

# Contents Modelling of Neo-Sumerian Ur III Economic Text Corpus

**Wojciech Jaworski**  
Institute of Informatics  
Warsaw University  
Banacha 2, 02-097 Warsaw, Poland  
wjaworski@mimuw.edu.pl

## Abstract

This paper describes a system for processing economic documents written in the ancient Sumerian language. The system is application-oriented and takes advantage of the simplicity of ancient economy. We have developed an ontology for a selected branch of economic activities. We translate the documents into a meaning representation language by means of a semantic grammar. The meaning representation language is constructed in a way that allows us to handle massive ambiguity caused by: the specifics of the Sumerian writing system (signs' polyvalence, lack of mid-word signs), our incomplete knowledge of the Sumerian language and frequent damages of documents. The system is augmented with the capability of processing documents whose parts describe concepts not included in the ontology and grammar. As an effect we obtain a structural description of the documents contents in the meaning representation language, ready to use in historical research.

## 1 Introduction

Sumerians lived from prehistoric times until late 3rd millennium BC in lower Mesopotamia (modern Iraq). Sumer was the first highly developed urban civilisation, which used cuneiform script. During the period of the 3rd dynasty of Ur (2100 BC-2000 BC), whose power extended as far as present

Iraq and western Iran, the state introduced a centrally planned economy with an extensive bureaucratic apparatus.

Civil servants reported on clay tablets agriculture and factory production, lists of worker salaries, summaries of executed work, distribution of commodities, goods, animals etc., lists of sacrificed animals, travel diets and other economical information.

Archaeologists dug out about 100000 of tablets from that period (known also as the Neo-Sumerian period). The corpus of 44365 tablets (as of March 26, 2008) is available in electronic version in the database of Cuneiform Digital Library Initiative (CDLI, 2006), run by the University of California in Los Angeles and Max Planck Institute for the History of Science. Tablets are stored in the form of Latin transliteration, often accompanied by photographs or drawings (which are irrelevant for us).

Economic documents are an essential source of information about ancient Sumer. The corpus contains crucial information about economic, social and political history of the state, as well as its political system and administration structure. The sources of this type provide the most complete information about the daily life of those days. For more information about the economy of the Ur III kingdom see for example (Stepien, 1996; Sharlach, 2004; Steinkeller, 1987).

As more and more Neo-Sumerian economic documents were published, the problem arose of searching for documents relevant to the investigated topic, and the problem of representing the content of a large number of documents in a concise way.

As a solution to these problems we propose a knowledge base, that would assist in prosopographic (identification of Sumerian officials), his-

© 2008. Licensed under the *Creative Commons Attribution-Noncommercial-Share Alike 3.0 Unported* license (<http://creativecommons.org/licenses/by-nc-sa/3.0/>). Some rights reserved.

Figure 1: An example of a transliterated cuneiform tablet from Ur III

```

&P123831 = OIP 121, 101
tablet
obverse
1. 1(disz) sila4 ur-mes ensi2      1 lamb Urmes governor
2. 1(disz)# sila4 da-da dumu lugal  1 lamb Dada son of king
3. 1(disz)# sila4 id-da-a          1 lamb Idda
reverse
1. u4 2(u) 3(asz@t)-kam           Day 23
$ 1 line blank
3. mu-DU                          delivery
4. ab-ba-sa6-ga i3-dab5          Abbasaga received
5. iti sze-KIN-ku5               month sze-kin-ku5
6. mu en {d}inanna ba-hun       Year when high priest of goddess Innana
left                               was elevated to office
1. 3(disz)                        3

```

torical and philological research. For our studies, we selected a subcorpus of 11891 documents concerning distribution of domestic animals.

The design of our system was determined by two factors: the characteristics of the underlying corpus and the functionality needed by historians who are end-users of the system.

We developed an ontology for the selected subcorpus. It determines the structure of the knowledge base. It also defines the components of the meaning representation language, which we use for storing data in the knowledge base.

We translate the documents into the meaning representation language by means of a semantic grammar. The grammar has the strength equal to a regular language. Grammar symbols are equipped with semantic values that carry information included in the texts. Semantic values are generated during the parsing process by means of semantic attachments to grammar symbols.

