A structurally diverse minimal corpus for eliciting structural mappings between languages

Katharina Probst and Alon Lavie

Language Technologies Institute
Carnegie Mellon University
{kathrin,alavie}@cs.cmu.edu

Abstract. We describe an approach to creating a small but diverse corpus in English that can be used to elicit information about any target language. The focus of the corpus is on *structural* information. The resulting bilingual corpus can then be used for natural language processing tasks such as inferring transfer mappings for Machine Translation. The corpus is sufficiently small that a bilingual user can translate and word-align it within a matter of hours. We describe how the corpus is created and how its structural diversity is ensured. We then argue that it is not necessary to introduce a large amount of redundancy into the corpus. This is shown by creating an increasingly redundant corpus and observing that the information gained converges as redundancy increases.[1]

1 Introduction and Motivation

Field linguistics has long recognized elicitation as an important means to documenting languages [2], [1]. In our work, we address a similar problem: we elicit data for the purpose of creating bilingual corpora that can be used for natural language processing tasks. The problem of gathering bilingual data by eliciting a carefully constructed corpus has also been addressed by a small number of projects, e.g. by [9], [4]. In our work, we use the elicited data to learn transfer rules for Machine Translation that capture how structures and features map from one language into another. The learned rules function at run-time the same way a hand-written transfer grammar is used [7], [3].

Elicitation focuses on gathering information about different structures and different linguistic features. An elicitation corpus is then a set of sentences and phrases that a bilingual speaker translates, so that information about the target language (TL) can be gathered from the corpus. Why is this worthwhile? In our work, we aim at building MT systems for language pairs where an abundance of information is given for one of the languages (such as English), while we have access to neither a large bilingual corpus nor to other resources such as a parser for the other language. In such a case, elicitation can be used to obtain information about this language. However, we do not simply elicit *any* data: A naturally occurring corpus exhibits all the structure and feature phenomena of a

[1] This research was funded in part by NSF grant number IIS-0121631.

language; the disadvantage of using a naturally occurring corpus for elicitation is that it is highly redundant, especially in the most frequent phenomena (e.g. `NP`→ `DET ADJ N`). An elicitation corpus, on the other hand, is a condensed set of sentences, where each sentence pinpoints a different phenomenon of interest. Because the elicitation will contain some noise, a certain amount of redundancy should be built into the corpus.

In our previous work on elicitation, we focused mainly on feature information [6], [8]. The resulting corpus contains sequences such as `He has one daughter, He has one son, He has two daughters`, and `He has two sons.` Clearly, such a sequence of sentences will be crucial in detecting the marking of gender, number, case, and other features. It will however be redundant if the goal is to learn how the structure `S`→`NP VP` is expressed in the target language. Redundancy in itself is not necessarily detrimental and can actually help in learning exceptions to general rules. However, redundancy will come at the expense of diversity. Precisely because in feature elicitation it is important to hold as many other factors as possible constant (i.e. elicit gender differences by comparing two sentences that differ *only* in gender), a very different elicitation corpus is needed when eliciting structures. For this reason, the most important design criterion for the corpus described in this paper is that it should be diverse by including a wide variety of structural phenomena. Further, it should be sufficiently small for a bilingual user to translate it within a matter of hours.

As was noted before [8], one inherent challenge of automatic language elicitation is a bias towards the source language (SL). The user is simply presented with a set of sentences to translate. Phenomena that might occur only in the TL are not easily captured without explanations, pictures, or the like. In parallel work, we are addressing the issue; meanwhile, we must be aware of this elicitation bias. We handle it in our rule learning module by only conservatively proposing TL structures. Structures that are not mirrored in the SL are not learned explicitly. For more details on the rule learning, refer to [7].

2 Creating a Structurally Diverse Elicitation Corpus

To create a structurally diverse minimal elicitation corpus, we begin with 122176 sentences from the Brown section of the Penn Treebank [5].Each of the sentences is annotated with a full parse. We map the tagset used in the Penn Treebank to a tagset that is more suitable to our task, e.g. different verb forms (VBD for past and VB for non-past) are collapsed. Our algorithm then traverses the parses from the top down, splitting each of them into multiple subparses and thereby creating a substantially larger training corpus. For each interior node, i.e. a node that does not pertain to only one specific word, the enhancement algorithm extracts the subtree rooted at this node. This allows us to obtain examples of NPs of different make-up and context, e.g. with adjectives, series of modifying PPs, etc. It also allows us to model the distribution of types of structures other than full sentences. In our experiment, the enhanced Penn Treebank training corpus contains 980120 sentences and phrases. Elicited examples of different types are used by the rule learner to transfer rules for different structures, so that the rules can compositionally combine to cover larger input chunks.

