labels *conditional positives* and denote this set as P_c, those with negative reference labels *conditional negatives* and denote as N_c. Now, a classifier would assign hypothetical labels to test instances. We denote the set of instances with positive hypothetical labels as P_h, the set of instances with negative hypothetical labels as N_h.

Since CMs are applied in binary classification settings, we can evaluate their effectiveness with standard binary classification metrics: Precision (P), Recall (R) and F-score (F), as in sentence alignment and word alignment. We rewrite their definitions below:

$$P = \frac{|P_h \cap P_c|}{|P_h|} \qquad R = \frac{|P_h \cap P_c|}{|P_c|} \qquad F = \frac{2PR}{P + R}$$

A very useful analyzing tool for CMs (or more generally, binary classifiers) is the Receiver Operating Characteristic (ROC) curve. We compute two metrics, False Positive Rate (FPR) and True Positive Rate (TPR):

$$FPR = \frac{|P_h \setminus P_c|}{|N_c|} \qquad TPR = \frac{|P_h \cap P_c|}{|P_c|}$$

where $P_h \setminus P_c$ is the set of instances that are member of P_h but not P_c, in other words, the false positives. TPR is the same with Recall. While the classifier changes, typically by varying the decision thresholds, both TPR and FPR would move accordingly. We can

then define the ROC space by putting FPR at the horizontal axe, TPR at the vertical axe. One point in the ROC space represents one classification result. For a confidence measure, varying the decision threshold leads to a ROC curve, which provides a very nice way to visualize performance.

We can derive several useful properties of the ROC curve (assuming all confidence scores and decision thresholds lie in the interval $[0, 1]$). These properties are illustrated in Figure 7.1.

- $0 \leq FPR \leq 1$, because $P_h - P_c \subseteq N_c$ ($P_h \subseteq P_c \cup N_c$); $0 \leq TPR \leq 1$, because $P_h \cap P_c \subseteq P_c$;
- With threshold 0, we have $P_h = P_c \cup N_c$, thus $FPR = 1, TPR = 1$, corresponding to the upper right corner point of the ROC space;
- With threshold 1, we have $P_h = \emptyset$, thus $FPR = 0, TPR = 0$, corresponding to the lower left corner point of the ROC space;
- Suppose a perfect threshold leads to $P_h = P_c$ (that is, a classification without any mistake), then $FPR = 0, TPR = 1$, corresponding to the upper left corner point of the ROC space;
- A classification by completely random guess would lead to a point at $(0.5, 0.5)$. Thus, the straight line from $(0, 0)$ to $(1, 1)$ through $(0.5, 0.5)$ corresponds to random classification. This is illustrated by the red dashed line in Figure 7.1.
- A classifier better than random guess would always lie above the red dashed line, that is, $TPR > FPR$. This is illustrated by the blue curve in Figure 7.1.
- A classifier with strong dicriminative ability would consistently assign small confidence scores to conditional negatives (members of N_c), and much larger scores to conditional positives (members of P_c). If this is the case, then most thresholds would lead to $P_h \approx P_c$, that is, small FPR (≈ 0) and large TPR (≈ 1) values. Hence, the ROC curve of such a CM is close to the path $(0, 0) - (0, 1) - (1, 1)$. This is illustrated by the green curve in Figure 7.1.

Based on the ROC space, the area under the curve (AUC) provides a quantitative summary of performance. As discussed above, a good classifier's ROC curve would approach the path $(0, 0) - (0, 1) - (1, 1)$. Thus, intuitively, the larger the AUC is, the better. It can be shown that the AUC of a classifier represents the probability that the classifier assigns higher score to a randomly chosen positive instance than to a randomly chosen negative instance [Fawcett, 2006]. Thus, the AUC can be used to compare classifiers, in particular, confidence measures.

ROC curves have many other nice properties, such as the insensitivity to changes in class distribution. Here, we focus on their usage as tools to compare CMs. The reader can refer to [Fawcett, 2006] for a more detailed and thorough discussion.

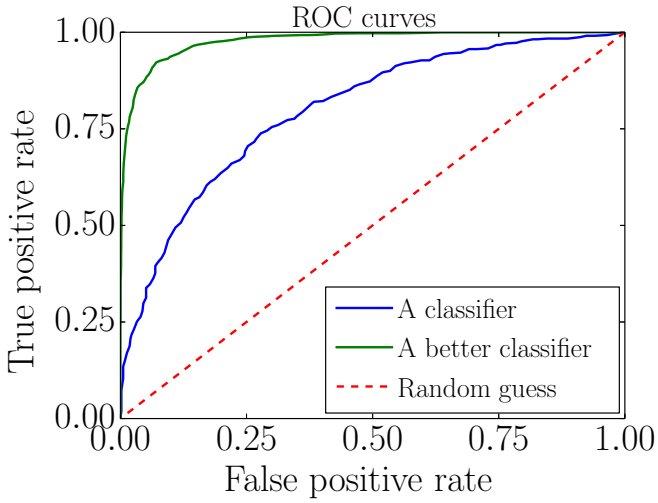


Figure 7.1: An illustration of ROC curves.

7.2 Confidence Measures for Sentence Alignment

For sentence alignment, we study the following confidence estimation problem: a (perhaps unknown) system aligns a pair of parallel texts $\mathbf{E}_1^M$ and $\mathbf{F}_1^N$ and returns an alignment $\mathcal{A} = (l_1, ..., l_k, ..., l_K)$ composed of K links, the CE system needs to estimate the confidence that each individual link l_k is correct.

7.2.1 Unsupervised Confidence Measures

When no manual sentence alignments are available, we have to use unsupervised CMs, which might rely on other types of resources.

Length difference ratio A very intuitive idea for evaluating the parallelism of sentence alignment links is by looking at the lengths of the two sides. Clearly this is not a naive indicator: if the lengths of two sides are too different, then the link is very likely a bad one. On the contrary, if the lengths of the two sides are comparable, we cannot tell whether they are parallel or not. So lengths are a very simple predictor that can be used as a baseline confidence measure. For a non-null sentence alignment link $[\mathbf{E}_o^q; \mathbf{F}_r^t]$, our first confidence measure is:

$$\texttt{CM}_{\texttt{sa}-1}(\mathbf{E}_o^q, \mathbf{F}_r^t) = 1 - \frac{|\texttt{len\_char}(\mathbf{E}_o^q) - \texttt{len\_char}(\mathbf{F}_r^t)|}{\max(\texttt{len\_char}(\mathbf{E}_o^q), \texttt{len\_char}(\mathbf{F}_r^t))} \tag{7.1}$$

where $\texttt{len\_char}(\cdot)$ is a function returning the number of non-space characters of its argument. The value of this CM is guaranteed to lie in $[0, 1]$. Larger values are better.

IBM Model 1 scores Brown et al. [1993] proposed IBM Models for word alignment. These are generative models in a noisy-channel setting:

$$P(E|F) = \sum_{\mathcal{A}} P(\mathcal{A}, E|F)$$

where F is a French sentence $F = \mathbf{f}_1^J$, E is an English sentence $E = \mathbf{e}_1^I$. For IBM Model 1, this quantity can be computed efficiently:

$$P_{IBM1}(F|E) \propto \prod_{j=1}^{J} \sum_{i=0}^{I} t(f_j|e_i)$$

where the parameters $t(f_j|e_i)$ can be estimated on external parallel corpora. Intuitively, $P_{IBM1}(F|E)$ models the probability of translating E into F. If E and F are parallel, we expect this value to be high. We can compute $P_{IBM1}(E|F)$ in the same fashion. There is a small problem in directly using $P_{IBM1}(F|E)$, in that longer sentences tend to receive too low scores. To remedy this problem, we slightly reformulate the computation:

$$S_{IBM1}(F|E) = \left(\prod_{j=1}^{J} \frac{1}{I+1} \sum_{i=0}^{I} t(f_j|e_i) \right)^{\frac{1}{J}}$$

$$S_{IBM1}(E|F) = \left(\prod_{i=1}^{I} \frac{1}{J+1} \sum_{j=0}^{J} t(e_i|f_j) \right)^{\frac{1}{I}}$$

For a non-null link $[\mathbf{E}_o^q; \mathbf{F}_r^t]$, a new confidence measure is:

$$\texttt{CM}_{\texttt{sa}-2}(\mathbf{E}_o^q, \mathbf{F}_r^t) = \sqrt{S_{IBM1}(\mathbf{F}_r^t|\mathbf{E}_o^q) \times S_{IBM1}(\mathbf{E}_o^q|\mathbf{F}_r^t)} \tag{7.2}$$

where we view both $\mathbf{E}_o^q$ and $\mathbf{F}_r^t$ as word sequences. In general, larger values of $\texttt{CM}_{\texttt{sa}-2}(\cdot)$ are better. This score lies in the interval $[0, 1]$, and tends to be very close to 0 (typically at the scales of 1e$-$7 $\sim$ 1e$-$2 in our experiments).

Bilingual word embeddings Recent years have witnessed successful applications of distributional representations of words in many NLP tasks. Mikolov et al. [2013] reported that word embeddings can capture semantic relationships between words. Thus, one can use distances between word vectors as a measure of difference between words.

Most works on word embeddings/vectors focus on monolingual settings, mainly capturing the cooccurrence relation between words in the same language. However, in some cases, it is useful to be able to map words of two different languages to one unique vector space. This encourages the development of bilingual word embeddings. Here we try to apply bilingual word embeddings in sentence alignment confidence estimation.

We use the BilBOWA package [Gouws et al., 2015] to train bilingual word embeddings, which requires only two monolingual corpora (respectively for the languages $\mathfrak{E}$ and $\mathfrak{F}$) and one (small) corpus of parallel sentences in $\mathfrak{E}$ and $\mathfrak{F}$. Other methods for training such bilingual word embeddings have been proposed in, e.g. [Zou et al., 2013; Faruqui and Dyer, 2014]. Compared to these methods, BilBOWA does not require pre-computed word alignments, and is very efficient. Briefly, the objective of BilBOWA is composed of two mono-lingual skip-gram objectives, optimized on mono-lingual corpora to ensure good modeling of each languages, and a cross-lingual objective, which is optimized on a parallel corpus to capture cross-lingual correspondences and plays the role of regularizer. Gouws et al. [2015] showed that BilBOWA bilingual word embeddings obtained good performance on several NLP tasks.

For sentence alignment CE, we represent a word sequence by the average of individual word vectors:

$$\texttt{SENT\_VEC}(\mathbf{e}_1^I) = \frac{1}{I}\sum_{i=1}^{I}\texttt{WORD\_VEC}(e_i)$$

where $\texttt{WORD\_VEC}(\cdot)$ is the embedding of the word. We measure the similarity between two sentences using the cosine similarity. Thus, our second confidence measure for sentence alignment is

$$\texttt{CM}_{\texttt{sa}-3}(\mathbf{E}_o^q, \mathbf{F}_r^t) = \frac{1}{2}\left(1 + \frac{\texttt{SENT\_VEC}(\mathbf{E}_o^q)^\top \texttt{SENT\_VEC}(\mathbf{F}_r^t)}{||\texttt{SENT\_VEC}(\mathbf{E}_o^q)|| \cdot ||\texttt{SENT\_VEC}(\mathbf{F}_r^t)||}\right) \tag{7.3}$$

The value of $\texttt{CM}_{\texttt{sa}-3}(\cdot)$ lies in the interval $[0, 1]$. Again larger confidence scores should be better.

7.2.2 Supervised Confidence Measures

When supervision data, i.e. manual sentence alignments, is available, we can use supervised learning techniques to obtain confidence measures combining various kinds of predictors. As discussed, we use logistic regression (LR) for this purpose.

We use the following set of features in the LR model:

- the length difference ratio, as the first unsupervised CM. We discretize the value into 10 binary indicator features.
- IBM Model 1 scores, i.e. the S_{IBM1} scores in Equation (7.2). We discretize each of the two scores into 10 binary indicator features.
- cosine similarity of the distributional representations of the two sides (as the CM_{sa-3} score), discretized into 10 binary features.
- the number of copies between the two sides, not including punctuations. We use five features: 1, 2, 3, 4, 5+.

– the number of cognates (not including copies) between the two sides. If two words share their first 4 characters, they are considered cognates. We use five features: 1, 2, 3, 4, and 5+.

– the word pair lexical features, one for each pair of words co-occurring at least once in a parallel sentence.

This set of features is close to those of § 6.5.2, except the one based on distributional representations of sentences.

7.3 Using Sentence Alignment Confidence Measures

7.3.1 Data

We use both the Reference Sentence Alignment (§ 5.1.1) and the Sentence Alignment Confidence Annotation (§ 5.1.2) datasets for evaluating confidence measures. The experiments are conducted on the English-French part of the two datasets.

For the Reference Sentence Alignment dataset, we collect all manual alignment links, as well as all the links from the reference alignment of the literary part of the BAF corpus. In total, this amount to 8,266 alignment links, which constitute the positive instance class. To better simulate real applications and obtain meaningful negative instances, we have run all aligners presented in Chapter 6 on all bitexts, and collect all wrong links. The negative class contains thus 4,233 instances. We refer to this data as the `cm-sa-literary` dataset.

The Sentence Alignment Confidence Annotation dataset contains 1,800 sentence pairs. Each pair is annotated by two annotators, using the five tags: *sure*, *partial*, *imperfect*, *wrong*, *undecidable*. This poses two interesting questions: (a) how to deal with sentence pairs that are assigned different tags? (b) which tags should be in the positive category, which in the negative? For the first question, we have decided to only consider sentence pairs to which two annotators give the same tag. In this way, the tag of each considered sentence pair is quite clear. Because of the high inter-annotator agreement ratio ($\kappa \approx 0.85$ in the lowest case) on this dataset, this only discards 139 pairs. It remains to classify the tags into two categories. This is easy for "sure" (positive) and "wrong" (negative). We ignore "undecidable", since there are only two such instances. For "imperfect", we consider it as positive, as it is common that the translation process can be somewhat loose. We consider "partial" as negative, since it corresponds to a common error pattern of aligners: a 1:m link with $m > 1$ is missed, and only a 1:1 link, a part of the 1:m link, is found. Under this classification, we collected 1,165 positive and 496 negative instances. We refer to this dataset as `cm-sa-subtitle`.

7.3.2 The Computation of Confidence Measures

The computation of some unsupervised CMs relies on external resources. We used MGiza [Gao and Vogel, 2008] to train IBM Model 1 translation tables, using the English-French version of the Europarl corpus, which contains approximately 2M parallel sentences, 56M English and 62M French words. Bilingual word embeddings were trained using the BilBOWA software, on 2M sentence pairs from the Open Subtitle corpus [Lison and Tiedemann, 2016].[6] Following Gouws et al. [2015], we set the dimension of word vectors to be 40.[7]

The supervised CM is trained on the parallel sentences from the corpus of literary bitext annotations of A. Farkas, in the same way with the MaxEnt model in § 6.5.3. The positive class of this training set contains 63,960 sentence pairs, and the negative class contains 127,920 pairs. The development set 16,000 positive and 32,000 negative pairs. We apply the ℓ_2 regularization with parameter 0.3 (obtained from experiences on the development set).

7.3.3 Performance of Sentence Alignment Confidence Measures

On the `cm-sa-literary` dataset, the performance of the three unsupervised CMs and the supervised CM are summarized in Figure 7.2 and Table 7.1. The performance of the CMs on the `cm-sa-subtitle` dataset is shown in Figure 7.3 and Table 7.2.

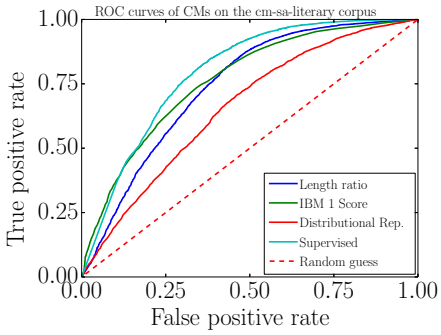


Figure 7.2: ROC curves of sentence alignment CMs on the `cm-sa-literary` dataset.

CM	AUC	Best F.
Length ratio	0.74	0.83
IBM 1 score	0.76	0.83
Distributional Rep.	0.66	0.80
Supervised	**0.80**	**0.85**

Table 7.1: AUCs and Best F-scores achieved by the CMs on the `cm-sa-literary` dataset.

We observe that the results on the two datasets are rather poor. Both datasets are difficult, though. In the first one (`cm-sa-literary`), the negative examples are wrong links produced by state-of-the-art aligners. The second one (`cm-sa-subtitle`) is also tricky,

[6] Available at `http://opus.lingfil.uu.se/`, originally from `http://www.opensubtitles.org/`.
[7] We have tried larger dimensions such as 100 and 300, but the change in performance was negligible.

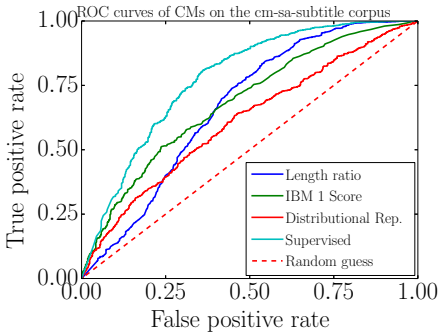


Figure 7.3: ROC curves of sentence alignment CMs on the `cm-sa-subtitle` dataset.

CM	AUC	Best F.
Length ratio	0.67	0.85
IBM 1 score	0.69	0.83
Distributional Rep.	0.61	0.82
Supervised	**0.78**	**0.86**

Table 7.2: AUCs and Best F-scores achieved by the CMs on the `cm-sa-subtitle` dataset.

because of the way we have grouped the classes: "imperfect" and "partial" links can be hard to classify.

An interesting finding is that the supervised CM performs consistently better than unsupervised ones, both for the AUC and the F-score. Note that the supervised CM is trained on examples different from the test material, for both datasets. For the `cm-sa-literary` dataset, both the supervision examples and the test examples are extracted from literary bitexts, but negative examples are constructed in very different ways. For the `cm-sa-subtitle` dataset, the test instances are movie subtitles, instead of sentences from bilingual fictions. Still, for both datasets, the supervised CM provides a very clear classification improvement over unsupervised CMs. This discrepancy between training and test examples is an important point for real confidence estimation applications, where test instances do not come with confidence labels, and it might be expensive to annotate large amounts of test instances to train an ideal classifier. For sentence alignment, we have shown that supervised confidence measures, even though not trained on examples exactly as the test ones, can still improve confidence estimation over unsupervised measures.

Tables 7.1 and 7.2 show that unsupervised confidence estimation for sentence alignment is a difficult task. The IBM 1 score barely outperforms the length difference ratio in terms of the AUC, which is a very naive baseline. Our attempt to use the distributional representation of sentences by employing bilingual word embeddings leads to worse results than the length difference ratio. However, many options remain to explore: other ways to produce bilingual word embeddings, other ways to combine word-level representations into sentence-level ones, etc.