The meaning representation language is constructed in a way that allows it to handle ambiguities caused by: the inherent ambiguity of the Sumerian writing system (signs' polyvalence, lack of mid-word signs), our incomplete knowledge of the Sumerian language and frequent damages of tablets.

The system is augmented with the capability of processing documents whose parts describe concepts not included in the ontology and grammar. We keep the ontology itself small (circa 30 concepts) because we include in it only concepts that are commonly found in documents. This approach is determined by the needs of the intended users: historians need a structural representation for typical information contained in the documents, which would provide insight into the structure and ac-

tivity of Sumerian administration and analyse it by means of statistical methods; rare, untypical phrases are thoroughly examined and interpreted manually.

The paper is organised as follows: In Section 2 we describe the Neo-Sumerian Ur III Economic Text Corpus. In Section 3 we present the ontology for documents. In Sections 4, 5, 6 we define accordingly: the meaning representation language, grammar, representation semantic values of grammar symbols and semantic attachments. In Section 8 we show how to extract transaction descriptions from the documents. In Section 9 we present applications of the system. Section 10 draws conclusions.

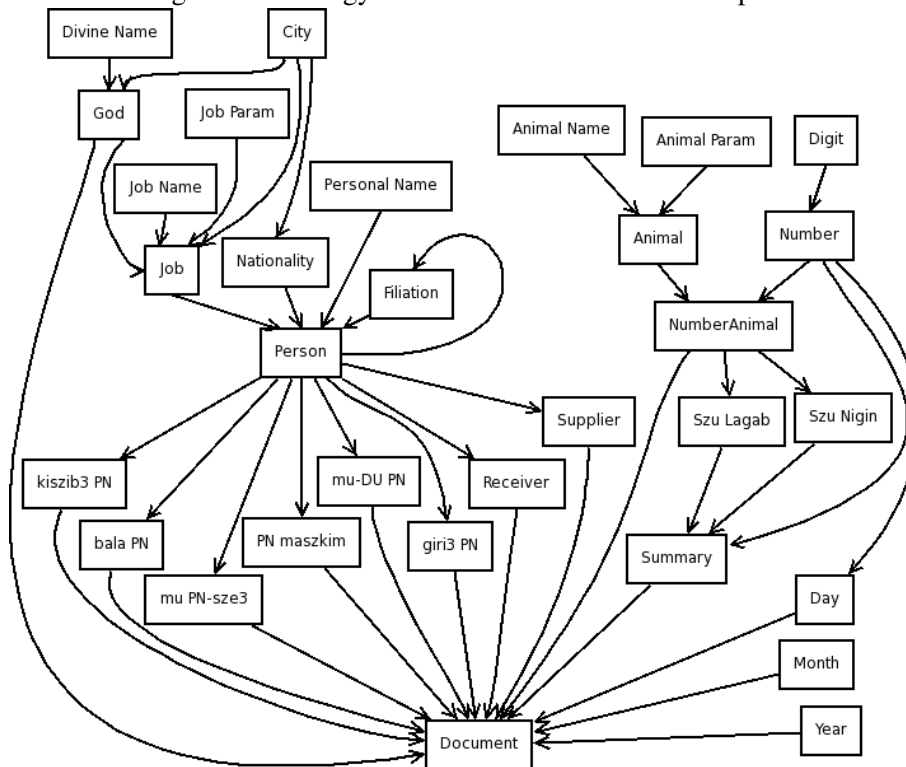
## 2 Ur III text corpus

Sumerian writing system was like a rebus, without punctuation signs, also with large amount of polyvalential signs. According to the rule of polyvalence one sign could, depending on the context, be read differently. For example the DU sign meaning — a leg, could have also other translations like: gin — to go, gub — to stand straight, tum2 — to bring, etc.

Phonetical values of signs are most often single syllables. Simple words are written by single ideograms equipped with phonetically written affixes. More complicated terms can be represented by combinations of multiple signs. Due to lack of mid-word signs and agglutinative character of the Sumerian language, it is hard to identify if we read a word or a phrase, and define its borders in the text. For more information about the Sumerian language see (Labat and Malbran-Labat, 1988).