The next step is to represent each parse by a meaningful identifier. This is done in order to determine how many different structures are present in the corpus, and how often each of these structures occurs. Consider the following two examples sentences:

```
The jury talked.
(<S>
  (<NP> (DET the-1) (N jury-2))
  (<VP> (V talked-3)))

Robert Snodgrass , state GOP chairman , said a meeting held Tuesday night
in Blue Ridge brought enthusiastic responses from the audience.
(<S>
  (<NP> [Robert Snodgrass ... chairman , ])
  (<VP> [said ... audience]))
```

We can see that they essentially instantiate the same high-level structure of a sentence, namely `S`→ `NP VP`. For this reason, we chose to represent each parse as an instance of its *component sequence*, which describes the parse's general *pattern.* Since a pattern is always uniquely represented by a component sequence, the two terms are essentially used interchangeably in the remainder of the paper. The component sequence of a parse is defined as a context-free rule whose left-hand side is the label of the parse's top node, and whose right-hand side is the series of node labels *one level down* from the parse's top node, `S`→ `NP VP` in the examples above.

The resulting training corpus contains a list of sentences and phrases, together with their parses and corresponding component sequences. We then create a list of all the unique patterns (component sequences) encountered in the training data and a count of how many times each such sequence occurred. The sequences are sorted by *types*, i.e. the label of the parse's top node. The elicitation corpus we want to create should contain example patterns from each type. We chose to focus on the following types: ADVP, ADJP, NP, PP, SBAR, and finally S, as we believe these to be stable constituents for transfer between many language pairs. Future work may address other types of structures, such as VP or WHNP (e.g. `'in what'`). Some pattens occur frequently, whereas others are rarely encountered. For example, NPs can exhibit different patterns, e.g. `NP`→`PRO`, or `NP`→`DET N`, both of which are very frequent patterns, but also, less frequently, `NP`→ `DET ADJP N N`. Table 1 shows the five most frequent patterns for each type, together with their frequency of occurrence in the training corpus.

In order to maximize the time effectiveness of the bilingual speaker who will translate the corpus, we wish to focus on those patterns that occur frequently. At the same time, we would like to know that we have covered most of the probability mass of the different patterns of a given type. We chose to use the following method: for each pattern, we plot a graph depicting the cumulative probability with the addition of each pattern. An example of such a graph can be seen in Figure 1 below. The y-axis in this graph is the cumulative probability covered (i.e. what portion of the occurences of this type in the training corpus are covered), and the x-axis is the cumulative number of patterns. In other words,

AdvP:		**AdjP:**	
Frequency	Component Sequence	Frequency	Component Sequence
27930	ADVP→ ADV	14046	ADJP→ ADJ
1631	ADVP→ ADV ADV	2650	ADJP→ ADV ADJ
1468	ADVP→ PREP	2186	ADJP→ ADJ PP
910	ADVP→ ADV PP	1057	ADJP→ ADJ CONJ ADJ
448	ADVP→ ADV ADV PP	848	ADJP→ V
NP:		**PP:**	
Frequency	Component Sequence	Frequency	Component Sequence
48337	NP→ N	106864	PP→ PREP NP
45424	NP→ PRO	2279	PP→ PREP S
40560	NP→ DET N	1407	PP→ PREP PP
15412	NP→ DET N PP	876	PP→ PREP ADJP
11797	NP→ DET ADJ N	838	PP→ ADVP PREP NP
SBAR:		**S:**	
Frequency	Component Sequence	Frequency	Component Sequence
6993	SBAR→ SUBORD S	26813	S→ NP VP
2649	SBAR→ WHADVP S	11622	S→ NP AUX VP
771	SBAR→ WHPP S	1535	S→ NP AUX NEG VP
239	SBARQ→ WHADVP SQ	1361	S→ PP PUNCT S
205	SBAR→ DET S	1246	S→ NP AUX ADVP VP