7.4 Confidence Measures for One-to-one Word Alignment Links

Confidence estimation for one-to-one word alignment links studies the following problem: given a pair of parallel sentences $E = \mathbf{e}_1^I$ and $F = \mathbf{f}_1^J$, and a word alignment $\mathcal{Z} = \{z_{ij} = 1 : 1 \leq i \leq I, 1 \leq j \leq J\}$, evaluate the confidence score for each alignment link z_{ij}.

7.4.1 Unsupervised Confidence Measures

Mutual information Mutual Information (MI) measures the mutual dependence of two random variables. Raybaud et al. [2009] used MI to measure the confidence of each word for machine translation. We apply word-level MT to word alignment link confidence estimation. Following Raybaud et al. [2009], we compute the MI between a pair of words e and f by:

$$\begin{aligned} p(e) &= \frac{N_E(e)}{N} \\ p(e,f) &= \frac{N(e,f)}{N} \\ MI(e,f) &= p(e,f)\log_2\left(\frac{p(e,f)}{p(e)p(f)}\right) \end{aligned}$$

where $N_E(e)$ is the number of English sentences containing the word e, $N(e,f)$ is the number of sentence pairs where the English side contains e and the French side contains f. To avoid zero probabilities, we use the following smoothing technique, again following Raybaud et al. [2009]:

$$p'(e,f) \leftarrow \frac{p(e,f) + \alpha p(e)p(f)}{1+\alpha}$$

We randomly set $\alpha = 0.1$. The confidence measure is:

$$\texttt{CM}_{\texttt{wa}-\texttt{1}}(e_i, f_j, \mathbf{e}_1^I, \mathbf{f}_1^J) = p'(e_i,f_j)\log_2\left(\frac{p'(e_i,f_j)}{p(e_i)p(f_j)}\right) \tag{7.4}$$

Since this CM is the mutual information, it is nonnegative (0 if and only if e_i and f_j are independent) and symmetric.

Link posterior probability under IBM Model 1 A very intuitive word alignment link CM is its posterior probability under IBM Model 1. This CM has been proposed by Huang [2009]. In IBM Model 1, the posterior probability of a (directional) link is:

$$P_{IBM1}(a_j = i|\mathbf{e}_1^I, \mathbf{f}_1^J) = \frac{t(f_j|e_i)}{\sum_{i'=0}^{I} t(f_j|e_{i'})}$$

thus, the CM of the alignment link z_{ij} is

$$\texttt{CM}_{\texttt{wa}-\texttt{2}}(e_i, f_j, \mathbf{e}_1^I, \mathbf{f}_1^J) = \sqrt{P_{IBM1}(a_j = i|\mathbf{e}_1^I, \mathbf{f}_1^J) P_{IBM1}(b_i = j|\mathbf{e}_1^I, \mathbf{f}_1^J)} \tag{7.5}$$

where $a_j = i$ is a directional word alignment link, as well as $b_i = j$. The value of this CM lies in the interval $[0, 1]$. To compute this confidence measure requires a pre-trained translation table.

Link posterior probability under IBM Model 2 We can also efficiently compute word alignment link posterior probability under IBM Model 2:

$$P_{IBM2}(a_j = i|\mathbf{e}_1^I, \mathbf{f}_1^J) = \frac{a(i|j, I, J)t(f_j|e_i)}{\sum_{i'=0}^{I} a(i'|j, I, J)t(f_j|e_{i'})}$$

where $a(i|j, I, J)$ are alignment probabilities. This alignment table can be retrieved from the output of MGIZA. Compared to IBM Model 1, IBM Model 2 has the advantage of taking positional information into account, which would help in the situation of recurring words in the same sentence.

The corresponding CM of the alignment link z_{ij} is

$$\texttt{CM}_{\texttt{wa}-3}(e_i, f_j, \mathbf{e}_1^I, \mathbf{f}_1^J) = \sqrt{P_{IBM2}(a_j = i|\mathbf{e}_1^I, \mathbf{f}_1^J)P_{IBM2}(b_i = j|\mathbf{e}_1^I, \mathbf{f}_1^J)} \tag{7.6}$$

The value of this CM also lies in the interval $[0, 1]$. The computation of this CM requires a pre-trained translation table and an alignment table.

Bilingual word vector distance As discussed in § 7.2.1, one can use bilingual word embeddings to measure the correspondences of words in different languages. We use the cosine distance between BilBOWA word vectors as a confidence measure for word alignment links:

$$\texttt{CM}_{\texttt{wa}-4}(e_i, f_j, \mathbf{e}_1^I, \mathbf{f}_1^J) = \frac{1}{2}\left(1 + \frac{\texttt{WORD\_VEC}(e_i)^\top \texttt{WORD\_VEC}(f_j)}{||\texttt{WORD\_VEC}(e_i)|| \cdot ||\texttt{WORD\_VEC}(f_j)||}\right) \tag{7.7}$$

The value of this CM lies in the interval $[0, 1]$.

7.4.2 Supervised Confidence Measures

The supervised CM for word alignment links is a Logistic Regression (LR) model combing all four unsupervised CMs. For word alignment, we could add many other features, as presented in [Tomeh et al., 2014], where a careful feature engineering for such a LR model leads to the best AER (intrinsic evaluation) and translation system BLEU scores (extrinsic evaluation), beating very strong baselines such as PostCat [Graça et al., 2010] and the global CRF of Niehues and Vogel [2008]. However, we try to study whether a combination of simple CMs leads to a better supervised CM. Thus, we do not include features other than the unsupervised CMs.

Each unsupervised CM is discretized into 10 binary features. Hence, the supervised LR model contains 41 features.

7.5 Using Confidence Measures for One-to-one Word Alignment Links

7.5.1 Data

We test the proposed CMs on the 1-to-1 word alignment dataset presented in § 5.2.2. As in using sentence alignment confidence annotations, two problems as arise: (a) how to classify the four labels into the positive and negative categories; (b) how to deal with annotator disagreements. In this study, we set the most conservative scenario: we only take alignment links that all annotators agree on the labels, use "sure" links as positive instances and "wrong" links as negative instances. We have also discarded the French-Greek dataset as it is too small ($\approx$ 500 links) compared to others (all more than 2200 links). In this way, we collect 1,267 positive and 411 negative instances for English-French; 1,266 positive and 1,042 negative ones for English-Spanish; 754 positive and 606 negative ones for English-Greek; 370 positive and 1,003 negative instances for French-Spanish.

It should be noted that even though we only take "sure" and "wrong" links, this dataset remains difficult. Recall that all links in this dataset were picked from the intersection of two directional MGiza alignments. Hence the dataset does not contain "easy" wrong links, and constitutes a very challenging task for the CMs. The result scores obtained on this dataset might thus be a little pessimistic.

For the supervised CM, we train a LR model for each language pair. For each language pair, we use 70% of the data as the training set, 15% as the development set, and 15% as the test set.

7.5.2 Performance of Word Alignment Link Confidence Measures

The results of unsupervised and supervised CMs for the four language pairs are shown below.

From these results, we make several observations:

- First, the difficulties of datasets vary a lot. For the English-French corpus, several CMs obtain high AUCs and F-scores. For other language pairs, the CMs never obtain AUCs higher than 0.9.

- Second, the ranking of unsupervised CMs is relatively stable. The IBM 2 score obtains the best AUCs and F-scores for three language pairs, and is only outperformed by the supervised CM on the French-Spanish corpus. The confidence measures based on posterior word alignment link probabilities under IBM Models always obtain AUCs larger than 0.8, and perform consistently better than the other two unsupervised measures. The CM based on bilingual word embeddings performs poorly. But again, many variations remain to be explored in this direction.

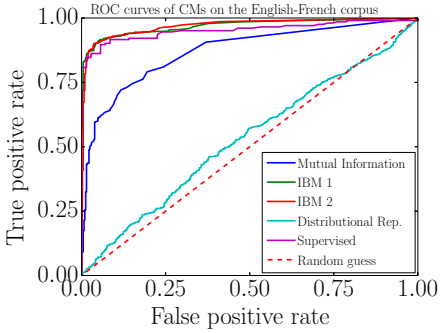


Figure 7.4: ROC curves of word alignment CMs on the English-French dataset.

CM	AUC	Best F.
Mutual Inf.	0.89	0.89
IBM 1 score	**0.97**	**0.95**
IBM 2 score	**0.97**	**0.95**
Distributional Rep.	0.53	0.86
Supervised	0.95	0.94

Table 7.3: AUCs and Best F-scores achieved by the CMs on the English-French dataset.

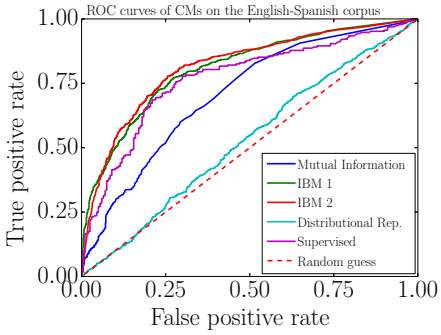


Figure 7.5: ROC curves of word alignment CMs on the English-Spanish dataset.

CM	AUC	Best F.
Mutual Inf.	0.71	0.74
IBM 1 score	0.81	0.77
IBM 2 score	**0.82**	**0.79**
Distributional Rep.	0.53	0.71
Supervised	0.77	0.76

Table 7.4: AUCs and Best F-scores achieved by the CMs on the English-Spanish dataset.

– Third, the supervised CM only works better than the unsupervised ones for one language pair. It seems that the LR models, which are simple combinations of unsupervised CMs and are trained on small amounts of supervision examples, are insufficient to obtain good CMs for word alignment. We can expect significant improvements in this direction, though, by adding more complex features and using more training data. As discussed, Tomeh et al. [2014] have shown that a finely-engineered LR model can achieve very good word alignment results.

Recall that this dataset has been intentionally designed to be difficult. It might be interesting to compare the results with those obtained on "normal" word alignments, that is, complete alignment matrices with all links, and to see whether significant changes emerge.

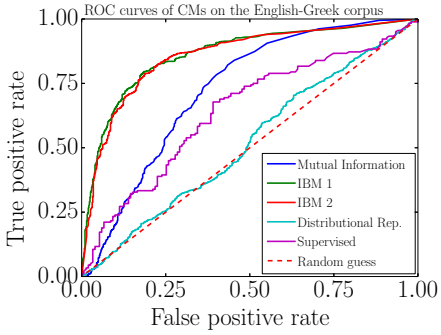


Figure 7.6: ROC curves of word alignment CMs on the English-Greek dataset.

CM	AUC	Best F.
Mutual Inf.	0.73	0.77
IBM 1 score	**0.86**	0.82
IBM 2 score	**0.86**	**0.83**
Distributional Rep.	0.52	0.72
Supervised	0.64	0.70

Table 7.5: AUCs and Best F-scores achieved by the CMs on the English-Greek dataset.

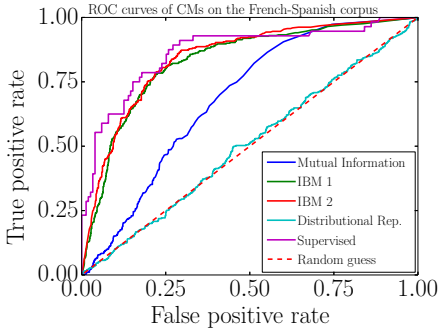


Figure 7.7: ROC curves of word alignment CMs on the French-Spanish dataset.

CM	AUC	Best F.
Mutual Inf.	0.69	0.52
IBM 1 score	0.83	0.66
IBM 2 score	0.84	0.66
Distributional Rep.	0.51	0.43
Supervised	**0.87**	**0.69**

Table 7.6: AUCs and Best F-scores achieved by the CMs on the French-Spanish dataset.

7.6 Conclusions

In this chapter, we have studied confidence measures for bitext alignment. We have proposed confidence measures for both sentence-level and word-level alignment links, and conducted experiments on evaluation datasets collected during this thesis. We can draw interesting conclusions from experimental results: for sentence alignment, a supervised CM trained on sufficiently large training corpora (even from different domains) beats unsupervised CMS; for word alignment, the CMs based on posterior probabilities under IBM Models work well in general, and IBM 2 consistently outperforms IBM 1.

This work is is still on-going, and many interesting directions are open. We list a few possibilities that we intend to follow in short terms:

– more complex supervised confidence measure for word alignment;

- other ways of employing distributional representations of words and sentences;
- many-to-many word alignment links. Our recursive bi-segmentation dataset (§ 5.2.3) can be useful. A good starting point is to study the behavior of the HMM word-to-phrase model of Deng and Byrne [2008] and the CutNalign algorithm [Lardilleux et al., 2013] with phrase-level confidence measures as association scores. This should be a difficult task, though: to the best of my knowledge, there is no clear intrinsic evaluation metrics;
- null links. It is particularly difficult to design CMs for null links, without knowledge of contextual links.

In the long term, we are interested in studying the effects of confidence measures in real applications, such as in bilingual reading and in cross-lingual transfer learning.

Chapter 8

Conclusions

In this chapter, we summarize the works presented in this thesis, and discuss perspectives.

8.1 Contributions

Our research activities are focused on the theme of obtaining high-confidence bitext alignments. We have investigated many aspects of bitext alignment, including the computation of sentence-level and word-level alignments, the representation of alignments, the creation of evaluation data, evaluation metrics, and confidence estimation of alignment links. Our main contribution is composed of three parts.

The first contribution consists of new methodologies and schemes for manual bitext alignments annotations, as well as collected reference data. These data is required in the evaluation of bitext alignment and confidence estimation. At the sentence-level, we identified the sparsity problem of suitable evaluation data, and collected gold alignments of literary bitexts. We also proposed a novel categorization of sentence alignment links, to make it possible to adjust confidence estimation according to targeted applications. At the word-level, we designed a finer classification of one-to-one word alignment links, the goal being to provide clear semantics to each class labels, so as to avoid ambiguities and misinterpretations. For many-to-many word alignment links, we proposed a novel recursive bi-segmentation approach, aiming to facilitate the annotation process and to improve annotation accuracy. All the collected data is released on-line at `https://transread.limsi.fr/resources.html`.

Our second major contribution is the development of two sentence alignment systems. Through a large-scale evaluation of state-of-the-art sentence aligners, we conclude that sentence alignment remains to be solved for difficult bitexts, and have identified several recurring problems in existing aligners: risky link type distribution assumptions, unsatisfactory modelings of null links, etc. Two models emerged from such studies. First, we attempted to employ external resources to improve typical sentence alignment systems,

and proposed a Maximum Entropy (MaxEnt) model. The second system, a 2D Conditional Random Fields model, investigated a novel structural representation of sentence alignment, which eliminated several risky assumptions and introduced an explicit representation of null links. Experimental results revealed an interesting fact: MaxEnt performed better on more regular bitexts, while CRF achieved better scores on more difficult bitexts. This finding calls for studies on the deployment strategy of alignment systems.

The third contribution is a set of confidence measures (CMs) for sentence-level and word-level alignment links. For both levels, we distinguish between unsupervised CMs and supervised CMs. We have conducted experiments on the collected datasets. For sentence alignment, the confidence estimation problem is surprisingly difficult, and the unsupervised CMs performed poorly. We have shown that the supervised CM for sentence alignment, even though trained on different types of alignment links, was capable of improving over unsupervised ones. For word alignment, we can conclude that posterior probabilities of alignment links under IBM Models 1 and 2 are fairly strong and robust CMs, and IBM Model 2 scores almost always archive better performance than the others proposed CMs. A simple combination of unsupervised CMs into a supervised one did not work well, but this should be investigated more extensively in following-up studies.

Other minor contributions include an interface for on-line manual sentence alignment, a representation format of bitext alignments, a gathering of existing bitext alignment resources (and a description of each other), etc.

8.2 Future Works

There are many directions of possible future works. Two main parts are the refinement of sentence alignment models and the extension of bitext alignment CMs.

One obvious direction of enforcing sentence alignment models is to extend the feature set. Currently the feature set is mainly based on lexical statistics, estimated on external parallel corpora. We can add other types of information: paraphrases, Babelfy entries, distributional representations, Wikipedia inverted index, etc. For the 2D CRF model, we should also try to enrich edge features, which have not been able to compensate strong single node features. Another important problem of the CRF model is the decoding algorithm. The current one seems not strong enough to eliminate noise.

In this thesis, we have only tested sentence alignment methods on relatively close language pairs. It will be interesting to experiment with more remote languages, such as English-Chinese, to analyze the usefulness of each type of information, and compare among language pairs.

For bitext alignment confidence estimation, a natural extension is to use scores provided by alignment systems as CMs, especially those of word alignment systems, which are often principled probabilistic models. Another very important research direction is to study many-to-many word alignments, for which the dataset presented in 5.2.3 would be useful. A

promising perspective is to study hierarchical alignment algorithms such as CutNalign, using confidence scores to locate split points, in order to obtain high-confidence sub-sentential, many-to-many word alignments.

Producing high-quality alignments is a challenging task. Sentence alignment is easy in some cases, yet fully automatic methods do not work well for difficult bitexts. Automatic sub-sentential alignment remains very tough. Hence, manual interventions are necessary in scenarios where a very high alignment quality is required. However, manually annotating bitext alignments is very labor-intensive. This calls for semi-automatic annotation tools with suitable interfaces, which can provide preliminary automatic annotations and let annotators perform post-editions instead of annotating from scratch. Such developments should be very useful.

Bitext alignments are ultimately used in various applications. Hence, the following-up work would pay more attention on how to effectively apply alignment techniques and CMs to improve application systems. The techniques and data presented in this thesis have already been employed in the TransRead bilingual reader. It is very interesting to pursuit the development of the reading device and to evolve the bitext alignment methods and confidence estimation techniques accordingly.

Appendices

Appendix A

Publications by the Author

2016

- Yong Xu, François Yvon. Novel Elicitation and Annotation Schemes for Sentential and Sub-sentential Alignments of Bitexts. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation*, pages 628-635, 2016.
- Yong Xu, François Yvon. A 2D CRF Model for Sentence Alignment. In *Proceedings of the 9th Workshop on Building and Using Comparable Corpora*, pages 11-20, 2016.
- François Yvon, Yong Xu, Marianna Apidianaki, Clément Pillias, and Pierre Cubaud. Transread: Designing a Bilingual Reading Experience with Machine Translation Technologies. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*, pages 27–31, 2016.

2015

- Yong Xu, Aurélien Max, François Yvon. Sentence Alignment for Literary Texts. *Linguistic Issues in Language Technology*, 12(6):1-29, 2015.

Appendix B

Reference Word Alignment Datasets

We briefly review some available manual word alignment datasets. As explained in § 4.2.3, our discussion can not be exhaustive. Only the data sets that we are aware of and are able to locate are presented here. We keep on searching for other available resources.

B.1 The Blinker project

The Blinker Project [Melamed, 1998a] is the first published work on the creation of reference word alignments. The guidelines [Melamed, 1998b] has been used in several subsequent studies. The data set is available at `http://nlp.cs.nyu.edu/blinker/`.

The motivation of the Blinker Project is to evaluate translation lexicons and statistical translation models in an objective and accurate way. Seven annotators aligned 250 verses of the Bible, which is composed by a version of modern English and one of modern French. Every verse was processed by five annotators. In Blinker, every word must be linked (unaligned words are linked to NULL). The links are not labeled with tags such as S or P.