The fact that administrative documents are written in bureaucratic language which is full of nota-

Figure 2: Ontology of Sumerian economic text corpus



tion abbreviations, lacks most of grammatical affixes and does not have uniformed spelling for non-Sumerian words, makes the situation even more complicated.

Texts are often physically damaged. Damages can concern one or many following signs. Sometimes many following lines are illegible. In the transliterations damages are marked by special symbols or descriptions in English, German, French, etc., depending on translator's nationality.

Fig. 1 presents the contents of a typical Sumerian document. This documents reports the transfers of lambs from 3 people to ab-ba-sa6-ga, an official of the Ur III state. They took place in the 23th day of the month sze-kin-ku5 in the year when the high priest of goddess Innana was elevated to office.

### 3 Ontology

As we mentioned above, we examine the subcorpus of documents concerning distribution of breeding animals. The distribution was organised in the form of transactions. During each transaction one **Person**, called **Supplier**, transfers a **Number** of animals to another Person, called **Receiver**. **Animal** description consists of information like: species, age, gender, fodder, etc. Person is de-

scribed by means of his/her **Name**, **Filiation**, **Job** and/or **Nationality**. Apart from the **Supplier** and **Receiver**, other Persons might have assisted in the transaction:

**giri3 PN** Middleman between Supplier and and Receiver

**mu-DU PN** Person on whose account the transaction took place, Receiver is probably Mu Du's representative

**mu PN-sze3** Person in whose name the Receiver or Supplier acted.

**kiszib3 PN** Person who sealed the document

**PN maszkim** Overseer of the transaction

**bala PN** Person who provides goods as royal tax.

The roles are named after the Sumerian phrases used to introduce them **PN** states for **Personal Name**), their meaning is still studied by sumerologists. One of applications of our research is to provide more facts on their meaning.

There are a few kinds of **Summaries** of various animal types in the documents, denoted by concepts: **Szu Lagab**, **Szu Nigin** as well as single **Number**. Dates of transactions are also provided.

Figure 3: The semantics of text. Observe the use of domain knowledge and the representation of ambiguous semantics for year name.

```

Number( $q_1, 1$ )  $\wedge$  Animal( $a_1, \text{sil}a_4$ )  $\wedge$  Name( $n_1, \text{ur-mes}$ )  $\wedge$  Job( $j_1, \text{ensi}2$ )  $\wedge$ 
   $\wedge$  Person( $m_1, n_1, j_1$ )  $\wedge$  Supplier( $s_1, m_1$ )  $\wedge$  NumberAnimalName( $nam_1, q_1, a_1, s_1$ )  $\wedge$ 
 $\wedge$  Number( $q_2, 1$ )  $\wedge$  Animal( $a_2, \text{sil}a_4$ )  $\wedge$  Name( $n_2, \text{da-da}$ )  $\wedge$  Job( $j_2, \text{lugal}$ )  $\wedge$ 
   $\wedge$  Person( $m_2, j_2$ )  $\wedge$  Filiation( $f_1, \text{dumu}, m_2$ )  $\wedge$  Person( $m_3, n_2, f_1$ )  $\wedge$  Supplier( $s_2, m_3$ )  $\wedge$  NumberAnimalName( $nam_2, q_2, a_2, s_2$ )  $\wedge$ 
 $\wedge$  Number( $q_3, 1$ )  $\wedge$  Animal( $a_3, \text{sil}a_4$ )  $\wedge$  Name( $n_3, \text{id-da-a}$ )  $\wedge$  Person( $m_4, n_3$ )  $\wedge$  Supplier( $s_3, m_4$ )  $\wedge$  NumberAnimalName( $nam_3, q_3, a_3, s_3$ )  $\wedge$ 
 $\wedge$  NumberAnimalNameList( $l_1, nam_1, nam_2, nam_3$ )  $\wedge$ 
 $\wedge$  Day( $d_1, 23$ )  $\wedge$ 
 $\wedge$  Delivery( $del_1$ )  $\wedge$ 
 $\wedge$  Name( $n_4, \text{ab-ba-sa}6\text{-ga}$ )  $\wedge$  Person( $m_5, n_4$ )  $\wedge$  Receiver( $r_1, m_5$ )  $\wedge$ 
 $\wedge$  Month( $month_1, 12$ )  $\wedge$ 
 $\wedge$  (Year( $y_1, \text{AS}5$ )  $\vee$  Year( $y_1, \text{IS}4$ ))  $\wedge$ 
 $\wedge$  Number( $q_4, 3$ )  $\wedge$  Summary( $s_1, q_4$ )  $\wedge$ 
 $\wedge$  Document( $t_1, l_1, d_1, del_1, r_1, m_1, y_1$ )