Table 1. Most frequent patterns for different types

the highest-ranking NP pattern accounts for about 17.5% of all occurrences of NPs in the training data; the highest and second highest ranking patterns together account for about 30% of NPs. We then linearly interpolate the data points in the graph, and choose as a cutoff the relative addition of probability mass by a pattern: We compute for each pattern the amount of probability that is added when adding a pattern. This can be computed by $\delta_{p_i} = \frac{c_{p_i}}{\sum_{j=1}^{n} c_{p_j}}$ where c_{p_i} is the number of times pattern p_i occurred in the training data and n is the number of unique patterns for a specific type. We include in the corpus all patterns whose δ_{p_i} falls above a threshold. For the experiments presented here, we chose this threshold at 0.001. This allows us to capture most of the relevant structures for each type, while excluding most idiosyncratic ones. For instance, the lowest-ranking NP pattern included in the corpus still occurred more than 300 times in the original corpus.

For each of the patterns that is to be included in the elicitation corpus, we would like to find an example that is both representative and as simple as possible. For instance, consider again the two sentences presented above: `The jury talked.` and `Robert Snodgrass , state GOP chairman , said a meeting held Tuesday night in Blue Ridge brought enthusiastic responses from the audience`. Clearly, the first sentence could be a useful elicitation sentence, while the second sentence introduces much more room for error: a number of reasons (such as lexical choice, the complex SBAR structure, etc.) could prevent

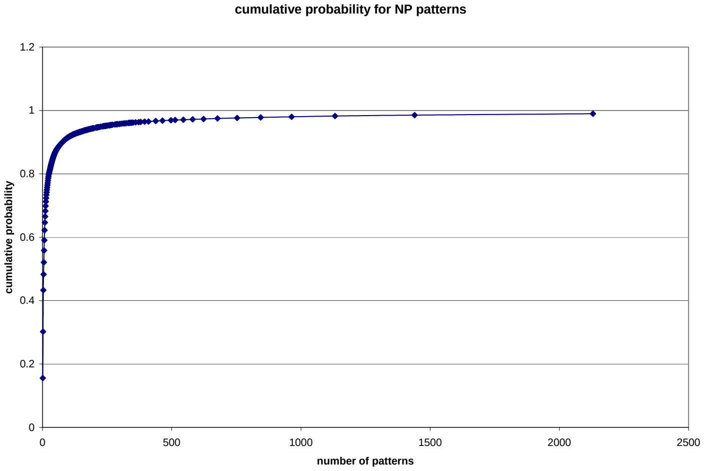

Fig. 1. An example of a cumulative probability graph

the user from translating this sentence into a similar structure. We therefore would like to create a corpus with representative yet simple examples. In order to automate this process somewhat, we extract for each pattern one of the instantiations with the fewest number of parse nodes. This heuristic can help create a full elicitation corpus automatically. It is however advisable to hand-inspect each of the automatically extracted examples, for a variety of reasons. For instance, the automatic selection process cannot pay attention to lexical selection, resulting in sentences that contain violent or otherwise inappropriate vocabulary. It can also happen that the automatically chosen example is in fact an idiomatic expression and would not easily transfer into the a TL, or that the structure, taken out of context, is ambiguous. Some patterns are also not appropriate for elicitation, as they are idiosyncratic to English, e.g. determiners can make up NPs (e.g the 'this' in `This was a nice dinner`), but this does not necessarily hold for other languages. Finally, the Penn Treebank contains some questionable parses. In all these problematic cases, we manually select a more suitable instantiation for the pattern, or eliminate the pattern altogether.

The resulting corpus contains 222 structures: 25 AdvPs, 47 AdjPs, 64 NPs, 13 PPs, 23 SBARs, and 50 Ss. Some examples of elicitation sentences and phrases can be seen in listing below. The examples are depicted here together with their parses and component sequences. The bilingual user that will translate the sentences is only optionally presented with the parses and/or component sequences. Since not all bilingual users of our system can be expected to be trained in linguistics, it may be appropriate to present them simply with the phrase or sentence to translate. Other context information, such as the parse as well as the complete sentence (if eliciting a phrase), can be provided. This can help the user to disambiguate the phrase if necessary.

```
SL: to the election
C-Structure:(<PP> (PREP to-1) (<NP> (DET the-2) (N election-3)))
CompSeq: PP-> PREP NP

SL: the chair in the corner
C-Structure:(<NP> (DET the-1) (N chair-2) (<PP> (PREP in-3)
  (<NP> (DET the-4) (N corner-5))))
CompSeq: NP-> DET N PP

SL: attorneys for the mayor
C-Structure:(<NP> (N attorneys-1) (<PP> (PREP for-2) (<NP>
  (DET the-3) (N mayor-4))))
CompSeq: NP-> N PP

SL: I can not run
C-Structure:(<S> (<NP> (PRO I-1)) (<AUX> (AUX can-2)) (<NEG>
  (ADV not-3)) (<VP> (V run-4)))
CompSeq: S-> NP AUX NEG VP
```