After the annotation finished, the inter-annotator agreement (IAA) scores were computed between each pair of annotators using the metric described in § 4.2.3, with link weights. On one set of 100 verses, the grand mean of the IAA scores is 83.12, with standard deviation 5.16; on another set of 150 verses, the grand mean is 80.62 with standard deviation 5.44. Melamed [1998a] observed that function words are an important source of annotation divergences. After ignoring all links involving a function word, the mean IAA score rose by approximately 10 points on the two sets. Melamed [1998a] finally resolved all annotator disagreements by a vote. The winning annotations enter into the final word alignment set. The final set contains 250 pairs of verses, composed of 7,510 English words and 8,191 French words, of 1,714 and 1,912 distinct word types, respectively.

B.2 The Word Alignment Set of Och and Ney

In order to intrinsically evaluate automatic word alignment methods, in the work of Och and Ney [2000], two annotators annotated 484 English-French sentence pairs of the Hansard, using the Blinker guidelines and the standard annotation scheme presented in § 4.2.1. The bitext contains 7,681 English words and 8,482 French words, or 1,844 English word types and 2,071 French word types. In the reference word alignment set, the statistics of links are:[1]

- not counting null links: 19,222 links; 4,376 S-links (22.77%); 14,846 P-tagged pairs (77.23%);
- counting null links: 19,963 links; 4,376 S-links (21.92%); 15,587 P-tagged pairs (78.08%).

Thus, the null/non-null ratio is around 0.039. The data set is available at `http://web.eecs.umich.edu/~mihalcea/wpt/`. In this work, Och and Ney [2000] also proposed the AER metric.

B.3 The NAACL 2003 Workshop on Building and Using Parallel Texts

In 2003, a workshop on parallel texts was held during the HLT-NAACL conference [Mihalcea and Pedersen, 2003]. The shared task of the workshop consisted of evaluating word alignment methods. The participants were invited to automatically align English-French and English-Romanian bitexts. The English-French task used the data described in § B.2 as the reference alignment set.

As for the English-Romanian task, the organizers manually annotated 265 English-Romanian pairs, again using an adapted version of the Blinker guidelines. This set contains only S-links. Annotators disagreements are resolved in an adjudication phase. The 265 sentence pairs contain 6,135 English words and 5,957 Romanian words, or 1,610 English word types and 1,895 Romanian word types. The reference alignment set contains 6,750 non-null S-links, and 989 null S-links. The null/non-null ratio is around 0.147.[2] This data set is available at `http://web.eecs.umich.edu/~mihalcea/wpt/`.

B.4 The ACL 2005 Workshop on Building and Using Parallel Texts

Another workshop on parallel texts was held during the ACL conference in 2005 [Martin et al., 2005]. Again, the shared task was to evaluate word alignment methods, with a

[1] All null links are P-tagged in this data set.

[2] Interestingly, this null/non-null ratio is much higher than the English-French data set.

focus on languages with scarce resources. Three language pairs were considered: English-Romanian, English-Inuktitut, and English-Hindi. The data sets are available at `http://web.eecs.umich.edu/~mihalcea/wpt05/`.

For the English-Romanian task, the manual English-Romanian word alignments of § B.3 were used as a part of the reference data set (actually, this set played the role of trial set). The organizers added some reference word alignments. The added part contained 4,562 English words and 4,365 Romanian words, or 1,286 English word types and 1,485 Romanian word types. The additional reference alignments contained 5,034 non-null S-links, and 956 null S-links, giving the null/non-null ratio 0.190. All disagreements were resolved in an adjudication phase, so the P tag was not used in this data set.

For the English-Inuktitut task, the annotators were instructed to deliver *phrase-to-phrase* alignments. That is, during the annotation process, the lexical units considered were *continuous groups of words*, instead of words. Each such phrasal link was converted into word alignment links by P-tagging all word pairs inside the Cartesian product of the two sides, except when the two sides both contained only one word, in which case the single word pair was S-tagged. 90 sentence pairs were annotated. The bitext contained 2,796 English words and 1,104 Inuktitut words, or 815 English word types and 793 Inuktitut word types. The reference alignment set did not contain any null word alignment link. However, there were some unaligned words which did not belong to any phrase, but they were not in the word alignment link set, nor did Martin et al. [2005] explain how to count them in the evaluation. This reference word alignment set contained 330 S-tagged word links (12.78%), and 2,252 P-tagged pairs (87.22%). This ratio between the P tag and the S tag is remarkable. Interestingly, a part of the reference data set (the trial set) comes with the phrase pairs. We can observe that for this language pair, a common pattern is that an English phrase containing several words is linked to an Inuktitut phrase of one single word. In this trial set, the English part contained 737 words and 325 phrases (the average is roughly 2.27 words per phrase), while the Inuktitut part contained 318 words and 314 phrases (roughly 1 word per phrase).

For the English-Hindi task, 115 sentence pairs were annotated, which contained 1,440 English words and 1,739 Hindi words, or 559 English word types and 619 Hindi word types. In this data set, all word alignment links were S-tagged. There were 1,891 non-null links, and 512 null links. The null/non-null ratio was 0.271, which was considerably higher than for English-Romanian data set.

B.5 The EPPS Word Alignment Set

Lambert et al. [2005] proposed a detailed manual word alignment guidelines for English-Spanish. Their reference alignment set, the EPPS corpus, was annotated following this guideline. It is available at `http://glicom.upf.edu/lambert/data/epps-alignref.html`.

Several design choices influenced the creation of the reference word alignment set. First,

Lambert et al. [2005] considered that null links add noise into the AER metric, thus the reference word alignment set did not contain null links. Second, Lambert et al. [2005] pointed out that a reference alignment set with high S/P ratio would favor high-recall computed alignments. They have accordingly limited the use of the P tag, since the intended application of their data set was to evaluate word alignments in machine translation, where high-recall alignments are preferred. Lambert et al. [2005] also proposed a novel scheme for merging annotations. For any word pair that at least one annotator linked together, they attributed a score computed in the following way: for each annotator, if he/she S-tagged the pair, it received +1; if P-tagged, it received 0; if the annotator did not link the pair, it got -1. They then computed the sum of the scores. If the sum is strictly larger than half of the number of annotators (1.5 in this case), the word pair would be given the final tag S; if the sum is strictly less than half of the number of annotators, the pair would not be linked; otherwise it would be P-tagged. Lambert et al. [2005] argued that this scheme was more coherent than typical methods such as those listed in § 4.2.3, and was harmless to the discriminative capacity of AER.

Three annotators aligned 500 sentence pairs from the English-Spanish Europarl. This contained 14,652 English words and 15,516 Spanish words, or 3,090 English word types and 3,606 Spanish word types. It contained 14,985 S-tagged word pairs (69.88%), and 6,459 P-tagged pairs (30.12%), totaling 21,444 word alignment links. Compared to previous studies such as [Och and Ney, 2000], the S/P ratio of the EPPS data set is particularly high.

B.6 The Multiple Language Word Alignment Set of Graça et al.

Graça et al. [2008] present a multilingual word alignment annotation experience. The contribution of this project is two-fold : first, this is the first publicly available manual word alignment resource for Portuguese; second, the alignments, made on parallel sentences over four languages (Portuguese, English, French and Spanish), makes it possible to study the versatility of word alignment guidelines, and to develop general purpose annotation schemes. The resource is available at `https://www.l2f.inesc-id.pt/wiki/index.php/Word_Alignments`.

The annotation was performed following the guidelines in [Graça et al., 2008]. They used the S/P annotation scheme, but interpreted them as follows:[3] the S tag indicates the two words constitute a valid mutual translation in every context; the P tag means that the two words are only in correspondence in a certain context, or in the presence of functional words (which might or might not be present in the other language). Annotators resolved the disagreements by discussion.

100 sentences were chosen from the Europarl corpus. Each sentence existed in four languages. The English part contained 1,072 words and 466 word types; the French part

[3]This interpretation was different from the original proposition by Och and Ney [2000].

1,227 words and 474 types; the Spanish part 1,106 words and 472 types; the Portuguese part 1,131 words and 513 types. For each language pair, a reference word alignment set was created, totaling 6 sub-sets. As in [Lambert et al., 2005], null links were not included. The English-French set contained 1,009 S-tagged word pairs (69.78%), 437 P-tagged pairs (30.22%), 1,446 in total; the English-Portuguese set contained 771 S pairs (53.58%), 668 P pairs (46.42%), 1,439 in total; the English-Spanish set contained 835 S pairs (59.86%), 560 P pairs (40.14%), 1,395 in total;[4] the French-Portuguese set contained 1,061 S pairs (74.40%), 365 P pairs (25.60%), 1,426 in total; the French-Spanish set contained 1,139 S pairs (79.59%), 292 P pairs (20.41%), 1,431 in total; the Portuguese-Spanish set contained 863 S pairs (64.12%), 483 P pairs (35.88%), 1,346 in total. The grand total amounts to 8,483 word alignment links, composed by 5,678 S (66.93%) and 2,805 P (33.07%).

It is interesting to note that, even under the same guidelines and the same group of annotators, the S/P ratio varies considerably across language pairs, ranging from 3.90 (French-Spanish) to 1.15 (English-Portuguese). One might attempt to explain this variation by the intuitive idea that French, Portuguese and Spanish are all in the same language family, while English is in another one, thus the S/P ratio tends to be lower when English is involved. However, even though Portuguese and Spanish are generally acknowledged to be very close, their S/P ratio turns out to be slightly lower than the average. More detailed analyses might be necessary to better understand this discrepancy.

B.7 The Word Alignment Set of the GALE Project

During the DARPA Global Autonomous Language Exploitation (GALE) program [Li et al., 2009, 2010b], manual word alignments were created for two language pairs: English-Arabic and English-Chinese. Li et al. [2009, 2010b] explored various text genres: newswire (NW), broadcast news (BN), broadcast conversation (BC), and weblog (WB). The data is distributed by the Linguistic Data Consortium.[5]

The English-Arabic data set contains:[6]

- NW: 425,419 Arabic tokens;
- BN: 249,841 Arabic tokens;
- BC: 233,049 Arabic tokens;
- WB: 298,778 Arabic tokens

totaling 1,207,087 Arabic tokens. For English-Arabic, Li et al. [2009] specify two genres of non-null links: (a) translated and correct; (b) translated and incorrect. They also define

[4] This S/P ratio is much lower than the data set of [Lambert et al., 2005] for the same language pair, although both extracted sentences from the Europarl.

[5] `https://www.ldc.upenn.edu/`

[6] The statistics of the English side and alignment links of the GALE project are not published.

two genres of null links: (a) not translated and correct; (b) not translated and incorrect. We have detailed the interpretation of these link genres in § 4.2.1. The guidelines for this task is inspired by [Melamed, 1998b], and is available at `https://catalog.ldc.upenn.edu/docs/LDC2013T10/GALE_Arabic_alignment_guidelines_v6.0.pdf`.

The English-Chinese data set contains (here one Chinese token is equivalent to one character):

- NW: 248,999 Chinese tokens;
- BN: 385,405 Chinese tokens;
- BC: 344,564 Chinese tokens;
- WB: 383,077 Chinese tokens

totaling 1,362,045 tokens. The English-Chinese data set is more than word alignments: all alignment links are tagged with pre-defined labels reflecting linguistic properties of involved words, e.g. content or function words, attached or indecent links, etc. Words are also tagged, again using labels reflecting linguistic properties. These supplemental tags constitute an important feature of this data set. The detailed annotation scheme is at `https://catalog.ldc.upenn.edu/docs/LDC2012T24/GALE_Chinese_alignment_guidelines_v4.0.pdf`.

Li et al. [2009, 2010b] report quite high inter-annotator agreement rate ($\approx$ 0.90 F-score) the newswire bitexts of the English-Chinese data set. They also observe that the inter-annotator agreement rate for "semantic" links are higher than for "function" links (96% `vs.` 91%).

B.8 The Word Alignment Set of Holmqvist and Ahrenberg

Holmqvist and Ahrenberg [2011] proposed guidelines to produce manual English-Swedish word alignments. The data set is available at `http://www.ida.liu.se/labs/nlplab/ges/`, upon simple request. Holmqvist and Ahrenberg [2011] divided the data set into a training set and a test set, with different annotation strategies.

The training set was purposed to train supervised word alignment methods. Thus, it was aligned by one annotator, using guidelines similar to Blinker's. Links are not labeled. The training set was composed of 972 sentence pairs, with 20,340 English words and 18,343 Swedish words, or 3,594 English word types and 4,394 Swedish word types. The reference word alignment for the training set contained 4,248 null links and 20,426 non-null links (null/non-null ratio 0.208), totaling 24,674 links.

The test set was used to evaluate automatic word alignments. It was aligned by two annotators, following an adapted version of the guidelines of Lambert et al. [2005]. Here, the S/P annotation scheme was employed. The test set was composed of 192 sentence pairs, with 4,263 English words and 3,81,37 Swedish words, or 1,401 English word types and 1,457 Swedish word types. The reference word alignment for the test set contained:

– not counting null links: 4,577 links; 3,340 S-links (72.97%); 1,237 P-tagged word pairs (27.03%).

– counting null links: 5,253 links; 4,016 S-links (76.45%); 1,237 P-tagged word pairs (23.55%).

Thus, the null/non-null ratio is 0.148. In fact, all null links are S-tagged.

Appendix C

Detailed Performance of Sentence Aligners on Large-Scale Corpora

Table C.1 displays the performance of baseline tools on the two manually aligned reference corpora BAF and `manual en-fr`. In Table C.2, we show the performance of baseline tools on the large corpus `auto en-fr`. The results on the corpus `auto en-es` are provided in Table C.3. In these tables, we denote "Gr" the MaxEnt-based alignment algorithm with greedy search, and "DP" the MaxEnt-based alignment with dynamic programming.

	Link level F-score						
	GMA	BMA	Hun	Garg	Yasa	Gr	DP
BAF	61.4	73.6	71.2	65.6	75.7	76.3	66.5
Du Côté de chez Swann	92.8	91.5	90.9	92.2	92.2	89.4	93.3
Emma	53.5	57.4	57.7	51.7	59.9	61.4	51.0
Jane Eyre	77.1	61.1	59.3	71.8	66.9	67.4	78.9
La Faute de l'Abbé Mouret	91.5	88.4	92.6	97.1	95.6	95.3	98.0
Les Confessions	71.9	59.6	54.3	68.6	66.7	67.8	74.0
Les Travailleurs de la Mer	80.8	83.4	79.9	87.3	83.8	80.8	85.3
The Last of the Mohicans	89.9	82.7	87.1	92.7	88.6	85.8	90.1

Table C.1: F-scores on gold references

	Link level F-score						
	GMA	BMA	Hun	Garg	Yasa	Gr	DP
20000 Lieues sous les Mers	97.6	96.4	97.2	98.1	98.8	95.9	97.7
Alice's Adventures in Wonderland	81.2	74.3	80.0	83.2	82.7	76.2	81.5
A Study in Scarlet	89.0	78.2	83.8	85.2	89.0	85.0	86.4
Candide	85.7	78.8	82.5	82.6	87.9	79.9	86.4
Germinal	97.2	94.7	97.5	97.6	97.3	95.1	97.9
La Chartreuse de Parme	97.1	94.1	96.1	96.8	97.4	94.2	97.0
La Dame aux Camelias	94.0	91.1	89.6	93.8	94.9	90.7	94.6
Le Grand Meaulnes	93.3	91.0	93.4	94.7	94.1	92.6	94.3
Le Rouge et le Noir	96.9	94.7	96.4	97.2	97.9	94.3	97.3
Les Trois Mousquetaires	88.0	83.3	89.2	88.0	89.9	83.6	87.9
Le Tour du Monde En 80 Jours	76.4	63.9	74.9	75.8	78.5	68.9	75.8
L'île Mystérieuse	93.4	93.3	96.0	94.5	94.8	93.5	94.6
Madame Bovary	93.9	90.7	93.9	94.5	94.1	91.8	95.0
Moll Flanders	80.5	76.9	83.1	81.8	83.3	78.0	82.7
Notre Dame de Paris	93.5	91.1	92.8	94.2	94.1	90.6	94.2
Pierre et Jean	91.4	89.3	91.5	91.9	91.5	88.6	90.7
Pride and Prejudice	62.1	47.1	56.6	56.4	62.3	56.7	57.7
Rodney Stone	88.6	83.7	90.2	90.7	90.1	85.5	89.3
The Fall of The House of Usher	99.5	98.4	99.5	97.4	98.4	97.5	95.7
The Great Shadow	81.7	74.9	83.4	84.0	82.8	79.4	86.0
The Hound of The Baskervilles	92.8	90.5	92.5	93.7	93.5	91.1	93.0
Therese Raquin	85.6	80.3	84.7	85.4	85.1	82.0	86.7
Three Men in a Boat	85.3	76.5	81.7	85.6	87.1	81.8	86.2
Voyage au Centre de la Terre	84.9	81.6	83.7	86.8	85.9	83.5	84.0
Mean	88.7	84.0	87.9	88.7	89.6	85.7	88.9

Table C.2: F-scores on the large approximate reference set `auto en-fr`

	Link level F-score						
	GMA	BMA	Hun	Garg	Yasa	GR	DP
20000 Lieues sous les Mers	88.9	85.2	89.6	89.5	90.2	84.9	89.2
Alice's Adventures in Wonderland	74.2	65.3	66.7	72.2	76.2	71.4	74.4
Anna Karenina Volume I	72.9	69.3	75.4	77.8	73.6	70.9	74.2
Anna Karenina Volume II	69.5	68.2	73.3	75.6	73.2	69.3	72.4
A Study in Scarlet	94.3	91.3	94.1	96.3	96.9	93.1	92.8
Candide	76.8	64.2	71.7	63.2	80.4	71.7	72.8
Die Verwandlung	88.6	80.3	84.9	85.6	87.2	83.1	88.8
Don Quijote de La Mancha	87.9	82.1	85.6	88.8	89.6	81.9	86.6
Jane Eyre	60.3	48.8	43.7	58.4	64.3	60.9	58.3
Les Trois Mousquetaires	82.3	79.2	82.9	83.8	83.3	77.6	83.0
Le Tour du Monde en 80 Jours	78.8	71.1	77.8	77.4	81.4	74.4	78.2
L'île Mystérieuse	77.9	82.5	81.2	80.7	80.1	81.1	79.4
Sense and Sensibility	92.4	88.6	89.9	92.0	93.4	87.8	91.6
The Adventures of Sherlock Holmes	93.7	91.8	93.0	91.3	93.8	91.9	93.5
The Fall of the House of Usher	94.6	98.0	96.4	94.9	100.0	98.8	98.4
The Hound of the Baskervilles	96.5	95.3	95.4	96.8	97.6	95.4	96.2
Voyage au Centre de la Terre	77.2	72.2	75.6	79.1	77.5	74.7	76.4
Mean	82.8	78.4	81.0	82.6	84.6	80.5	82.7

Table C.3: F-scores on the large approximate reference set `auto en-es`

Appendix D

A Format for Representing Bitext Alignment (in French)

1 Introduction

En linguistique, un **corpus** est un ensemble structuré de textes. Les corpus sont souvent annotés afin qu'ils soient plus utilisables. Un **treebank** est un corpus de textes analysés syntaxiquement et/ou sémantiquement avec les annotations correspondantes. Un corpus peut être monolingue ou multilingue. Un **bitexte** est composé d'un côté d'un texte dans une langue et d'autre côté un texte d'une autre langue, et ces deux textes sont mutuellement en relation de traduction. Un **lien** d'alignement met en relation un groupe d'unités textuelles (par exemple des paragraphes, des phrases ou des mots) d'un côté du bitexte avec un groupe de l'autre côté (souvent on distingue le côté *source* du côté *cible*). Il est possible qu'une unité d'un côté ne soit alignée à rien de l'autre, dans ce cas le lien ne contient qu'un côté, ce qui donne le nom *lien nul*. Un **alignement** est l'ensemble des liens entre les deux textes. On se reportera par exemple à [Véronis, 2000, Melamed, 2001, Tiedemann, 2011] pour une présentation des méthodes pour construire et utiliser des alignements.