```

The dependencies between concepts have meronymical nature. They witness the fact that one object is a part, property, attribute or description of another one. Fig. 2 presents the diagram of dependencies between concepts.

Animals, roles and dates are transaction attributes. Yet, often one document describes many transactions and their common attribute values are written only once in the document. As a result, transactions are entangled and the individual transaction do not appear explicitly until the very last phase of processing.

#### 4 Meaning representation language

The meaning representation language represents concepts included in the data and dependencies between them. Syntax of the language is defined as follows: We have a set of constants and a set of predicate names. Constants play the role of labels for entities described in the data. Predicates consist of lists of one or more arguments. Number of arguments for a given predicate is not fixed. Predicates represent relations on finite sequences of entities. Names of most predicates are identical to the names of the related ontological concepts. The first argument of a predicate is an object which belongs to the ontological concept denoted by predicate name.

Predicates are atomic formulas. Compound formulas are composed of one or more atomic formulae connected by conjunctions and/or disjunctions. We do not use quantifiers, functions and negation.

Fig. 3 we provide our example tablet written in formal language. Number, Animal, Name, Person, Receiver, Year, Document, etc. denote corresponding ontological categories.  $q_1, q_2, a_1, m_1, del_1$ , etc. represent objects described in the text.

For example:

- Day( $d_1, 23$ ) means that  $d_1$  is a **Day** and its number is 23.
- Name( $n_1, \text{ur-mes}$ ) means that  $n_1$  is a **Name** written as  $\text{ur-mes}$ .
- Job( $j_1, \text{ensi}2$ ) means that  $j_1$  is a **Job** called  $\text{ensi}2$ .
- Name( $n_1, \text{ur-mes}$ )  $\wedge$  Job( $j_1, \text{ensi}2$ )  $\wedge$  Person( $m_1, n_1, j_1$ ) means that  $n_1$  is a **Name** written as  $\text{ur-mes}$  and  $j_1$  is a **Job** called  $\text{ensi}2$  and  $m_1$  is a **Person** described by  $n_1$  and  $j_1$ .
- NumberAnimalName( $nam_1, q_1, a_1, s_1$ ) refers to the triple of **Number**, **Animal** and **Supplier** referenced by  $q_1, a_1, s_1$  and represented by  $nam_1$ .
- Year( $y_1, \text{AS}5$ )  $\vee$  Year( $y_1, \text{IS}4$ ) means that  $y_1$  is 5th **Year** of reign of king Amar-Sin or 4th **Year** of reign of king Ibbi-Sin.

#### 5 Syntax analysis

From the linguistic point of view, an economic document is a single sentence, whose length may vary from below 50 up to more than 5000 signs. The documents from the subcorpus are in general words of a regular language. Connections between words and phrases are determined by their positions in text. Taking this into account we decided to perform the syntax analysis by means of a grammar that generates a regular language.

We use semantic categories (such as divine names, personal names, job names, year names etc.) to describe dependencies between words. Apart from describing word connections, grammar plays the role of a lexicon and is used for determining word borders.

Morphological analysis is not needed, due to the agglutinative character of Sumerian and absence of most grammatical suffixes in economic documents.