3 Multiple Corpora

In this section, we argue that an elicitation corpus as small as the one we describe can be useful without losing important information. This is shown by creating an increasingly redundant corpus and observing that the information gained converges as redundancy increases, as described below.

One common problem is that lexical selection in the elicitation language can lead to unexpected or non-standard translations. For example, when eliciting the pattern NP→ DET ADJ N with a TL such as Spanish or French, depending on the adjective in the example, it will occur either before or after the noun in the TL. The ultimate goal of elicitation is to learn both the general rule (i.e. adjective after the noun) as well as the exceptions; it is however more important that we not miss the more general rule. This would happen if the elicitation instance will contain an adjective that represents an exception. Redundancy in the corpus can serve as a safeguard against this issue. We have therefore created three corpora, each of which contain different examples for the *same* list of SL patterns. Whenever possible, the structure of the training example was slightly altered; the high-level structure, i.e. the component sequence, however, always remained the same. For instance, each of the corpora contains an example of the high-level structure (type and component sequence) NP→ DET N PP.

```
SL: the size of this city
C-Structure:(<NP> (DET the-1) (N size-2) (<PP> (PREP of-3) (<NP> (DET this-4) (N city-5))))
CompSeq: NP-> DET N PP

SL: a dispute with the school board
C-Structure:(<NP> (DET a-1) (N dispute-2) (<PP> (PREP with-3) (<NP> (DET the-4) (N school-5)
   (N board-6))))
CompSeq: NP-> DET N PP

SL: the chair in the corner
```

```
C-Structure:(<NP> (DET the-1) (N chair-2) (<PP> (PREP in-3) (<NP> (DET the-4) (N corner-5))))
CompSeq: NP-> DET N PP
```

For evaluation purposes, we have translated the three corpora into German and have word-aligned the parallel phrases and sentences by hand. Two of the corpora were also translated into Hebrew by an informant. Below, we evaluate our structural elicitation corpus based on the translations obtained for German and Hebrew.

3.1 Results on Component Alignments

We elicited examples with different component sequences in the hope of obtaining information about similar structures in in the TL. In order to measure how much more information we gather by adding additional corpora, we seek to determine how the component sequences are mapped into the other language. For example, we may find that the sequence `NP→ DET ADJ N` maps to the TL structure `NP→ DET N ADJ` with the alignments ((1,1),(2,3),(3,2)), i.e. the second component in the SL maps to the third component in the TL, etc. The basic premise of designing a small elicitation corpus is that these component alignments would mostly stay constant for different instantiations of the same pattern. It can always happen that the instance for a specific pattern exhibits an exception to a more general rule, in which case the general mapping rule would not be learned. Redundancy in the corpus can overcome this problem. As was said above, however, the redundancy in our corpus must be kept to a minimum in order to keep the translation task relatively small for the bilingual informant. A balance must be struck between these two competing interests.

We can measure how many different component alignments are added with the addition of each corpus. This is done by comparing the component alignment for each pattern between corpora. For instance, we check whether the elicitation resulted in the same or different component alignments for `‘the new management’`, `‘a favorable report’`, and `‘the first year’`, all instances of the pattern `NP→ DET ADJ N`.