L'objectif du projet TransRead est d'étudier de nouvelles applications multilingues destinées à faciliter la consultation de documents dans plusieurs langues par des utilisateurs imparfaitement bilingues, pour qui des bitextes et, lorsqu'ils sont disponibles, des alignements sont des ressources de valeur. À l'inverse des approches « boîte noire » en traduction, qui ciblent un public monolingue, TransRead s'intéresse donc en premier lieu à la visualisation de textes bilingues et des alignements qui les lient.

Le domaine du projet étant principalement le traitement de documents, il est nécessaire que tous les participants emploient le même format d'annotation. D'une part, le système de visualisation des documents et celui qui calcule les alignements doivent avoir exactement la même notion de position sur chaque unité textuelle ; d'autre part, un format commun prédéfini facilite les développements individuels des programmes et logiciels.

Du point de vue de l'alignement, l'exigence pour ce format est que nous puissions représenter les alignements de tous les niveaux : non seulement entre des unités classiques comme les phrases ou les mots, mais aussi des segments des mots, des unités grammaticales, etc. Par ailleurs, d'autres types des données utiles à aider la compréhension des textes, comme la désambiguïsation du sens des mots, doivent aussi être représentées. Au final, il s'avère que nous devons avoir un mécanisme pour identifier uniquement chaque entité lexicale (une entité lexicale étant une chaine de caractères qui correspond à un symbole, qui est généralement un mot) des textes alignés. Avec ce mécanisme, la représentation des liens entre les entités et ses informations devient facile.

Du point de vue de la visualisation, le problème du format est plus compliqué. Dans la mesure où le calcul d'alignements traite principalement des textes bruts (sans indication de format ou de mise en page), les formats standards du domaine ne prennent généralement pas en compte les informations de présentation (les fontes, la mise en page, etc). Il est donc difficile (voire impossible) de développer des applications de lecture bilingue (sur des applications Web ou sur des terminaux mobiles comme des liseuses) à partir d'eux. En conséquence, nous avons décidé d'encoder les documents originaux dans le format du EPUB (Electronic PUBlication) [2], afin de faciliter les développements des modules de visualisation.

Enfin, du point de vue de la valorisation du projet, les applications de TransRead doivent être capables de fournir une bonne expérience utilisateur. Les plus décisifs sont la précision des résultats et la vitesse de réaction du système. Le format des fichiers est un facteur important pour la vitesse de réaction. Donc il faut que le format permette d'effectuer les requêtes efficaces sur des annotations disponibles dans les fichiers d'alignement. Compte-

2. http ://www.idpf.org/epub/30/spec/epub30-overview.html

tenu de la capacité limitée de calcul des terminaux mobiles, la représentation des alignements doit être claire, complète, bien structurée, et permettre de les récupérer sans calculs lourds. Ces exigences nous orientent vers un format basé sur le standard XML.

2 Des formats standards de la traduction

Dans cette partie, nous étudions des formats du domaine de la traduction, en nous focalisant sur l'alignement de phrases et l'alignement de mots, qui sont les unités d'alignement les plus étudiées. À partir de cette étude, il sera facile de représenter les liens aux autres niveaux. La traduction étant un domaine très dynamique à la fois dans le monde académique et dans l'industrie, plusieurs formats sont largement diffusés dans la communauté. Nous introduisons ceux qui sont considérés comme les standards et les analysons, dans le but de mettre en évidence leurs qualités et défauts par rapport aux exigences de TransRead.

On remarque qu'il existe plusieurs formats pour représenter les treebanks : Penn Treebank [3], Susanne [4], TIGER-XML [5], etc. Ces formats, n'étant pas conçus pour l'alignement, peuvent nous aider à développer la partie des annotations linguistiques pour le format de TransRead.

2.1 Des formats pour l'alignement de phrases

La phrase est probablement l'unité textuelle la plus souvent traitée dans l'industrie de la localisation et de l'internationalisation. Elle est aussi beaucoup étudiée dans les applications de traitement de bitextes. En traduction statistique, l'alignement de phrases est le point de départ de la chaine de traitements. Il est donc normal qu'un riche ensemble de formats existe pour représenter les traitements de phrases.

2.1.1 Le format de Uplug

Uplug [6] [Tiedemann, 2003] est un logiciel dédié à la recherche de l'alignement de textes. Le format qu'il propose est basé sur le *Corpus Encoding Standard for XML* (XCES) [7]. Essentiellement quatre balises, `<text>`, `<p>`, `<s>` et `<w>` sont employées pour repérer respectivement un document, un paragraphe, une phrase et un mot dans les documents constituant le bitexte. Les éléments `<p>`, `<s>` et `<w>` contiennent l'attribut d'identifiant (*id*) en respectant la hiérarchie linguistique classique. Par exemple, le premier paragraphe possède l'*id* "`1`", la première phrase du paragraphe "`1.1`", et le premier mot de la phrase "`1.1.1`". Donc tous les mots possèdent un *id* (unique au sein d'un document source ou cible) qui indique leur position dans le document. En ce qui concerne la représentation des alignements de phrases, un fichier tiers est créé dans lequel la balise `<cesAlign>` est utilisée pour décrire un alignement et la balise `<link>` pour décrire les liens le constituant. La balise `<cesAlign>` inclut les attributs "`fromDoc`" et "`toDoc`" indiquant les documents respectivement source et cible. Chaque `<link>` possède un attribut "`xtargets`" désignant les identifiants des unités mises en relation par ce lien, un attribut "`id`" qui est l'identifiant unique du lien et un attri-

3. http ://www.cis.upenn.edu/ treebank/
4. http ://www.grsampson.net/SueDoc.html
5. http ://www.ims.uni-stuttgart.de/forschung/ressourcen/werkzeuge/TIGERSearch/doc/html/TigerXML.html
6. `http://stp.lingfil.uu.se/~joerg/Uplug/home.html`
7. `http://www.xces.org/`

but "certainty" une valeur donnant la confiance selon une certaine mesure. Il est possible d'ajouter d'autres attributs au besoin.

Les figures 1 et 2 représentent deux documents originaux et 3 est le fichier d'alignement correspondant. Dans le listing 3, l'attribut "type" de l'élément <cesAlign> a pour valeur "sent", indiquant que ce groupe de liens décrit des relations entre phrases. Dans chaque élément <link>, la valeur de l'attribut "xtargets" est composée d'une série d'identifants sources et une série d'identifiants cibles. Les deux séries sont séparées par un point-virgule, et les identifiants dans une série sont séparés par un espace. Ainsi, un "xtargets" dont la valeur ne contient pas de point-virgule représente un lien nul.

Listing 1 – Le document source

```
<?xml version="1.0" encoding="utf-8"?>
<text>
  <p id="1">
    <s id="1.1">
      <w id="1.1.1">DE</w>
      <w id="1.1.2">LA</w>
      <w id="1.1.3">TERRE</w>
      <w id="1.1.4">A</w>
      <w id="1.1.5">LA</w>
      <w id="1.1.6">LUNE</w>
    </s>
    <s id="1.2">
      <w id="1.2.1">Trajet</w>
      <w id="1.2.2">Direct</w>
      <w id="1.2.3">en</w>
      <w id="1.2.4">97</w>
      <w id="1.2.5">Heures</w>
      <w id="1.2.6">20</w>
      <w id="1.2.7">Minutes</w>
    </s>
    <s id="1.3">
      <w id="1.3.1">par</w>
      <w id="1.3.2">Jules</w>
      <w id="1.3.3">Verne</w>
    </s>
    <s id="1.4">
      <w id="1.4.1">I</w>
    </s>
  </p>
</text>
```

Listing 2 – Le document cible

```
<?xml version="1.0" encoding="utf-8"?>
<text>
```

```
  <p id="1">
    <s id="1.1">
      <w id="1.1.1">FROM</w>
      <w id="1.1.2">THE</w>
      <w id="1.1.3">EARTH</w>
      <w id="1.1.4">TO</w>
      <w id="1.1.5">THE</w>
      <w id="1.1.6">MOON</w>
    </s>
    <s id="1.2">
      <w id="1.2.1">CHAPTER</w>
      <w id="1.2.2">I</w>
    </s>
  </p>
</text>
```

Listing 3 – Le fichier d'alignement

```
<?xml version="1.0" encoding="utf-8"?>
<!-- Doctype cesAlign  PUBLIC "-//CES//DTD cesAlign//EN" -->
<cesAlign fromDoc="xml/fr.xml" toDoc="xml/en.xml" type="sent">
  <linkList>
    <linkGrp>
      <link certainty="1" id="SL2" xtargets="1.1;1.1" />
      <link certainty="1" id="SL3" xtargets="1.2 1.3;" />
      <link certainty="1" id="SL4" xtargets="1.4;1.2" />
    </linkGrp>
  </linkList>
</cesAlign>
```

Une représentation également dérivée des propositions de XCES a été adopté pour le projet Européen PANACEA [8] ; elle utilise une version « modernisée » du format de UPLUG [9]. Dans la mesure où notre proposition, développée plus bas, repose sur des principes relativement proches, la conversion depuis et vers le format de PANACEA ne semble pas poser de difficultés majeures. Notons enfin qu'une variante de ce format est également utilisée dans le projet OpenCorp, qui inclut un outil pour visualiser les alignements de phrases [10].

2.1.2 Le format XLIFF

XLIFF [11] (XML Localization Interchange File Format) est un format standard dans l'industrie de localisation Il est proposé par l'OASIS (*Organization for the Advancement*

8. http://www.panacea-lr.eu/
9. Voir en particulier http://www.panacea-lr.eu/system/xcesXSD/TO1-documentation-v1.pdf
10. http://wanthalf.saga.cz/doc_intertext
11. La dernière version de XLIFF (1.2) est décrite dans le document : http://docs.oasis-open.org/xliff/xliff-core/xliff-core.html

of Structured Information Standards) [12] afin de standardiser les échanges des données de localisation entre les outils. Même si cela signifie que les documents XLIFF sont souvent des intermédiaires dans le processus de localisation, leur structure peut être une source d'inspiration pour la tâche d'alignement.

Dans le processus de localisation, la première étape consiste à séparer les parties visibles d'un document des données de mise en page représentées, par exemples, par des balises. Les données de mise en page sont stockées dans un fichier squelette, dans lequel une marque spéciale est attribuée à chaque partie visible. Le document représenté Figure 4 est un document original, et 5 le squelette correspondant.

Listing 4 – Exemple.html, un document HTML contennant 2 phrases visibles

```
<html>
  <head>
    <title>Un titre</title>
  </head>
  <body>
    <p>Un paragraphe</p>
  </body>
</html>
```

Listing 5 – Exemple.skl, le fichier squelette correspondant

```
<html>
   <head>
      <title>%%%1%%%</title>
   </head>
   <body>
      <p>%%%2%%%</p>
   </body>
 </html>
```

Les textes visibles sont segmentés avant d'être mis dans un document XML au format XLIFF, qui est essentiellement une liste d'éléments `<trans-unit>`, chacun contenant une unité de traduction. En pratique, une unité est souvent une phrase. Un élément `<trans-unit>` est composé d'un sous-élément `<source>` qui contient une phrase source, et d'un sous-élément `<target>` qui contient la traduction proposée. En plus de ces deux éléments, il peut y avoir un nombre non limité de sous-éléments `<alt-trans>` qui contiendront chacun une alternative de traduction, par exemple des versions anciennes ou les traductions dans d'autres langues (même si XLIFF est principalement conçu pour une seule paire de langues), sachant que tous ces éléments peuvent spécifier la langue de leur contenu par l'attribut "`xml:lang`".

Chaque élément `<trans-unit>` doit avoir un `id` correspondant à la marque spéciale dans le fichier squelette afin d'établir le resultat final de la localisation. Selon le même principe,

12. `https://www.oasis-open.org`

le fichier squelette doit être indiqué dans le document XLIFF. Le listing 6 représente un document XLIFF correspondant aux exemples 4 et 5.

Listing 6 – Le document XLIFF

```
<? xml version="1.0" ?>
<xliff version="1.0">
  <file original="Exemple.html"
        source-language="fr"
        datatype="HTML Page">
    <header>
      <skl>
        <external-file href="Exemple.skl"/>
      </skl>
    </header>
    <body>
      <trans-unit id="%%%1%%%">
        <source xml:lang="fr">Un titre</source>
        <target xml:lang="en">A title</target>
      </trans-unit>
      <trans-unit id="%%%2%%%">
        <source xml:lang="fr">Un paragraphe</source>
        <target xml:lang="en">A paragraph</target>
        <alt-trans xml:lang="en">One paragraph</alt-trans>
      </trans-unit>
    </body>
  </file>
</xliff>
```

2.1.3 Le format TMX

TMX (Translation Memory eXchange) [13] est un autre format standard dans la localisation. Il est développé par le LISA (*Localisation Industry Standards Association*) comme un format commun pour les bases de données des mémoires de traduction (TM, Translation Memory).

En général, les spécifications de TMX ressemblent beaucoup à celles de XLIFF. Il s'agit d'une représentation XML dans laquelle les unités de traduction sont entourées par la balise `<tu>`. Cette balise contient plusieurs sous-éléments `<tuv>`, chacun contenant un sous-élément `<seg>`, dont le contenu est un segment du texte. Nous pouvons choisir, pour chaque `<tu>`, le type de segment en utilisant l'attribut `"segtype"`, dont quatre valeurs sont possibles : `"block"`, `"paragraph"`, `"sentence"`, et `"phrase"`. En pratique, un segment est souvent une phrase. Les différences entre TMX et XLIFF viennent principalement de leurs différentes utilités : dans le format TMX, l'ordre des unités n'est pas pertinente.

Le listing 7 représente un document au format TMX.

13. `http://www.gala-global.org/oscarStandards/tmx/tmx14b.html`

Listing 7 – Un document TMX

```
<?xml version="1.0" ?>
<tmx version="1.4">
 <header datatype="PlainText" segtype="sentence"
  adminlang="en-us" srclang="EN" o-tmf="ABCTransMem">
 </header>
 <body>
  <tu>
   <tuv xml:lang="EN">
    <seg>Text <bpt i="1">&lt;B&gt;</bpt>bold<ept i="1">&lt;/B&gt;</ept></seg>
   </tuv>
   <tuv xml:lang="FR">
    <seg>Texte <bpt i="1">&lt;B&gt;</bpt>gras<ept i="1">&lt;/B&gt;</ept></seg>
   </tuv>
  </tu>
 </body>
</tmx>
```

2.1.4 L'analyse des formats standards

Nous avons brièvement présenté, dans les sections précédentes, trois formats standardisés d'alignement. XCES propose une annotation à la fois pour la représentation des documents originaux et l'alignement, tandis que XLIFF et TMX représentent principalement des alignements. Du point de vue de TransRead, XCES et TMX ne sont pas directement utilisables, puisque :

1. L'encodage des documents originaux de XCES (donc celui de Uplug) n'est pas suffisamment fin. Bien que le mécanisme d'annotation de XCES soit théoriquement capable d'encoder tous les objets dans les documents (les textes, les images, les tableaux, etc), il ne fournit pas de moyens pour gérer les informations de présentation, concernant tant la mise en forme (la police, la disposition, etc.), que la mise en page des textes. Cela pose un problème pour la visualisation et l'interaction humaine-machine avec les bitextes. L'objectif de TransRead étant de faciliter la consultation de documents existant en plusieurs langues, ce défaut devient rédhibitoire.
2. TMX ne permet pas d'établir des correspondances entre les documents originaux et le fichier d'alignement. Si nous pouvons parfaitement stocker tous les liens trouvés dans un fichier TMX, il ne fournit aucun moyen pour repositionner les segments extraits dans les document originaux, ce qui rend leur visualisation impossible.

En revanche, XLIFF et la partie représentant les alignements de XCES peuvent tous les deux servir de point de départ pour la représentation d'alignements adaptée aux besoins du projet TransRead. XCES propose un jeu de balise très complet et adapté à l'alignement dans sa DTD **cesAlign** [14]. XLIFF dispose d'une remarquable extensibilité pour les balises et les attributs, qui rend l'encodage des autres niveaux possible. Le format proposé sera donc basé sur ces deux formats.

14. http://www.cs.vassar.edu/CES/sgml/cesAlign.dtd

2.2 Des formats pour l'alignement de mots

L'alignement de mots est un sujet rarement traité dans les applications industrielles (sinon pour des systèmes de traduction automatique). Il est, au contraire, très important dans les travaux de recherche en traduction automatique. En conséquence, la plupart des formats d'alignement de mots ont été initialement proposés dans la communauté scientifique. La figure 1 est une illustration en matrice d'un alignement de mots d'une paire de phrases simples.

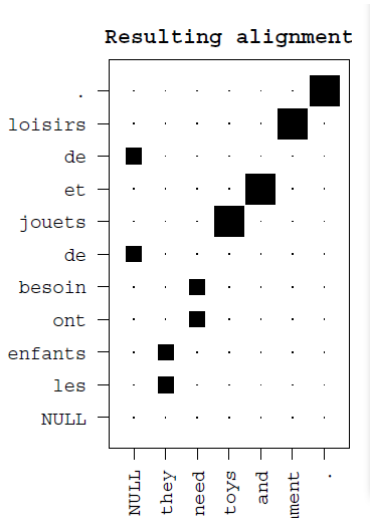


Figure 1 – Un exemple de l'alignement de mots, représenté en matrice

L'alignement de mots étant un problème de recherche en pleine évolution, il n'existe pas, à notre connaissance, de format standardisé. Nous allons néanmoins introduire plusieurs formats populaires dans la communauté, afin de décrire quelques problèmes et conventions de l'alignement de mots.

2.2.1 L'évaluation de 2003

En 2003, a eu lieu un atelier d'alignement de mots lors de la conférence NAACL (North American Chapter of the Association for Computational Linguistics) [Mihalcea and Pedersen, 2003]. Les participants étaient invités à aligner les mots dans des paires de phrases parallèles. Chaque paire de phrases parallèles est identifiée par un `id`. L'alignement des tokens d'une paire (L_e, L_f) est représenté par plusieurs lignes dans un fichiers selon le format :

`id_paire_phrase`	`pos_dans_`L_e	`pos_dans_`L_f	`[S` *or* `P]`	`[score]`

où `id_paire_phrase` est l'identifiant de la paire de phrases, `pos_dans_`L_e (resp. L_f) est la position d'un mot de L_e, (resp. L_f). `[S` *or* `P]` indique si ce lien est *sûr* ou seulement *possible*, `[score]` un score de confiance de ce lien. Les deux derniers champs sont facultatifs.

Dans ce format, tous les symboles séparés par un espace dans les textes parallèles sont considérés comme des mots et sont donc alignables ; y compris les ponctuations. Tous les

mots doivent être alignés, ceux qui n'ont pas de correspondances doivent être liés au mot supplémentaire `NULL` ajouté au début de chaque phrase, à la position conventionnelle 0. Un mot d'un coté peut être aligné à un ou plusieurs mots de l'autre coté. Cela permet de représenter les liens entre les groupes de mots.

Ce format est inspiré de celui d'alignement de mots utilisé dans le corpus Blinker [15] dont les principes d'annotation sont expliqués dans [Melamed, 1998]. Il a été utilisé dans plusieurs projets de recherche, par exemple le corpus décrit dans [de Almeida Varelas Graça et al., 2008], décrivant des alignements dans six langues [16]. Un trait caractéristique de ce format, qui a été repris dans de nombreuses expériences ultérieures, est la distinction entre les liens surs et possibles. Cette distinction s'est progressivement imposée pour distinguer les configurations dans lesquels l'alignement est sans ambiguïté des cas où la traduction est non littérale ou bien non compositionnelle, ou bien encore les cas où des mots ne sont pas traduits [17].

Le listing 8 est l'alignement de mots d'une paire de phrases ayant l'*id* 18.

Listing 8 – Un alignement de mots

```
#<s snum=18> They had gone . </s>
#<s snum=18> Ils étaient allés . </s>
18 1 1 1
18 2 2 P 0.7
18 3 3 S
18 4 4 S 1
```

On note sur cet exemple le lien de type `[P]` entre *have* et *sont*, qui, bien que l'un ne soit pas la traduction de l'autre, marque ici le fait qu'ils occupent la même fonction d'auxiliaire.

2.2.2 Le format de Moses

Moses [Koehn et al., 2007] est un logiciel très utilisé dans la communauté de la traduction automatique. Il inclut un paquet GIZA++, décrit dans [Och and Ney, 2003] dédié à la prédiction des alignement de mots, qui est le point de départ de la plupart des activités de la recherche du domaine. L'alignement de mots d'une phrase parallèle est représenté par trois lignes dans le fichier de résultat de Moses : la première pour indiquer l'*id* de la paire de phrases ; la deuxième est la phrase source ; la troisième ligne est la plus critique : un mot supplémentaire `"NULL"` est ajouté au début de la phrase cible, chaque mot cible est suivi par la liste des mots sources correspondants. Le listing 9 est un alignement de mots représenté dans ce format .

Listing 9 – Un alignement de mots dans Moses

15. `http://nlp.cs.nyu.edu/blinker/index.html`
16. `https://www.l2f.inesc-id.pt/wiki/index.php/Word_Alignments`
17. Cette distinction est introduite dans [Och and Ney, 2003] avec la définition suivante (p. 33) :

    (...) an S (sure) alignment, for alignments that are un- ambiguous, and a P (possible) alignment, for ambiguous alignments. The P label is used especially to align words within idiomatic expressions and free translations and missing function words (...)

```
# Paire de phrase 18
He meets her with pleasure .
NULL ({}) Avec ({4}) plaisir ({5}) ,({}) il ({1}) la ({3}) voit ({2}) . ({6})
```

2.2.3 Le format du *Alignment Set Toolkit*

Alignment Set Toolkit [18] est un logiciel destiné à traiter les alignements manuels établis en respectant les principes décrits dans [Lambert et al., 2005]. Trois fichiers séparés sont utilisés pour représenter une tâche d'alignement : un pour les phrases sources, un pour les phrases cibles, et un troisième pour les alignements. Dans chaque fichier, chaque ligne correspond à une paire de phrases parallèles, et l'ordre des paires de phrases est le même dans les trois fichiers. Ainsi, pour une paire de phrases parallèles, la phrase source, la cible, et les liens d'alignement relatifs à la paire de phrase ont le même indice de ligne. Un lien est représenté par l'indice d'un mot source et l'indice du mot cible aligné, separé par un "`s`" ou un tiret pour les liens sûrs, ou un "`p`" pour les liens non sûrs. Les indices commencent par 1, car 0 est utilisé pour "`NULL`". Le listing 10 est un alignement de mots représenté dans ce format.

Listing 10 – Un alignement de mots dans le format de Alignment Set Tookkit

```
Le document source
    I can not say anything at this stage .
Le document cible
    En ce moment , je ne peux rien dire .
Le fichier d'alignement source-cible
    0-4 1-5 2-7 3-6 4-9 5p8 6-1 7-2 8-3 9-10
```

2.2.4 Le format du projet SMULTRON

SMULTRON (The Stockholm MULtilingual parallel TReebank) est un projet d'annotation linguistique réalisé à l'Université de Stockholm et l'Université de Zurich [Volk et al., 2010], qui vise à produire des alignements sous-phrastiques entre structures linguistiques. La banque d'arbres représentant d'un document original est stockée dans un fichier XML respectant le format TIGER-XML. En général, une phrase est décrite par un élément `<s>` ayant l'attribut "`id`" et deux sous-éléments `<termninals>` et `<nonterminals>`. Le `<nonterminals>` contient plusieurs sous-éléments `<nt>` encodant les différents constituants. Le `<termninals>` contient une liste ordonnée de sous-éléments `<t>`, chacun correspondant à un mot de la phrase. Les attributs d'un `<t>` ne sont pas spécifiés. Normalement le groupe d'attributs contient "`id`", "`word`" dont la valeur est le mot, "`pos`" qui indique la catégorie grammaticale du mot, "`lemma`" pour le lemme, etc.

Le format d'alignement proposé par SMULTRON est également basé sur XML. L'alignement est représenté dans un fichier tiers. Deux éléments `<treebank>` sont présents pour indiquer les deux *treebanks* qui sont alignées. Ils possèdent trois attributs : "`id`", "`language`" et "`filename`". L'élément `<alignments>` regroupe tous les `<align>`. Un `<align>` possède

18. http ://www-lium.univ-lemans.fr/ lambert/software/AlignmentSet.html

comme attribut "`type`", dont les valeurs possibles sont "`good`" et "`fuzzy`" pour indiquer respectivement les liens sûrs et les approximatifs. Il a deux sous-éléments `<node>`, chacun encodant un mot d'un *treebank*. Un `<node>` a deux attributs : "`treebank_id`" qui doit se référer à l'`id` d'un *treebank*, et "`node_id`" dont la valeur doit être une référence à l'`id` d'un mot dans un fichier de *treebank*.

Les listings 11, 12 et 13, extraits du corpus « Sophie's world » délivré par SMULTRON, illustrent l'utilisation de ce format. Ci-dessous, le fichier *smultron_en_sophie.xml* est le *treebank* de la version anglaise, et *smultron_sv_sophie.xml* celui de la version suédoise.

Listing 11 – Un extrait de smultron_en_sophie.xml

```
<?xml version="1.0"?>
<corpus>
  <head>...</head>
  <body>
    ...
    <s id="s2">
      <graph root="s2_508">
        <terminals>
          <t id="s2_1" word="..." pos=":" morph="--"/>
          <t id="s2_2" word="at" pos="IN" morph="--"/>
          <t id="s2_3" word="some" pos="DT" morph="--"/>
          <t id="s2_4" word="point" pos="NN" morph="--"/>
          <t id="s2_5" word="something" pos="NN" morph="--"/>
          <t id="s2_6" word="must" pos="MD" morph="--"/>
          <t id="s2_7" word="have" pos="VB" morph="--"/>
          <t id="s2_8" word="come" pos="VBN" morph="--"/>
          <t id="s2_9" word="from" pos="IN" morph="--"/>
          <t id="s2_10" word="nothing" pos="NN" morph="--"/>
          <t id="s2_11" word="..." pos=":" morph="--"/>
        </terminals>
        <nonterminals>
          <nt id="s2_500" cat="NP">
            <edge label="--" idref="s2_3"/>
            <edge label="--" idref="s2_4"/>
          </nt>
          <nt id="s2_501" cat="NP">
            <edge label="--" idref="s2_5"/>
          </nt>
          <nt id="s2_502" cat="NP">
            <edge label="--" idref="s2_10"/>
          </nt>
          <nt id="s2_503" cat="PP">
            <edge label="--" idref="s2_2"/>
            <edge label="--" idref="s2_500"/>
          </nt>
          <nt id="s2_504" cat="PP">
            <edge label="--" idref="s2_9"/>
```

```
            <edge label="--" idref="s2_502"/>
          </nt>
          <nt id="s2_505" cat="VP">
            <edge label="--" idref="s2_8"/>
            <edge label="CLR" idref="s2_504"/>
          </nt>
          <nt id="s2_506" cat="VP">
            <edge label="--" idref="s2_7"/>
            <edge label="--" idref="s2_505"/>
          </nt>
          <nt id="s2_507" cat="VP">
            <edge label="--" idref="s2_6"/>
            <edge label="--" idref="s2_506"/>
          </nt>
          <nt id="s2_508" cat="S">
            <edge label="--" idref="s2_1"/>
            <edge label="--" idref="s2_11"/>
            <edge label="SBJ" idref="s2_501"/>
            <edge label="TMP" idref="s2_503"/>
            <edge label="--" idref="s2_507"/>
          </nt>
        </nonterminals>
      </graph>
    </s>
    ...
  </body>
</corpus>
```

Listing 12 – Un extrait de smultron_sv_sophie.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<corpus>
  <head>...</head>
  <body>
    ...
    <s id="s2">
      <graph root="s2_506">
        <terminals>
          <t id="s2_1" word="..." pos="DL" morph="--" lemma="--" type="--"/>
          <t id="s2_2" word="en" pos="DT" morph="--" lemma="en" type="--"/>
          <t id="s2_3" word="gång" pos="NN" morph="UTR" lemma="gång" type="--"/>
          <t id="s2_4" word="i" pos="PR" morph="--" lemma="i" type="--"/>
          <t id="s2_5" word="tiden" pos="NN" morph="UTR" lemma="tid" type="--"/>
          <t id="s2_6" word="måste" pos="VBFIN" morph="--" lemma="måste" type="--"/>
          <t id="s2_7" word="ändå" pos="AB" morph="--" lemma="ändå" type="--"/>
          <t id="s2_8" word="allting" pos="PN" morph="--" lemma="allting" type="--"/>
```

```
    <t id="s2_9" word="ha" pos="VBINF" morph="--" lemma="ha" type="--"/>
    <t id="s2_10" word="blivit" pos="VBSUP" morph="--" lemma="bliva" type="--"/>
    <t id="s2_11" word="till" pos="PL" morph="--" lemma="till" type="--"/>
    <t id="s2_12" word="av" pos="PR" morph="--" lemma="av" type="--"/>
    <t id="s2_13" word="noll" pos="RG" morph="--" lemma="noll" type="--"/>
    <t id="s2_14" word="och" pos="KN" morph="--" lemma="och" type="--"/>
    <t id="s2_15" word="ingenting" pos="PN" morph="--" lemma="ingenting" type="--"/>
    <t id="s2_16" word="." pos="DL" morph="--" lemma="--" type="--"/>
  </terminals>
  <nonterminals>
    <nt id="s2_500" cat="PP">
      <edge label="HD" idref="s2_4"/>
      <edge label="NK" idref="s2_510"/>
    </nt>
    <nt id="s2_501" cat="CNP">
      <edge label="CJ" idref="s2_13"/>
      <edge label="CD" idref="s2_14"/>
      <edge label="CJ" idref="s2_507"/>
    </nt>
    <nt id="s2_502" cat="NP">
      <edge label="NK" idref="s2_2"/>
      <edge label="HD" idref="s2_3"/>
      <edge label="MNR" idref="s2_500"/>
    </nt>
    <nt id="s2_503" cat="PP">
      <edge label="HD" idref="s2_12"/>
      <edge label="NK" idref="s2_501"/>
    </nt>
    <nt id="s2_504" cat="VP">
      <edge label="HD" idref="s2_10"/>
      <edge label="SVP" idref="s2_11"/>
      <edge label="MO" idref="s2_503"/>
    </nt>
    <nt id="s2_505" cat="VP">
      <edge label="HD" idref="s2_9"/>
      <edge label="OC" idref="s2_504"/>
    </nt>
    <nt id="s2_506" cat="S">
      <edge label="HD" idref="s2_6"/>
      <edge label="MO" idref="s2_508"/>
      <edge label="SB" idref="s2_509"/>
      <edge label="MO" idref="s2_502"/>
      <edge label="OC" idref="s2_505"/>
    </nt>
    <nt id="s2_507" cat="NP">
      <edge label="HD" idref="s2_15"/>
    </nt>
    <nt id="s2_508" cat="AVP">
```

```
            <edge label="HD" idref="s2_7"/>
          </nt>
          <nt id="s2_509" cat="NP">
            <edge label="HD" idref="s2_8"/>
          </nt>
          <nt id="s2_510" cat="NP">
            <edge label="HD" idref="s2_5"/>
          </nt>
        </nonterminals>
      </graph>
    </s>
    ...
  </body>
</corpus>
```

Listing 13 – Un extrait de l'alignement pour Sophie's world

```
<?xml version="1.0" encoding="UTF-8"?>
<treealign subversion="3" version="2">
  <head>
    ...
    <treebanks>
      <treebank id="en" language="en_US" filename="smultron\_en\_sophie.xml"/>
      <treebank id="sv" language="sv_SE" filename="smultron\_sv\_sophie.xml"/>
    </treebanks>
    ...
  </head>
  <alignments>
    ...
    <align type="good">
      <node treebank_id="en" node_id="s2_6"/>
      <node treebank_id="sv" node_id="s2_6"/>
    </align>
    <align type="good">
      <node treebank_id="en" node_id="s2_7"/>
      <node treebank_id="sv" node_id="s2_9"/>
    </align>
    <align type="good">
      <node treebank_id="en" node_id="s2_8"/>
      <node treebank_id="sv" node_id="s2_10"/>
    </align>
    <align type="good">
      <node treebank_id="en" node_id="s2_8"/>
      <node treebank_id="sv" node_id="s2_11"/>
    </align>
    <align type="good">
      <node treebank_id="en" node_id="s2_9"/>
```

```
      <node treebank_id="sv" node_id="s2_12"/>
    </align>
    <align type="good">
      <node treebank_id="en" node_id="s2_10"/>
      <node treebank_id="sv" node_id="s2_15"/>
    </align>
    <align type="good">
      <node treebank_id="en" node_id="s2_502"/>
      <node treebank_id="sv" node_id="s2_507"/>
    </align>
    <align type="fuzzy">
      <node treebank_id="en" node_id="s2_503"/>
      <node treebank_id="sv" node_id="s2_502"/>
    </align>
    <align type="good">
      <node treebank_id="en" node_id="s2_504"/>
      <node treebank_id="sv" node_id="s2_503"/>
    </align>
    <align type="good">
      <node treebank_id="en" node_id="s2_505"/>
      <node treebank_id="sv" node_id="s2_504"/>
    </align>
    <align type="good">
      <node treebank_id="en" node_id="s2_506"/>
      <node treebank_id="sv" node_id="s2_505"/>
    </align>
    <align type="fuzzy">
      <node treebank_id="en" node_id="s2_508"/>
      <node treebank_id="sv" node_id="s2_506"/>
    </align>
    ...
  </alignments>
</treealign>
```

2.2.5 L'analyse des formats

Ces formats sont actuellement très utilisés dans la communauté de la traduction automatique. Le format de l'atelier 2003 permet essentiellement de représenter des éléments désirés (identifiant des mots, scores de confiance, etc). Le format de SMULTRON fournit une spécification plus complète qui permet de représenter les informations linguistiques et des alignements entre syntagmes ou plus généralement segments de mots. Il est d'ailleurs plus standardisé et extensible. Nous allons concevoir un format basé aussi sur XML pour l'alignement, en nous appuyant principalement sur ces deux propositions.

3 Le format proposé

En nous inspirant de la DTD d'annotation d'alignements **cesAlign**, nous proposons un format basé sur le standard XML. Afin de faciliter la visualisation des textes, nous avons adopté la stratégie d'enregistrer les annotations dans un document tiers, au lieu de les fusionner avec les documents originaux. La seule contrainte pour ces derniers est qu'ils soient représentés au format XHTML, sans autre restriction particulière. Les correspondances entre les documents originaux et le document d'annotation sont établies à l'aide d'un mécanisme d'adressage qui généralise celui de **cesAlign**. Dans cette section, nous allons d'abord présenter le mécanisme d'adressage pour les textes des documents originaux, ensuite décrire progressivement la proposition de format.

La DTD précise et le schéma XML sont présentés dans l'annexe 5.1 et 5.2.

3.1 L'adressage

Dans la section 1, nous avons noté la nécessité d'associer chaque unité textuelle à un identifiant unique. Idéalement, cette identification peut être faite en attribuant un attribut `"ID"` à chacun des éléments du texte. Cependant, cette méthode n'est pas toujours possible parce que cet attribut est le plus souvent absent dans des documents, et qu'il n'est souvent pas désirable (voire parfois impossible [19]) de les rajouter. Nous avons donc décidé d'identifier chaque unité par le chemin de la racine vers cette unité dans l'arbre DOM (Document Object Model [20]) du document. Le DOM représente un document XHTML comme un arbre généalogique dont la racine supérieure est toujours un nœud `<document>`, qui a pour fils le nœud `<html>`. Toutes les balises comprises dans le document XHTML sont considérées comme des nœuds. Il faut en particulier remarquer que le contenu texte de chaque balise est également considéré comme un nœud de type texte nommé `<text>`.

Le mécanisme d'adressage pour les documents XHTML ne peut être conçu au niveau de mots comme dans le format de Moses, parce que la tokenisation est plus compliquée. Dans les fichiers XHTML, il est possible qu'un mot soit décomposé dans plusieurs éléments. Le tokeniseur ne peut donc considérer les balises ou les espaces comme les frontières des mots. En conséquence, nous ne pouvons pas utiliser la notion d'indice pour des mots dans des éléments. L'adressage est donc fait au niveau des caractères.

Nous définissons le chemin d'un caractère comme étant composé par trois parties : un identifiant du document, la position de l'élément le contenant dans le document (donc toujours un nœud `<text>`), et la position relative du caractère dans cet élément `<text>`. Tous les indices positionnels commencent par la valeur 0. La position de l'élément contenant ce caractère est déterminée dans l'arbre DOM du document de façon descendante : à partir du `<document>`, on traverse le chemin vers le nœud en question en prennant l'indice de chaque nœud parmi ses frères, ce qui donne une série de nombres. Cette série est séparée par le point. La position de l'élément et la position relative du caractère dans cet élément sont combinées par un tiret. Par exemple, le quatrième caractère du troisième fils du deuxième fils du `<document>` a pour position `"1.2-3"`. Cette position et l'identifiant du document est séparés par une espace. Donc pour l'exemple ci-dessus, si l'*id* du document est `"doc"`, alors la position totale du caractère en question est `"doc 1.2-3"`. Avec cette définition, nous pouvons calculer le chemin d'une entité en indiquant la position de son premier et sont dernier

19. En toute généralité, les mots des documents sources et cibles peuvent contenir un nombre arbitraire de balises de mise en forme.

20. `http://www.w3.org/DOM/`

caractère dans le document. En conséquence, un chemin comprend toujours deux positions. Pour les unités non-textuelles, nous mettons la valeur 0 pour la troisième partie des deux positions. Ci-dessous, nous donnons un exemple pour un vrai document XHTML.

Listing 14 – Un fichier XHTML