The component alignments can be determined with high confidence from the word alignments. They represent the order in which the SL components are transferred to the TL. In the simplest case, the word alignments contain only one-to-one alignments. In this case, we can simply gather all the indices under each component into one index on the SL side. On the TL side, we use the word alignments to create sets of TL components (i.e. all indices that align to all SL indices of a specific SL component). For example:

```
;;SL: that he would not sleep ;;TL(Hebrew): $ HWA LA II$N
ConstSeq: SBAR-> SUBORD S
word alignment: ((1,1),(2,2),(4,3),(5,4))
component alignment: ((1,1)(2,2))
```

In this case, the S `‘he would not sleep’` is a component, so that SL indices 2-5 together form a component. The aligned TL indices form a set with no alignments to any other component (other than the SL S), so that it can be postulated that they form a TL component. However, things are not always this simple. For instance, it can happen that there are 0-1 or 1-0 word alignments, 1-many or many-1 word alignments, discontinuous constituents, or boundary crossings. Discontinuous constituents result in splitting the sets for one component into two. Many-1 or 1-many word alignments are handled by gathering them in one set of indices to form a component. For instance, if SL index 1 aligns to TL indices 3 and 4, then we create a TL component containing indices 3 and 4, as in the pattern below.

```
;;SL: federal aid to education ;;TL: staatliche Ausbildungshilfe
CompSeq: NP-> ADJ N PP
word alignment: ((1,1),(2,2),(4,2))
component alignment: ((1,1)(2,2)(3,2))
```

Each corpus contains 222 SL patterns. When adding a second corpus for German, we obtained an additional 52 patterns. The addition of the third corpus resulted in only an additional 25 patterns. For Hebrew, we only have translations for two of the corpora available. It was found that in the first corpus, 209 unique component sequences and alignments were elicited. Some patterns (15) were not translated; others (10) had more than one translation, while not all translations resulted in different component analyses. The second corpus added 55 new patterns with unique component alignments, and an additional 15 that were not translated in the first corpus. In the second corpus, 9 patterns were not translated. It can be seen from the results in the below table that the addition of corpora does add patterns that had not been observed before. However, each additional corpus adds less to the number of patterns observed, as expected. The main conclusion is that the number of additional patterns drops off very quickly. With the third German corpus, only 11% of the patterns in the corpus result in a component sequence that was previously unobserved. This leads us to argue that we have good evidence that in the case of German, the most common structure mappings appear to be covered already by the first two instances. The addition of a third corpus adds additional redundancy and protection from information loss. Hebrew is a more difficult case for elicitation, so that a third (and maybe fourth) corpus appears to be advisdable.

	German	Hebrew
$corpus_1$	222	209
$corpus_1 + corpus_2$	52	55+15
$corpus_1 + corpus_2 + corpus_3$	25	n/a

3.2 Results on Learned Grammars

The previous evaluation metric is important because it allows us to gain insight into how different the elicited structures are between different corpora. The ultimate purpose of elicitation for our work is however to learn structural transfer rules. In this section, we describe and discuss the rules that were learned from the three German and two Hebrew corpora.

In our rule learning system, we pay special attention to not overgeneralize the learned rules. This is achieved in part by leaving unaligned words lexicalized.

Similarly, words that are not aligned one-to-one are often left lexicalized, so as to not postulate structures in the TL that are merely caused by specific lexical choices. Thus some of the rules contain lexical items and are thus not as general as they would be if a human grammar writer had designed them. This means that we will often learn different rules for the same pattern, even if the component alignment as described above is the same. This indicates a measure of safeguarding in the training corpus. In order to determine how effective our elicitation corpus is for learning rules, we trained our system on the three German and two Hebrew corpora separately and measured how many unique rules are learned in each case. The results can be seen in the table below.

	German	Hebrew
$corpus_1$	222	209
$corpus_1 + corpus_2$	96	134
$corpus_1 + corpus_2 + corpus_3$	73	n/a

As expected, there is more overlap in the rules learned for German between the different corpora, because English and German are more closely related than English and Hebrew. In particular, it was observed that many words in the English-Hebrew rules are left lexicalized due to word alignments that were not one-to-one. This again leads us to conclude that it would be useful to obtain additional data.