```
<?xml version='1.0' encoding='UTF-8'?>
<!DOCTYPE html PUBLIC '-//W3C//DTD XHTML 1.1//EN'
'http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd'>

<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    <title>le titre</title>
    <meta content="text/css" http-equiv="Content-Style-Type" />
    <link href="pgepub.css" rel="stylesheet" type="text/css" />
  </head>

  <body>
    <p>L'exemple est fait par M<sup>me.</sup> XXX.</p>
  </body>
</html>
```

Donc, pour la page très simple représentée Figure 14 ayant pour identifiant "ex_doc", nous allons montrer l'adressage réalisé pour le mot "exemple". Il faut d'abord déterminer la position du premier caractère "e". La racine <document> a deux nœuds fils <documenttype> et <html>. Le nœud <html> possède cinq fils : un <text> pour l'espace entre la balise <html> et la balise <head>, <head>, <text> pour l'espace entre <head> et <body>, <body>, <text> pour l'espace entre <body> et <html>. Ainsi nous pouvons déduire que la première partie du chemin du premier caractère (qui est entourné par un nœud <text>) est : "1.3.1.0" puisque <html> est le deuxième fils du nœud <document> ; <body> le quatrième fils du <html>, <p> le deuxième fils de <body> ; <text> le premier fils du <p>. La deuxième partie de la position est l'indice de ce caractère dans l'élément <text> (ici, 2, car c'est le troisième caractère dans cet élément), ce qui nous donne la position complète du caractère : "ex_doc 1.3.1.0-2". De même façon, nous obtenons la position du dernier caractère du mot « exemple » comme étant égal à "ex_doc 1.3.1.0-8". Ce mot est donc identifié par les deux adresses "ex_doc 1.3.1.0-2" et "ex_doc 1.3.1.0-8".

Un point important de ce mécanisme est que nous supposons que les chemins définissent toujours des unités *connexes*. C'est-à-dire que tous les caractères entre les deux positions appartiennent à cette unité. Dans l'exemple 14, le mot « Mme. » étant divisé dans deux éléments différents, son chemin est "ex_doc 1.3.1.0-23 - ex_doc 1.3.1.1.0-2", où les premières parties de l'adresse renvoient à des éléments différents. Si, d'un autre point de vue, nous considérons que « Mme. XXX » est une unité unique, alors notre méthode donne le chemin "ex_doc 1.3.1.0-27 - ex_doc 1.3.1.2-3". La règle de continuité s'applique ici : l'élément <sup> ("1.3.1.1") étant entre les deux positions du chemin, son contenu fait partie de l'unité. Grâce à la règle de contiguité, un algorithme simple permet de facilement retrouver le contenu d'une unité à partir de son chemin.

3.2 Le format d'annotation

Dans cette section nous décrivons le format d'annotation. Le schéma XML correspondant, ainsi qu'un exemple de document annoté selon ce schéma sont donnés en annexe.

3.2.1 L'élément `trAnnot`

L'élément hiérarchiquement le plus haut est `<trAnnot>`. Cette balise marque le début et la fin d'un document d'annotation. Elle contient un sous-élément `<docList>`, un ou plusieurs sous-éléments `<linkList>`. Il dispose d'un seul attribut obligatoire "`version`", dont la valeur indique la version du schéma XML utilisée par le document.

3.2.2 L'élément `docList`

Un élément `<docList>` contient au moins un sous-élément `<docName>`.

3.2.3 L'élément `docName`

L'élément `<docName>` décrit un document original. Il a deux attributs :
- id : l'identifiant du document. Cet identifiant est utilisé dans les positions des unités.
- xml :lang : la langue du document.

3.2.4 L'élément `linkList`

Un élément `<linkList>` regroupe tous les liens d'un même niveau linguistique (phrase, mot, etc). Il contient un ou plusieurs sous-éléments `<linkGroup>`. Un `<linkList>` possède un attribut obligatoire "`level`", dont les valeurs possibles sont "`sentence`" (annotations au niveau de phrases), "`token`" (celles au niveau de mots) et "`chunk`" (au niveau de segments).

3.2.5 L'élément `linkGroup`

Un `<linkGroup>` est un groupe de liens, dont les contenus sont extraits d'un même fragment d'un document. Un `<linkGroup>` contient plusieurs sous-éléments `<docPart>` qui indiquent les fragments des documents, suivis par une liste de `<link>` ou une liste de `<annotation>`. Tous les `<link>` ou `<annotation>` d'un `<linkGroup>` ont le niveau linguistique indiqué par le parent `<linkList>`. Les unités textuelles dans les `<link>` ou `<annotation>` se trouvent strictement dans les fragments indiqués par les `<docPart>`. Un `<linkGroup>` a un attribut obligatoire "`type`", dont les valeurs possibles sont "`alignment`" et "`annotation`". Si la valeur de "`type`" est "`alignment`", ce `<linkGroup>` ne peut pas contenir des sous-éléments `<annotation>` ; si cette valeur est "`annotation`", il ne peut pas contenir des `<link>`.

3.2.6 L'élément `docPart`

Un `<docPart>` indique un fragment d'un document. Il a trois attributs :
- `doc` : attribut obligatoire, indique un document. La valeur doit être une référence d'un *id* d'un élément `<docName>`.
- `beginPos` : attribut facultatif, indique la position du début de ce fragment dans le document.

– `endPos` : attribut facultatif, indique la position de la fin de ce fragment dans le document.

Un `<docPart>` est un élément vide. La présence de cette balise a pour objectif d'accélérer la recherche des informations dans les documents d'annotations volumineux.

3.2.7 L'élément `link`

Un `<link>` spécifie une unité textuelle d'un document et sa correspondance dans le document parallèle. Il inclut un sous-élément `<docSpan>` qui décrit une unité, et un autre sous-élément *facultatif* `<docSpan>` qui décrit l'unité correspondante dans l'autre document. L'absence du deuxième `<docSpan>` signifie un lien nul. Il faut remarquer que, dans un `<linkGroup>`, tous les sous-éléments `<docSpan>` doivent venir d'un fragment parmi les sous-éléments `<docPart>` de ce `<linkGroup>`. Un `<link>` possède deux attributs :

– `id` : attribut obligatoire, la valeur doit indiquer le niveau de l'unité alignée, par exemple `"align_tok_15"`.
– `certainty` : attribut facultatif, la valeur est un nombre entre 0 et 1, indiquant le niveau de confiance de l'annotateur sur ce `<link>`.

3.2.8 L'élément `annotation`

Un `<annotation>` encode des propriétés attachées aux unités. Par exemple, des analyses linguistiques peuvent permettre d'identifier son lemme, sa catégorie grammaticale, etc ; la relation entre un mot et des entrées de dictionnaires est un autre type d'informations. L'unité peut être un objet non textuel, par exemple une image ou une vidéo, où nous pouvons annoter la durée, la langue, etc. Un `<annotation>` contient un `<docSpan>`, et 0, 1 ou plusieurs `<mark>`.

Un `<annotation>` possède deux attributs :

– `id` : attribut obligatoire, l'identifiant de ce `<annotation>`.
– `type` : attribut obligatoire, qui encode le type de l'annotation. Les valeurs possibles sont `"gram"`, `"QE"`, `"URI"`. Le type `"gram"` est pour enregistrer les résultats obtenus par l'analyse syntaxique ; `"QE"` encode les résultats de l'estimation de qualité de traduction ; `"URI"` indique les liens vers les ressources externes, qui sont utilisés par exemple pour la désambiguïsation des mots.

La liste des types d'information est évolutive et pourra être complétée au fur de l'avancement du projet.

3.2.9 L'élément `docSpan`

Un élément `<docSpan>` permet d'identifier une unité à l'intérieur du document. Il possède trois attributs :

– `beginPos` : attribut facultatif, la valeur doit indiquer la position du début de l'unité.
– `endPos` : attribut facultatif, la valeur doit indiquer la position de la fin de l'unité.
– `context` : attribut facultatif, la valeur doit être une série des *ids* des `<link>` ou `<annotation>`, dans lesquels se trouvent les contextes de l'unité.

Nous pouvons mettre ou non le contenu textuel de l'unité dans l'élément `<docSpan>`.

3.2.10 L'élément `mark`

L'élément `<mark>` contient des informations relativement riches, car cet élément stocke les informations attachées aux unités. Souvent un `<annotation>` contient plusieurs `<mark>`. Il se peut à l'inverse qu'aucun `<mark>` ne figure dans un `<annotation>`, au cas où aucune information n'a été trouvée.

Un `<mark>` dispose de nombreux attributs :

- `certainty` : attribut facultatif, qui indique le niveau de confiance de l'annotateur sur ce `<mark>`. La valeur doit être un nombre entre 0 et 1.
- `cat` : attribut facultatif, utilisé dans les `<annotation>` de type `"gram"`. La valeur indique la catégorie du label linguistique. Les valeurs possibles sont `"POS"` (*Part Of Speech*) et `"lemma"` (le lemme), mais il sera possible d'étendre au besoin les possibilités des valeurs avec d'autres informations linguistiques. Nous pouvons se référer au ISOcat (ISO TC 37 Terminology and Other Language and Content Resources) [21] afin d'avoir une idée pour des développements possibles.
- `resource` : attribut facultatif, utilisé dans les `<annotation>` de type `"URI"`. La valeur indique la ressource externe. Par exemple `"babelnet"`, `"wordnet"`, `"wiktionnary"` etc.
- xml :lang : attribut facultatif, la valeur doit être un code de langue existant dans la norme ISO 639-1 [22]. Cet attribut peut être utilisé pour, par exemple, indiquer la langue des explications trouvées dans les ressources externes.
- entry : attribut facultatif, utilisé dans les `<annotation>` de type `"URI"`. La valeur est une entrée de la ressource externe.
- qescore : attribut facultatif, utilisé dans les `<annotation>` de type `"QE"`. La valeur est le score de l'estimation de qualité sur l'unité.
- method : attribut facultatif, utilisé dans les `<annotation>` de type `"QE"`. La valeur indique la méthode de l'estimation de qualité.

Pour les `<mark>` apparaissant dans les `<annotation>` de type `"gram"`, le contenu de l'élément `<mark>` doit être le label linguistique ; pour ceux de type `"URI"`, le contenu de l'élément `<mark>` doit être des informations associées au `<docSpan>`.

Un exemple complètement annoté correspondant aux deux premiers chapitres du livre de F. Cooper « The last of the Mohicans » est disponible sur le site du projet [23].

4 Conclusion

Nous avons dans ce document analysé les exigences du projet TransRead pour définir un formalisme permettant de stocker les document originaux et de représenter les annotations associées, en particulier les liens d'alignement. Notre solution contient deux aspects : pour les documents originaux, nous allons tous convertir au format du EPUB ; pour les annotations, nous avons proposé un format dans la section 3 qui est, à notre avis, conforme aux toutes les exigences.

Certainement, toutes les situations des traitements des documents ne peuvent être prévues. Des évolutions sont attendus au fur et à mesure des développements du projet. Néanmoins, ce format possède d'une grande flexibilité et extensibilité, qui rendra les modifications faciles et efficaces.

21. `http://www.isocat.org/`
22. `http://www.loc.gov/standards/iso639-2/php/code_list.php`
23. `http://transread.limsi.fr/Resources/`

Références

João de Almeida Varelas Graça, Joana Paulo Pardal, Luísa Coheur, and Diamantino António Caseiro. Building a golden collection of parallel multi-language word alignment. In *6th International Conference on Language Resources and Evaluation, LREC 2008*, 2008.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. Moses : Open source toolkit for statistical machine translation. In *Proc. Annual Meeting of the Association for Computational Linguistics (ACL), demonstration session*, pages 177–180, Prague, Czech Republic, 2007.

Patrik Lambert, Adrià Gispert, Rafael Banchs, and José B. Mariño. Guidelines for word alignment evaluation and manual alignment. *Language Resources and Evaluation*, 39 (4) :267–285, 2005. ISSN 1574-020X. doi : 10.1007/s10579-005-4822-5. URL `http://dx.doi.org/10.1007/s10579-005-4822-5`.

Dan Melamed. *Empirical methods for exploiting parallel texts*. The MIT Press, Cambridge, 2001. ISBN 0262133806.

I. Dan Melamed. Annotation style guide for the Blinker project. Technical Report IRCS-98-06, University of Pennsylvania Institute for Research in Cognitive Science, 1998.

Rada Mihalcea and Ted Pedersen. An evaluation exercise for word alignment. In *Proceedings of the HLT-NAACL 2003 Workshop on Building and using parallel texts : data driven machine translation and beyond - Volume 3*, HLT-NAACL-PARALLEL '03, pages 1–10, Stroudsburg, PA, USA, 2003. Association for Computational Linguistics. doi : 10.3115/1118905.1118906. URL `http://dx.doi.org/10.3115/1118905.1118906`.

Franz Josef Och and Hermann Ney. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1) :19–51, 2003.

Jörg Tiedemann. *Recycling Translations – Extraction of Lexical Data from Parallel Corpora and their Application in Natural Language Processing*. PhD thesis, Uppsala University, Uppsala, Sweden, 2003. URL `http://uu.diva-portal.org/smash/record.jsf?pid=diva2:163715`. Anna Sågvall Hein, Åke Viberg (eds) : Studia Linguistica Upsaliensia.

Jörg Tiedemann. *Bitext Alignment*. Number 14 in Synthesis Lectures on Human Language Technologies, Graeme Hirst (ed). Morgan & Claypool Publishers, 2011. URL `http://dx.doi.org/10.2200/S00367ED1V01Y201106HLT014`.

Jean Véronis, editor. *Parallel Text Processing*. Text, Speech and Language Technology. Kluwer Academic Publishers, 2000.

Martin Volk, Anne Göhring, Torsten Marek, and Yvonne Samuelsson. SMULTRON (version 3.0) — The Stockholm MULtilingual parallel TReebank. http ://www.cl.uzh.ch/research/paralleltreebanks_en.html, 2010.

5 Annexes

5.1 La DTD d'annotation

Listing 15 – La DTD d'annotation

```
<!--                                                                -->
<!--                                                                -->
<!--            La DTD d'annotation pour TransRead                  -->
<!--                                                                -->
<!--                        Version  1.1                            -->
<!--                                                                -->

<!ELEMENT trAnnot     (docList, linkList+)                           >
<!ATTLIST trAnnot
          version     CDATA                              #REQUIRED >

<!ELEMENT docList     (docName+)                                     >
<!ELEMENT docName     (#PCDATA)                                      >
<!ATTLIST docName
          id          ID                                 #REQUIRED
          xml:lang    CDATA                              #IMPLIED  >

<!ELEMENT linkList    (linkGroup+)                                   >
<!ATTLIST linkList
          level       (sentence | token | chunk)         #REQUIRED >

<!ELEMENT linkGroup   (docPart+, (link+ | annotation+))              >
<!ATTLIST linkGroup
          type        (alignment | annotation)           #REQUIRED >

<!ELEMENT docPart     EMPTY                                          >
<!ATTLIST docPart
          doc         IDREF                              #REQUIRED
          beginPos    CDATA                              #IMPLIED
          endPos      CDATA                              #IMPLIED  >

<!ELEMENT link        (docSpan, docSpan?)                            >
<!ATTLIST link
          certainty   CDATA                              #IMPLIED
          id          ID                                 #REQUIRED >

<!ELEMENT annotation  (docSpan, mark*)                               >
<!ATTLIST annotation
          type        (gram | URI | QE)                  #REQUIRED
          id          ID                                 #REQUIRED >

<!ELEMENT docSpan     (#PCDATA)                                      >
```

```
<!ATTLIST docSpan
          beginPos    CDATA                          #REQUIRED
          endPos      CDATA                          #REQUIRED
          context     IDREFS                         #IMPLIED  >

<!ELEMENT mark        (#PCDATA)                                 >
<!ATTLIST mark
          cat         (POS | lemma)                  #IMPLIED
          resource    CDATA                          #IMPLIED
          xml:lang    CDATA                          #IMPLIED
          entry       CDATA                          #IMPLIED
          certainty   CDATA                          #IMPLIED
          qescore     CDATA                          #IMPLIED
          method      CDATA                          #IMPLIED  >
```

5.2 Le schéma XML

Listing 16 – Le schéma XML d'annotation

```
<?xml version="1.0" encoding="utf-8"?>

<!--                                                              -->
<!--                                                              -->
<!--        Le schéma XML d'annotation pour TransRead             -->
<!--                                                              -->
<!--                   Version  1.1                               -->
<!--                                                              -->


<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  targetNamespace="http://transread.limsi.fr"
  xmlns="http://transread.limsi.fr"
  elementFormDefault="qualified">

  <xsd:import namespace="http://www.w3.org/XML/1998/namespace"
              schemaLocation="http://www.w3.org/2001/xml.xsd"/>

  <xsd:simpleType name="certaintytype">
    <xsd:restriction base="xsd:decimal">
      <xsd:minInclusive value="0.0"/>
      <xsd:maxInclusive value="1.0"/>
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:simpleType name="qescoretype">
    <xsd:restriction base="xsd:decimal">
      <xsd:minInclusive value="0.0"/>
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:simpleType name="shortpostype">
    <xsd:restriction base="xsd:string">
      <xsd:pattern value="([0-9]+\.)+[0-9]+-[0-9]+"/>
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:simpleType name="tmppostype">
    <xsd:union memberTypes="xsd:IDREF shortpostype"/>
  </xsd:simpleType>

  <xsd:simpleType name="postype">
    <xsd:restriction>
      <xsd:simpleType>
```

```
      <xsd:list itemType="tmppostype"/>
    </xsd:simpleType>
    <xsd:pattern value="[0-9a-zA-Z_-]+ ([0-9]+\.)+[0-9]+-[0-9]+"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="cattype">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="POS"/>
    <xsd:enumeration value="lemma"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="annotoption">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="gram"/>
    <xsd:enumeration value="URI"/>
    <xsd:enumeration value="QE"/>
  </xsd:restriction>
</xsd:simpleType>

  <xsd:simpleType name="linkgroupoption">
    <xsd:restriction base="xsd:string">
      <xsd:enumeration value="alignment"/>
      <xsd:enumeration value="annotation"/>
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:simpleType name="methodtype">
    <xsd:restriction base="xsd:string">
      <xsd:enumeration value="method1"/>
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:simpleType name="leveltype">
    <xsd:restriction base="xsd:string">
      <xsd:enumeration value="sentence"/>
      <xsd:enumeration value="token"/>
      <xsd:enumeration value="chunk"/>
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:complexType name="marktype">
    <xsd:simpleContent>
      <xsd:extension base="xsd:string">
        <xsd:attribute name="cat" type="cattype"/>
        <xsd:attribute name="resource" type="xsd:string"/>
        <xsd:attribute ref="xml:lang"/>
```

```
        <xsd:attribute name="entry" type="xsd:string"/>
        <xsd:attribute name="certainty" type="certaintytype"/>
        <xsd:attribute name="qescore" type="qescoretype"/>
        <xsd:attribute name="method" type="methodtype"/>
      </xsd:extension>
    </xsd:simpleContent>
  </xsd:complexType>

<xsd:complexType name="docspantype">
  <xsd:simpleContent>
    <xsd:extension base="xsd:string">
      <xsd:attribute name="beginPos" type="postype" use="required"/>
      <xsd:attribute name="endPos" type="postype" use="required"/>
      <xsd:attribute name="context" type="xsd:IDREFS"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>

<xsd:complexType name="annotationtype">
  <xsd:sequence>
    <xsd:element name="docSpan" type="docspantype"/>
    <xsd:element name="mark" type="marktype" maxOccurs="unbounded"/>
  </xsd:sequence>
  <xsd:attribute name="type" type="annotoption" use="required"/>
  <xsd:attribute name="id" type="xsd:ID" use="required"/>
</xsd:complexType>

<xsd:complexType name="linktype">
  <xsd:sequence>
    <xsd:element name="docSpan" type="docspantype" minOccurs="1" maxOccurs="2"/>
  </xsd:sequence>
  <xsd:attribute name="id" type="xsd:ID" use="required"/>
  <xsd:attribute name="certainty" type="certaintytype"/>
</xsd:complexType>

<xsd:complexType name="docparttype">
  <xsd:attribute name="doc" type="xsd:IDREF" use="required"/>
  <xsd:attribute name="beginPos" type="postype"/>
  <xsd:attribute name="endPos" type="postype"/>
</xsd:complexType>

<xsd:complexType name="linkgrouptype">
  <xsd:sequence>
    <xsd:element name="docPart" type="docparttype" maxOccurs="2"/>
    <xsd:choice>
      <xsd:element name="link" type="linktype" maxOccurs="unbounded"/>
      <xsd:element name="annotation" type="annotationtype" maxOccurs="unbounded"/>
    </xsd:choice>
```

```
    </xsd:sequence>
    <xsd:attribute name="type" type="linkgroupoption" use="required"/>
  </xsd:complexType>

  <xsd:complexType name="linklisttype">
    <xsd:sequence>
      <xsd:element name="linkGroup" type="linkgrouptype" maxOccurs="unbounded"/>
    </xsd:sequence>
    <xsd:attribute name="level" type="leveltype"/>
  </xsd:complexType>

  <xsd:complexType name="docnametype">
    <xsd:simpleContent>
      <xsd:extension base="xsd:string">
        <xsd:attribute name="id" type="xsd:ID" use="required"/>
        <xsd:attribute ref="xml:lang"/>
      </xsd:extension>
    </xsd:simpleContent>
  </xsd:complexType>

  <xsd:complexType name="doclisttype">
    <xsd:sequence>
      <xsd:element name="docName" type="docnametype" maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:complexType name="trannottype">
    <xsd:sequence>
      <xsd:element name="docList" type="doclisttype"/>
      <xsd:element name="linkList" type="linklisttype" maxOccurs="unbounded"/>
    </xsd:sequence>
    <xsd:attribute name="version" type="xsd:decimal" use="required"/>
  </xsd:complexType>