Some examples of learned rules can be seen below. As was mentioned above, the rules are learned for different types, so that they can combine compositionally at run time.

```
;;SL: MOST RURAL AREAS ;;TL(German): DIE MEISTEN LAENDLICHEN GEGENDEN
NP::NP [ADJ ADJ N] -> ["DIE" ADJ ADJ N]
((X1::Y2)(X2::Y3)(X3::Y4))

;;SL: I WILL SOON READ THE BOOK ;;TL(German): ICH WERDE DAS BUCH BALD LESEN
S::S [NP AUX ADVP V NP] -> [NP AUX NP ADVP V]
((X1::Y1)(X2::Y2)(X3::Y4)(X4::Y5)(X5::Y3))

;;SL: THE CITY EXECUTIVE COMMITTEE ;;TL(Hebrew): H W&DH H MNHLT $L H &IRIH
NP::NP ["THE" N N N] -> ["H" N "H" N "$L" "H" N]
((X2::Y7)(X3::Y4)(X4::Y2))

;;SL: A SPECIAL CONSTITUTIONAL QUESTION ;;TL(Hebrew): $ALH XWQTIT MIWXDT
NP::NP ["A" ADJ ADJ N] -> [N ADJ ADJ]
((X2::Y3)(X3::Y2)(X4::Y1))
```

4 Conclusions and Future Work

We have presented an approach to designing a very small elicitation corpus that covers a large portion of the probability mass of English patterns that of interest to our work. Because of the structural diversity of the corpus, we can utilize the time of a bilingual user efficiently: translating and hand-aligninining a corpus of 222 sentences or phrases is a task of one or two hours, as reported by a bilingual speaker who translated the corpus into Hebrew. One or more additional corpora of the same size add additional TL (i.e. elicited) patterns and introduce important redundancy into the corpus, so that is unlikely that only exceptions, not general rules are learned.

We observed that for closely related languages, such as English and German, a smaller corpus is sufficient for learning rules, while for not closely related languages, such as English and Hebrew, more data collection is appropriate. We observed however that even for Hebrew there is significant overlap between the rules and TL structures observed from the first and the second corpus. Additional examples for each pattern are expected to yield even fewer new rules and structures.

Future work in this area can be divided into two different areas. First, we have already begun investigating methods to identify elicitation examples that are not appropriate for a given TL. Some structures are heavily biased towards English, and might only be interesting for closely related languages such as German. Others can result in overgeneralized transfer rules. Second, we plan to expand the corpus. With a bilingual informant available for more than a few hours, we can elicit additional information. In such a case, we can begin to focus on eliciting important exceptions to general rules, and to infer the cases in which these exceptions occur. The ultimate goal is for this structural corpus to be integrated with our elicitation corpus that focuses on linguistic features. An online navigation algorithm will help the system elicit only pertinent sentences or phrases, and eliminate phenomena that are not relevant to a given TL.

References

1. Bouquiaux, Luc and J.M.C. Thomas. Studying and Describing Unwritten Languages, The Summer Institute of Linguistics, Dallas, TX, 1992.
2. Comrie, Bernard and N. Smith. Lingua Descriptive Series: Questionnaire, Lingua, 42, 1-72, 1977.
3. Lavie, Alon, S. Vogel, L. Levin, E. Peterson, K. Probst, A. Font Llitjos, R. Reynolds, J. Carbonell, R. Cohen. Experiments with a Hindi-to-English Transfer-based MT System under a Miserly Data Scenario, ACM Transactions on Asian Language Information Processing (TALIP), 2:2, 2003.
4. Jones, Douglas and R. Havrilla. Twisted Pair Grammar: Support for Rapid Development of Machine Translation for Low Density Languages, Third Conference of the Association for Machine Translation in the Americas (AMTA-98), 1998.
5. Marcus, Mitchell, A. Taylor, R. MacIntyre, A. Bies, C. Cooper, M. Ferguson, A. Littmann. The Penn Treebank Project, http://www.cis.upenn.edu/ treebank/home.html, 1992.
6. Probst, Katharina, R. Brown, J. Carbonell, A. Lavie, L. Levin, E. Peterson. Design and Implementation of Controlled Elicitation for Machine Translation of Low-density Languages, Workshop MT2010 at Machine Translation Summit VIII, 2001.
7. Probst, Katharina, L. Levin, E. Peterson, A. Lavie, J. Carbonell. MT for Resource-Poor Languages Using Elicitation-Based Learning of Syntactic Transfer Rules, Machine Translation, Special Issue on Embedded MT, 2003.
8. Probst, Katharina and L. Levin. Challenges in Automated Elicitation of a Controlled Bilingual Corpus, 9th International Conference on Theoretical and Methodological Issues in Machine Translation (TMI-02), 2002.
9. Sherematyeva, Svetlana and S. Nirenburg. Towards a Unversal Tool for NLP Resource Acquisition, Second International Conference on Language Resources and Evaluation (LREC-00), 2000.