  <xsd:element name="trAnnot" type="trannottype"/>
</xsd:schema>
```

Bibliography

Lauriane Aufrant, Guillaume Wisniewski, and François Yvon. Cross-lingual and Supervised Models for Morphosyntactic Annotation: a Comparison on Romanian. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation*, 2016.

Necip Fazil Ayan and Bonnie J. Dorr. A maximum entropy approach to combining word alignments. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 96–103, 2006.

Nguyen Bach, Fei Huang, and Yaser Al-Onaizan. Goodness: A method for measuring machine translation confidence. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 211–219, 2011.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. In *Proceedings of the International Conference on Learning Representations*, 2015.

Eduard Barbu. Spotting false translation segments in translation memories. In *Proceedings of the Workshop Natural Language Processing for Translation Memories*, pages 9–16, 2015.

Roberto Basili, Maria Teresa Pazienza, and Paola Velardi. Computational lexicons: the neat examples and the odd exemplars. In *Proceedings of the Third Conference on Applied Natural Language Processing*, pages 96–103, 1992.

Frédérique Bisson and Christian Fluhr. Sentence alignment in bilingual corpora based on crosslingual querying. In *Proceedings of the Conference on Recherche d'Information Assistée par Ordinateur*, pages 529–542, 2000.

John Blatz, Erin Fitzgerald, George F. Foster, Simona Gandrabur, Cyril Goutte, Alex Kulesza, Alberto Sanchis, and Nicola Ueffing. Confidence estimation for machine translation. In *Proceedings of the 20th International Conference on Computational Linguistics*, 2004.

Phil Blunsom and Trevor Cohn. Discriminative word alignment with conditional random fields. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 65–72, 2006.

Julien Bourdaillet, Stéphane Huet, Fabrizio Gotti, Guy Lapalme, and Philippe Langlais. Enhancing the bilingual concordancer transsearch with word-level alignment. In *Proceedings of the 22nd Canadian Conference on Artificial Intelligence*, pages 27–38, 2009.

Fabienne Braune and Alexander Fraser. Improved unsupervised sentence alignment for symmetrical and asymmetrical parallel corpora. In *Proceedings of the 23rd International Conference on Computational Linguistics*, pages 81–89, 2010.

Chris Brew, David McKelvie, and Buccleuch Place. Word-pair extraction for lexicography. In *Proceedings of the Second International Conference on New Methods in Language Processing*, pages 45–55, 1996.

Peter F. Brown, Jennifer C. Lai, and Robert L. Mercer. Aligning sentences in parallel corpora. In *Proceedings of the 29th Annual Meeting of the Association for Computational Linguistics*, pages 169–176, 1991.

Peter F. Brown, Vincent J. Della Pietra, Stephen A. Della Pietra, and Robert L. Mercer. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19(2):263–311, 1993.

David Burkett and Dan Klein. Fast inference in phrase extraction models with belief propagation. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 29–38, 2012.

Yin-Wen Chang, Alexander M. Rush, John DeNero, and Michael Collins. A constrained viterbi relaxation for bidirectional word alignment. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, pages 1481–1490, 2014.

Kuang-hua Chen and Hsin-Hsi Chen. A part-of-speech-based alignment algorithm. In *Proceedings of the 15th Conference on Computational Linguistics*, pages 166–171, 1994.

Stanley F. Chen. Aligning sentences in bilingual corpora using lexical information. In *Proceedings of the 31st Annual Meeting of hte Association for Computational Linguistics*, pages 9–16, 1993.

Yun-Chuang Chiao, Olivier Kraif, Dominique Laurent, Thi Minh Huyen Nguyen, Nasredine Semmar, François Stuck, Jean Véronis, and Wajdi Zaghouani. Evaluation of multilingual text alignment systems: the ARCADE II project. In *Proceedings of the 5th International Conference on Language Resources and Evaluation*, 2006.

Jacob Cohen. A Coefficient of Agreement for Nominal Scales. *Educational and Psychological Measurement*, 20(1):37–46, 1960.

Fabien Cromières and Sadao Kurohashi. An alignment algorithm using belief propagation and a structure-based distortion model. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, pages 166–174, 2009.

Aron Culotta and Andrew McCallum. Confidence estimation for information extraction. In *Proceedings of HLT-NAACL 2004: Short Papers*, pages 109–112, 2004.

Mark W. Davis, Ted E. Dunning, and William C. Ogden. Text alignment in the real world: Improving alignments of noisy translations using common lexical features, string matching strategies and n-gram comparisons. In *Proceedings of the Seventh Conference on European Chapter of the Association for Computational Linguistics*, pages 67–74, 1995.

John DeNero and Klaus Macherey. Model-based aligner combination using dual decomposition. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 420–429, 2011.

John Sturdy DeNero. *Discriminative Alignment Models For Statistical Machine Translation*. PhD thesis, University of California, Berkeley, 2012.

Yonggang Deng and William Byrne. Hmm word and phrase alignment for statistical machine translation. *IEEE Transactions on Audio, Speech, and Language Processing*, 16(3): 494–507, 2008.

Yonggang Deng, Shankar Kumar, and William Byrne. Segmentation and alignment of parallel text for statistical machine translation. *Natural Language Engineering*, 13(03): 235–260, 2007.

Mona Diab and Philip Resnik. An unsupervised method for word sense tagging using parallel corpora. In *Proceedings of 40th Annual Meeting of the Association for Computational Linguistics*, pages 255–262, 2002.

Ludmila Dimitrova, Tomaz Erjavec, Nancy Ide, Heiki-Jaan Kaalep, Vladimir Petkevic, and Dan Tufis. Multext-east: Parallel and comparable corpora and lexicons for six central and eastern european languages. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics*, pages 315–319, 1998.

Chris Dyer, Jonathan H. Clark, Alon Lavie, and Noah A. Smith. Unsupervised word alignment with arbitrary features. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, pages 409–419, 2011.

Chris Dyer, Victor Chahuneau, and Noah A. Smith. A simple, fast, and effective reparameterization of ibm model 2. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 644–648, 2013a.

Chris Dyer, Victor Chahuneau, and Noah A. Smith. A simple, fast, and effective reparameterization of IBM model 2. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 644–648, 2013b.

Steffen Eger. Multiple many-to-many sequence alignment for combining string-valued variables: A G2P experiment. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, pages 909–919, 2015.

Manaal Faruqui and Chris Dyer. Improving vector space word representations using multilingual correlation. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 462–471, 2014.

Tom Fawcett. An introduction to ROC analysis. *Pattern Recognition Letters*, 27(8):861–874, 2006.

Alexander Fraser and Daniel Marcu. Measuring word alignment quality for statistical machine translation. *Computational Linguistics*, 33(3):293–303, 2007.

Pascale Fung and Kenneth Ward Church. K-vec: A new approach for aligning parallel texts. In *Proceedings of the 15th Conference on Computational Linguistics*, pages 1096–1102, 1994.

William A. Gale and Kenneth W. Church. A program for aligning sentences in bilingual corpora. In *Proceedings of the 29th Annual Meeting of the Association for Computational Linguistics*, pages 177–184, 1991.

William A. Gale and Kenneth W. Church. A program for aligning sentences in bilingual corpora. *Computational Linguistics*, 19(1):75–102, 1993.

Kuzman Ganchev, João V. Graça, and Ben Taskar. Better alignments = better translations? In *Proceedings of the 2008 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 986–993, 2008.

Simona Gandrabur and George F. Foster. Confidence estimation for translation prediction. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 95–102, 2003.

Simona Gandrabur, George F. Foster, and Guy Lapalme. Confidence estimation for NLP applications. *ACM Transactions on Speech and Language Processing*, 3(3):1–29, 2006.

Qin Gao and Stephan Vogel. Parallel implementations of word alignment tool. In *Software Engineering, Testing, and Quality Assurance for Natural Language Processing*, pages 49–57, 2008.

Qin Gao and Stephan Vogel. Consensus versus expertise : A case study of word alignment with mechanical turk. In *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, pages 30–34, 2010.

Ulrich Germann. Yawat: Yet Another Word Alignment Tool. In *Proceedings of the ACL-08: HLT Demo Session*, pages 20–23, 2008.

Cyril Goutte, Marine Carpuat, and George F. Foster. The impact of sentence alignment errors on phrase-based machine translation performance. In *Proceedings of the 2012 Conference of the Association for Machine Translation in the Americas*, 2012.

Stephan Gouws, Yoshua Bengio, and Greg Corrado. Bilbowa: Fast bilingual distributed representations without word alignments. In *Proceedings of the 32nd International Conference on Machine Learning*, 2015.

João Graça, Joana Paulo Pardal, Luísa Coheur, and Diamantino Caseiro. Multi-language word alignments annotation guidelines. Technical report, L^2F – INESC-ID Lisboa/IST, 2008.

João Graça, Kuzman Ganchev, and Ben Taskar. Expectation maximization and posterior constraints. In *NIPS*, pages 569–576, 2007.

João Graça, Joana Paulo Pardal, Luísa Coheur, and Diamantino Caseiro. Building a golden collection of parallel multi-language word alignment. In *Proceedings of the 6th International Conference on Language Resources and Evaluation*, 2008.

João Graça, Kuzman Ganchev, and Ben Taskar. Learning tractable word alignment models with complex constraints. *Computational Linguistics*, 36(3):481–504, 2010.

Masahiko Haruno and Takefumi Yamazaki. High-performance bilingual text alignment using statistical and dictionary information. In *Proceedings of the 34th Annual Meeting of the Association for Computational Linguistics*, pages 131–138, 1996.

Maria Holmqvist and Lars Ahrenberg. A gold standard for english-swedish word alignment. In *Proceedings of the 18th Nordic Conference of Computational Linguistics*, pages 106–113, 2011.

Fei Huang. Confidence measure for word alignment. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 932–940, 2009.

Hui Jiang. Confidence measures for speech recognition: A survey. *Speech communication*, 45(4):455–470, 2005.

Hidetaka Kamigaito, Taro Watanabe, Hiroya Takamura, and Manabu Okumura. Unsupervised word alignment using frequency constraint in posterior regularized EM. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 153–158, 2014.

Max Kaufmann. JMaxAlign: A maximum entropy parallel sentence alignment tool. In *Proceedings of the 24th International Conference on Computational Linguistics*, pages 277–288, 2012.

Martin Kay and Martin Röscheisen. Text-translation alignement. *Computational Linguistics*, 19(1):121–142, 1993.

Sue J. Ker and Jason S. Chang. A class-based approach to word alignment. *Computational Linguistics*, 23(2):313–343, 1997.

Judith Klavans and Evelyne Tzoukermann. The bicord system: Combining lexical information from bilingual corpora and machine readable dictionaries. In *Proceedings of the 13th Conference on Computational Linguistics*, pages 174–179, 1990.

Reinhard Kneser and Hermann Ney. Improved backing-off for m-gram language modeling. In *1995 International Conference on Acoustics, Speech, and Signal Processing*, volume 1, pages 181–184, 1995.

Kevin Knight and Jonathan Graehl. Machine transliteration. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics*, pages 128–135, 1997.

Philipp Koehn. Europarl: A parallel corpus for statistical machine translation. In *Proceedings of the Tenth Machine Translation Summit*, pages 79–86, 2005.

Philipp Koehn. *Statistical Machine Translation*. Cambridge University Press, Cambridge, England, 1st edition, 2010.

Philipp Koehn, Franz Josef Och, and Daniel Marcu. Statistical phrase-based translation. In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 48–54, 2003.

Olivier Kraif and Agnès Tutin. Using a bilingual annotated corpus as a writing aid: an application for academic writing for EFL users. In *Corpora, Language, Teaching, and*

Resources: From Theory to Practice. Selected papers from TaLC7, volume 12. The Peter Lang Publishing Group, 2011.

Shankar Kumar, Franz J. Och, and Wolfgang Macherey. Improving word alignment with bridge languages. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 42–50, 2007.

Andrey Kutuzov. Improving English-Russian sentence alignment through POS tagging and Damerau-Levenshtein distance. In *Proceedings of the 4th Biennial International Workshop on Balto-Slavic Natural Language Processing*, pages 63–68, 2013.

Simon Lacoste-Julien, Ben Taskar, Dan Klein, and Michael I. Jordan. Word alignment via quadratic assignment. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 112–119, 2006.

Patrik Lambert, Adrià De Gispert, Rafael Banchs, and José Mariño. Guidelines for word alignment evaluation and manual alignment. *Language Resources and Evaluation*, 39(4): 267–285, 2005.

Patrik Lambert, Simon Petitrenaud, Yanjun Ma, and Andy Way. Statistical analysis of alignment characteristics for phrase-based machine translation. In *Proceedings of the 14th Annual Conference of the European Association for Machine Translation*, 2010.

Fethi Lamraoui and Philippe Langlais. Yet another fast, robust and open source sentence aligner. time to reconsider sentence alignment? In *Proceedings of the XIV Machine Translation Summit*, pages 77–84, 2013.

Philippe Langlais. A System to Align Complex Bilingual Corpora. Technical report, CTT, KTH, 1998.

Philippe Langlais, Michel MSimard, and Jean Véronis. Methods and practical issues in evaluating alignment techniques. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics*, pages 711–717, 1998a.

Philippe Langlais, Michel Simard, Jean Véronis, S Armstrong, P. Bonhomme, F. Débili, Pierre Isabelle, E. Soussi, and P Théron. The ARCADE: A Cooperative Research Project on Bilingual Text Alignment. In *Proceedings of the First International Conference On Language Resources and Evaluation*, 1998b.

Adrien Lardilleux, François Yvon, and Yves Lepage. Hierarchical sub-sentential alignment with Anymalign. In *Proceedings of the Annual Meeting of the European Association for Machine Translation*, pages 279–286, 2012.

Adrien Lardilleux, François Yvon, and Yves Lepage. Generalizing sampling-based multilingual alignment. *Machine Translation*, 27(1):1–23, 2013.

Charlotte Lecluze. *Alignement de documents multilingues sans présupposé de parallélisme*. PhD thesis, Université de Caen Basse-Normandie, 2011.

Peng Li, Maosong Sun, and Ping Xue. Fast-champollion: A fast and robust sentence alignment algorithm. In *Proceedings of the 23rd International Conference on Computational Linguistics*, pages 710–718, 2010a.

Xuansong Li, Xiaoyi Ma, Stephen Grimes, Stephanie Strassel, Gary Krug, and Dalal Zakhary. Word alignment for improved machine translation. In *Handbook of Natural Language Processing and Machine Translation: DARPA Global Autonomous Language Exploitation*, chapter Data Acquisition and Linguistic Resources, pages 31–39. Springer, 2009.

Xuansong Li, Niyu Ge, Stephen Grimes, Stephanie M. Strassel, and Kazuaki Maeda. Enriching word alignment with linguistic tags. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation*, pages 2189–2195, 2010b.

Xuansong Li, Stephanie Strassel, Stephen Grimes, Safa Ismael, Mohamed Maamouri, Ann Bies, and Nianwen Xue. Parallel aligned treebanks at ldc: New challenges interfacing existing infrastructures. In *Proceedings of the 8th International Conference on Language Resources and Evaluation*, pages 1848–1855, 2012.

Percy Liang, Ben Taskar, and Dan Klein. Alignment by agreement. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 104–111, June 2006.

Pierre Lison and Jörg Tiedemann. Opensubtitles2016: Extracting large parallel corpora from movie and TV subtitles. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation*, pages 923–929, 2016.

Chunyang Liu, Yang Liu, Maosong Sun, Huanbo Luan, and Heng Yu. Generalized agreement for bidirectional word alignment. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1828–1836, 2015.

Dong C. Liu and Jorge Nocedal. On the limited memory BFGS method for large scale optimization. *Mathematical Programming*, 45:503–528, 1989.

Wei Liu, Zhipeng Chang, and William John Teahan. Experiments with a PPM compression-based method for english-chinese bilingual sentence alignment. In *Statistical Language and Speech Processing - Second International Conference*, pages 70–81, 2014.

Yang Liu, Qun Liu, and Shouxun Lin. Log-linear models for word alignment. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics*, pages 459–466, 2005.

Adam Lopez and Philip Resnik. Word-based alignment, phrase-based translation: What's the link? In *Proceedings of the 7th Conference of the Association for Machine Translation in the Americas*, pages 90–99, 2006.

Ngoc Quang Luong, Laurent Besacier, and Benjamin Lecouteux. Word confidence estimation for SMT N-best list re-ranking. In *Proceedings of the EACL 2014 Workshop on Humans and Computer-assisted Translation*, pages 1–9, 2014.

Xiaoyi Ma. Champollion: A robust parallel text sentence aligner. In *Proceedings of the 5th International Conference on Language Resources and Evaluation*, 2006.

Elliot Macklovitch. Using bi-textual alignment for translation validation: the TransCheck system. In *Proceedings of the Conference of the Association for Machine Translation in the Americas*, pages 157–168, 1994a.

Elliott Macklovitch. Using bi-textual alignment for translation validation: the transcheck system. In *Proceedings of the First Conference of the Association for Machine Translation in the Americas*, 1994b.

Daniel Marcu and William Wong. A phrase-based, joint probability model for statistical machine translation. In *Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing*, pages 133–139, 2002.

Joel Martin, Rada Mihalcea, and Ted Pedersen. Word alignment for languages with scarce resources. In *Proceedings of the ACL Workshop on Building and Using Parallel Texts*, pages 65–74, Ann Arbor, Michigan, June 2005. Association for Computational Linguistics.

I. Dan Melamed. Automatic detection of omissions in translations. In *Proceedings of the 16th Conference on Computational Linguistics*, pages 764–769, 1996.

I. Dan Melamed. Manual annotation of translational equivalence : The blinker project. Technical report, Departement of Computer and Information Science, University of Pennsylvania, 1998a.

I. Dan Melamed. Annotation style guide for the blinker project. Technical report, Department of Computer and Information Science, University of Pennsylvania, 1998b.

I. Dan Melamed. Bitext maps and alignment via pattern recognition. *Computational Linguistics*, 25:107–130, 1999.

I. Dan Melamed. Models of translational equivalence among words. *Computational Linguistics*, 26(2):221–249, 2000.

Coskun Mermer and Murat Saraclar. Bayesian word alignment for statistical machine translation. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 182–187, 2011.

Rada Mihalcea and Ted Pedersen. An evaluation exercise for word alignment. In *Proceedings of the HLT-NAACL 2003 Workshop on Building and Using Parallel Texts: Data Driven Machine Translation and Beyond*, pages 1–10, 2003.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems 26*, pages 3111–3119. 2013.

Ruslan Mitkov, Viktor Pekar, Dimitar Blagoev, and Andrea Mulloni. Methods for extracting and classifying pairs of cognates and false friends. *Machine Translation*, 21(1):29–53, 2007.

Robert C. Moore. Fast and accurate sentence alignment of bilingual corpora. In *Proceedings of the Annual Meeting of tha Association for Machine Translation in the Americas*, pages 135–144, 2002.

Robert C. Moore. Improving IBM word-alignment model 1. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, pages 519—-526, 2004.

Robert C. Moore. A discriminative framework for bilingual word alignment. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 81–88, 2005.

Éva Mújdricza-Maydt, Huiqin Köerkel-Qu, Stefan Riezler, and Sebastian Padó. High-precision sentence alignment by bootstrapping from wood standard annotations. *The Prague Bulletin of Mathematical Linguistics*, 99:5–16, 2013.

Dragos Stefan Munteanu and Daniel Marcu. Improving machine translation performance by exploiting non-parallel corpora. *Computational Linguistics*, 31(4):477–504, 2005.

Kevin P. Murphy, Yair Weiss, and Michael I. Jordan. Loopy belief propagation for approximate inference: An empirical study. In *Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence*, pages 467–475, 1999.

John Nerbonne. *Parallel Texts in Computer-Assisted Language Learning*, chapter 15, pages 354–369. Text Speech and Language Technology Series. Springer Netherlands, 2000.

Jan Niehues and Stephan Vogel. Discriminative word alignment via alignment matrix modeling. In *Proceedings of the Third Workshop on Statistical Machine Translation*, pages 18–25, 2008.

Franz Josef Och and Hermann Ney. A comparison of alignment models for statistical machine translation. In *Proceedings of the 18th International Conference on Computational Linguistics*, pages 1086–1090, 2000.

Franz Josef Och and Hermann Ney. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51, 2003.

Harris Papageorgiou, Lambros Cranias, and Stelios Piperidis. Automatic alignment in parallel corpora. In *Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics*, pages 334–336, 1994.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 311–318, 2002.

Clément Pillias and Pierre Cubaud. Bilingual reading experiences: What they could be and how to design for them. In *Proceedings of the 15th IFIP TC 13 International Conference on Human-Computer Interaction*, pages 531–549, 2015.

Emmanuel Prochasson and Pascale Fung. Rare word translation extraction from aligned comparable documents. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 1327–1335, 2011.

Emily Prud'hommeaux and Brian Roark. Graph-based word alignment for clinical language evaluation. *Computational Linguistics*, 41(4):549–578, 2015.

Xiaojun Quan, Chunyu Kit, and Yan Song. Non-monotonic sentence alignment via semisupervised learning. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, pages 622–630, 2013.

Sylvain Raybaud, Caroline Lavecchia, David Langlois, and Kamel Smaïli. Word- and sentence-level confidence measures for machine translation. In *Proceedings of the 13th Annual Meeting of the European Association for Machine Translation*, pages 104–111, 2009.

Alexander M. Rush, David Sontag, Michael Collins, and Tommi Jaakkola. On dual decomposition and linear programming relaxations for natural language processing. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 1–11, 2010.

Matthew Stephen Seigel. *Confidence Estimation for Automatic Speech Recognition Hypotheses*. PhD thesis, University of Cambridge, 2013.

Rico Sennrich and Martin Volk. MT-based sentence alignment for OCR-generated parallel texts. In *The Ninth Conference of the Association for Machine Translation in the Americas*, 2010.

Michel Simard. The BAF: a corpus of English-French bitext. In *Proceedings of the First International Conference on Language Resources and Evaluation*, pages 489–494, 1998.

Michel Simard. Text-translation alignment: Three languages are better than two. In *Proceedings of the 1999 Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, 1999.

Michel Simard and Philippe Langlais. Statistical translation alignment with compositionality constraints. In *Proceedings of the HLT-NAACL 2003 Workshop on Building and Using Parallel Texts: Data Driven Machine Translation and Beyond*, pages 19–22, 2003.

Michel Simard and Pierre Plamondon. Bilingual sentence alignment: Balancing robustness and accuracy. *Machine Translation*, 13(1):59–80, 1998.

Michel Simard, George F. Foster, and Pierre Isabelle. Using cognates to align sentences in bilingual corpora. In *Proceedings of the 1993 Conference of the Centre for Advanced Studies on Collaborative Research*, pages 1071–1082, 1993a.

Michel Simard, George F. Foster, and François Perrault. Transsearch: A bilingual concordance tool. Technical report, Centre for Information Technology Innovation, 1993b.

Andrei Simion, Michael Collins, and Cliff Stein. A convex alternative to IBM model 2. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1574–1583, 2013.

Andrei Simion, Michael Collins, and Cliff Stein. On a strictly convex IBM model 1. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 221–226, 2015.

Frank Smadja, Kathleen R. McKeown, and Vasileios Hatzivassiloglou. Translating collocations for bilingual lexicons: A statistical approach. *Computational Linguistics*, 22(1): 1–38, 1996.

Jason R. Smith, Chris Quirk, and Kristina Toutanova. Extracting parallel sentences from comparable corpora using document level alignment. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 403–411, 2010.

Noah A. Smith Smith and Michael E. Jahr. Cairo: An alignment visualization tool. In *Proceedings of the Second International Conference on Language Resources and Evaluation*, 2000.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems 27*, pages 3104–3112, 2014.

Charles Sutton. *Efficient Training Methods for Conditional Random Fields.* PhD thesis, University of Massachusetts, 2008.

Oscar Täckström, Dipanjan Das, Slav Petrov, Ryan McDonald, and Joakim Nivre. Token and type constraints for cross-lingual part-of-speech tagging. *Transactions of the Association for Computational Linguistics*, 1:1–12, 2013.

Akihiro Tamura, Taro Watanabe, and Eiichiro Sumita. Recurrent neural networks for word alignment model. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, pages 1470–1480, 2014.

Ben Taskar, Lacoste-Julien Simon, and Klein Dan. A discriminative matching approach to word alignment. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 73–80, 2005.

Jörg Tiedemann. Combining clues for word alignment. In *Proceedings of the Tenth Conference on European Chapter of the Association for Computational Linguistics*, pages 339–346, 2003a.

Jörg Tiedemann. *Recycling Translations – Extraction of Lexical Data from Parallel Corpora and their Application in Natural Language Processing.* PhD thesis, Uppsala University, 2003b.

Jörg Tiedemann. *Bitext Alignment.* Number 14 in Synthesis Lectures on Human Language Technologies. Morgan & Claypool Publishers, 2011.

Christoph Tillmann and Sanjika Hewavitharana. A unified alignment algorithm for bilingual data. *Natural Language Engineering*, 19:33–60, 2013.

Nadi Tomeh. *Discriminative Alignment Models For Statistical Machine Translation.* PhD thesis, Université Paris-Sud, 2012.

Nadi Tomeh, Alexandre Allauzen, and François Yvon. Maximum-entropy word alignment and posterior-based phrase extraction for machine translation. *Machine Translation*, 28 (1):19–56, 2014.

Kristina Toutanova and Michel Galley. Why initialization matters for IBM model 1: Multiple optima and non-strict convexity. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 461–466, 2011.

Nicola Ueffing and Hermann Ney. Word-level confidence estimation for machine translation. *Computational Linguistics*, 33(1):9–40, 2007.

Jakob Uszkoreit, Jay M. Ponte, Ashok C. Popat, and Moshe Dubiner. Large scale parallel document mining for machine translation. In *Proceedings of the 23rd International Conference on Computational Linguistics*, pages 1101–1109, 2010.

Takehito Utsuro, Hiroshi Ikeda, Masaya Yamane, Yuji Matsumoto, and Makoto Nagao. Bilingual text, matching using bilingual dictionary and statistics. In *Proceedings of the 15th Conference on Computational Linguistics*, pages 1076–1082, 1994.

Dániel Varga, László Németh, Péter Halácsy, András Kornai, Viktor Trón, and Viktor Nagy. Parallel corpora for medium density languages. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing*, pages 590–596, 2005.

Ashish Vaswani, Liang Huang, and David Chiang. Smaller alignment models for better translations: Unsupervised word alignment with the l0-norm. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*, pages 311–319, 2012.

Jean Véronis and Philippe Langlais. Evaluation of Parallel Text Alignment Systems. In *Parallel Text Processing*, Text Speech and Language Technology Series, chapter X, pages 369–388. Springer Netherlands, 2000.

Stephan Vogel, Hermann Ney, and Christoph Tillmann. HMM-based word alignment in statistical translation. In *Proceedings of the 16th Conference on Computational Linguistics*, pages 836–841, 1996.

Martin Volk, Sofia Gustafson-Capková, Joakim Lundborg, Torsten Marek, Yvonne Samuelsson, and Frida Tidström. Xml-based phrase alignment in parallel treebanks. In *Proceedings of the 5th Workshop on NLP and XML: Multi-Dimensional Markup in Natural Language Processing*, pages 93–96, 2006.

Martin J. Wainwright and Michael I. Jordan. Graphical models, exponential families, and variational inference. *Foundations and Trends in Machine Learning*, 1(1-2):1–305, 2008.

Xiaolin Wang, Masao Utiyama, Andrew Finch, Taro Watanabe, and Eiichiro Sumita. Leave-one-out word alignment without garbage collector effects. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1817–1827, 2015.

Guillaume Wisniewski, Anil Kumar Singh, and François Yvon. Quality estimation for machine translation: some lessons learned. *Machine Translation*, 27(3):213–238, 2013.

Guillaume Wisniewski, Nicolas Pécheux, Souhir Gahbiche-Braham, and François Yvon. Cross-lingual part-of-speech tagging through ambiguous learning. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1779–1785, 2014.

Dekai Wu. Aligning a parallel English-Chinese corpus statistically with lexical criteria. In *Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics*, pages 80–87, 1994.

Dekai Wu. Alignment. In *CRC Handbook of Natural Language Processing*, pages 367–408, 2010.

Deyi Xiong, Min Zhang, and Haizhou Li. Error detection for statistical machine translation using linguistic features. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 604–611, 2010.

Nan Yang, Shujie Liu, Mu Li, Ming Zhou, and Nenghai Yu. Word alignment modeling with context dependent deep neural network. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, pages 166–175, 2013.

Jonathan S. Yedidia, William T. Freeman, and Yair Weiss. Generalized belief propagation. In *Advances in Neural Information Processing Systems*, pages 689–695. 2001.

Qian Yu, Aurélien Max, and François Yvon. Aligning bilingual literary works: a pilot study. In *Proceedings of the NAACL-HLT 2012 Workshop on Computational Linguistics for Literature*, pages 36–44, Montréal, Canada, 2012a.

Qian Yu, Aurélien Max, and François Yvon. Revisiting sentence alignment algorithms for alignment visualization and evaluation. In *Proceedings of the Fifth Workshop on Building and Using Comparable Corpora*, 2012b.

François Yvon, Yong Xu, Marianna Apidianaki, Clément Pillias, and Pierre Cubaud. Transread: Designing a bilingual reading experience with machine translation technologies. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*, pages 27–31, 2016.

Hamed Zamani, Heshaam Faili, and Azadeh Shakery. Sentence alignment using local and global information. *Computer Speech and Language*, 39(C):88–107, 2016.

Hui Zhang and David Chiang. Kneser-Ney smoothing on expected counts. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, pages 765–774, 2014.

Will Y. Zou, Richard Socher, Daniel Cer, and Christopher D. Manning. Bilingual word embeddings for phrase-based machine translation. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1393–1398, 2013.

université PARIS-SACLAY
ÉCOLE DOCTORALE
Sciences et technologies de l'information et de la communication (STIC)

Titre : Mesures de confiance pour l'alignement et pour la traduction automatique

Mots clés : Mesure de Confiance, Estimation de Confiance, Alignement de Bi-textes, Alignement Phrastique, Alignement de Mots, Schème d'Annotation, Corpus de Référence, Traduction Automatique

Résumé : En linguistique informatique, la relation entre langues différentes est souvent étudiée via des techniques d'alignement automatique. De tels alignements peuvent être établis à plusieurs niveaux structurels. En particulier, les alignements de bi-textes aux niveaux phrastiques et sous-phrastiques constituent des sources importantes d'information pour diverses applications du Traitement Automatique du Language Naturel (TALN) moderne, la Traduction Automatique étant un exemple proéminent.
Cependant, le calcul effectif des alignements de bi-textes peut être une tâche compliquée. Les divergences entre les langues sont multiples, de la structure de discours aux constructions morphologiques. Les alignements automatiques contiennent, majoritairement, des erreurs nuisant aux performances des applications. Dans cette situation, deux pistes de recherche émergent. La première est de continuer à améliorer les techniques d'alignement. La deuxième vise à développer des mesures de confiance fiables qui permettent aux applications de sélectionner les alignements selon leurs besoins.

Les techniques d'alignement et l'estimation de confiance peuvent toutes les deux bénéficier d'alignements manuels. Des alignements manuels peuvent jouer un rôle de supervision pour entraîner des modèles, et celui des données d'évaluation. Pourtant, la création de telles données est elle-même une question importante, en particulier au niveau sous-phrastique, où les correspondances multilingues peuvent être implicites et difficiles à capturer.
Cette thèse étudie des moyens pour acquérir des alignements de bi-textes utiles, aux niveaux phrastiques et sous-phrastiques. Les contributions majeures incluent (a) nouveaux schèmes d'annotation pour collecter des données d'évaluation de reference pour l'alignement de bi-textes, ainsi que les corpus qui sont publiés; (b) deux systèmes d'alignement phrastique qui ouvrent nouvelles directions de recherche sur ce problème; (c) un jeu de mesures de confiance pour les liens d'alignement aux niveaux des phrases et des mots. Les contributions apportées dans cette thèse sont employées dans une application réelle: le développement d'une liseuse qui vise à faciliter la lecture des livres électroniques multilingues.

Title : Confidence measures for alignment and for machine translation

Keywords : Confidence Measure, Confidence Estimation, Bitext Alignment, Sentence Alignment, Word Alignment, Annotation Scheme, Reference Corpus, Machine Translation

Abstract : In computational linguistics, the relation between different languages is often studied through automatic alignment techniques. Such alignments can be established at various structural levels. In particular, sentential and sub-sentential bitext alignments constitute an important source of information in various modern Natural Language Processing (NLP) applications, a prominent one being Machine Translation (MT).
Effectively computing bitext alignments, however, can be a challenging task. Discrepancies between languages appear in various ways, from discourse structures to morphological constructions. Automatic alignments would, at least in most cases, contain noise harmful for the performance of application systems which use the alignments. To deal with this situation, two research directions emerge: the first is to keep improving alignment techniques; the second is to develop reliable confidence measures which enable application systems to selectively employ the alignments according to their needs.

Both alignment techniques and confidence estimation can benefit from manual alignments. Manual alignments can be used as both supervision examples to train scoring models and as evaluation materials. The creation of such data is, however, an important question in itself, particularly at sub-sentential levels, where cross-lingual correspondences can be only implicit and difficult to capture.
This thesis focuses on means to acquire useful sentential and sub-sentential bitext alignments. Our major contributions include (a) new annotation schemes for collecting reference evaluation data for bitext alignment, as well as the corpora, which are available for the public; (b) two sentence alignment systems that open new research directions for this problem; (c) a set of confidence measures for sentence-level and word-level alignment links. These contributions have been applied to a real world application: the development of a bilingual reading tool aimed at facilitating the reading in a foreign language.

Université Paris-Saclay
Espace Technologique / Immeuble Discovery
Route de l'Orme aux Merisiers RD 128 / 91190 Saint-Aubin, France