We define the grammar as follows:

$$G = (\Sigma, N, X_I, R, +, \prec)$$

where

- $\Sigma$  is a finite set of terminal symbols,
- $N$  is a finite set of non-terminal symbols.
- $X_I \in N$  is the start-symbol of  $G$ .
- $R$  is a finite set of production rules. Each production has the form  $A \rightarrow \alpha$  or  $A \rightarrow \beta+$ , where  $A$  is a non-terminal and  $\alpha$  is a sequence of terminals and non-terminals and  $\beta \in \Sigma \cup N$ .  $A \rightarrow \beta+$  is a shortcut for an infinite set of rules:  $A \rightarrow \beta, A \rightarrow \beta\beta, A \rightarrow \beta\beta\beta, \dots$ . We call such rules *accumulation rules*.
- $\prec$  is binary relation on  $\Sigma \cup N$  such that  $A \prec B$  if and only if there is a rule  $A \rightarrow \alpha$  in  $R$  such that  $B$  belongs to  $\alpha$  or there is a rule  $A \rightarrow B+$ .
- $\prec$  is an irreflexive and transitive partial order. This guarantees that  $G$  is recursion-free and generates a regular language.

We will call every subsequence parsed to a grammar symbol a *phrase*.

**Proposition 1** *Language  $L$  can be generated by a grammar of the form defined above if and only if  $L$  is regular.*

Names of the symbols in the grammar reflect the concept names. Example rules:

Head	Body
Name	::= ur mes
Person	::= Name Job
Person	::= Name
Person	::= Job
NumberAnimal	::= Number Animal
NumberAnimalList	::= Animal +

We represent all the possible derivation trees for a given text and grammar by means of a directed acyclic graph whose edges are labelled by grammar symbols. We call it a *chart*.

The text is a sequence of signs. We represent this sequence as an oriented graph which is a list with signs as edges. Vertices of this graph are numbers pointing to the position in text. Then for each

rule we find all paths in the graph with sequences of edge labels that match the body of the rule, and add to the graph new edges from the beginnings to the ends of those paths, and label them with the rule head. We will denote an edge labelled  $\alpha$  from vertex  $i$  to vertex  $j$  by  $\alpha_{i,j}$ .

## 6 Semantic analysis

In case of an ambiguous grammar (as in our case), the number of possible syntax derivation trees may be exponential in the sequence length. The concept of chart is intended to be their common, compact representation. The number of possible semantic values of the sequence is equal to the number of syntax derivation trees. That is why we cannot represent them directly. Instead, we distribute the semantic values across the chart. Our method resembles to the approach known in the literature as Minimal Recursion Semantics (Copestake et al., 1999)

The formulas of the meaning representation language are spread across the chart. Consider the edge  $\alpha_{i,j}$  of chart. This edge was created as the result of parsing a phrase. The phrase describes an entity. We represent this entity by means of a constant  $a_{\alpha,i,j}$ . We describe its properties derived from the phrase by the formula of meaning representation language. Such a formula is composed out of constants which represent entities and predicates that describe syntactic structure connecting them. We name this formula the *semantic value of the grammar symbol* and denote it as  $[[\alpha]]_{i,j}$ .

The formula  $[[\alpha]]_{i,j}$  has the following structure:

$$\bigvee_{k=1}^n p_k(a_{\alpha,i,j}, a_{\alpha_1^k, i_1^k, j_1^k}, \dots, \dots, a_{\alpha_{m_k}^k, i_{m_k}^k, j_{m_k}^k}) \wedge \bigwedge_{l=1}^{m_k} [[\alpha_l^k]]_{i_l^k, j_l^k},$$

where  $\alpha_l^k$  are labels corresponding to the symbols in the body of the rule used to create the edge  $\alpha_{i,j}$ . Each  $[[\alpha_l^k]]_{i_l^k, j_l^k}$  is assigned to the edge  $\alpha_l^k$ , so only the set of atomic formulae

$$\{p_1(a_{\alpha,i,j}, a_{\alpha_1^1, i_1^1, j_1^1}, \dots, a_{\alpha_{m_1}^1, i_{m_1}^1, j_{m_1}^1}), \dots, p_n(a_{\alpha,i,j}, a_{\alpha_1^n, i_1^n, j_1^n}, \dots, a_{\alpha_{m_n}^n, i_{m_n}^n, j_{m_n}^n})\}$$

must be associated with graph edge on the implementation level. Semantics for a terminal symbol  $\alpha_{i,j}$  is a one-argument predicate whose name is  $\alpha$  and whose argument is variable  $a_{\alpha,i,j}$ .

Figure 4: Chart

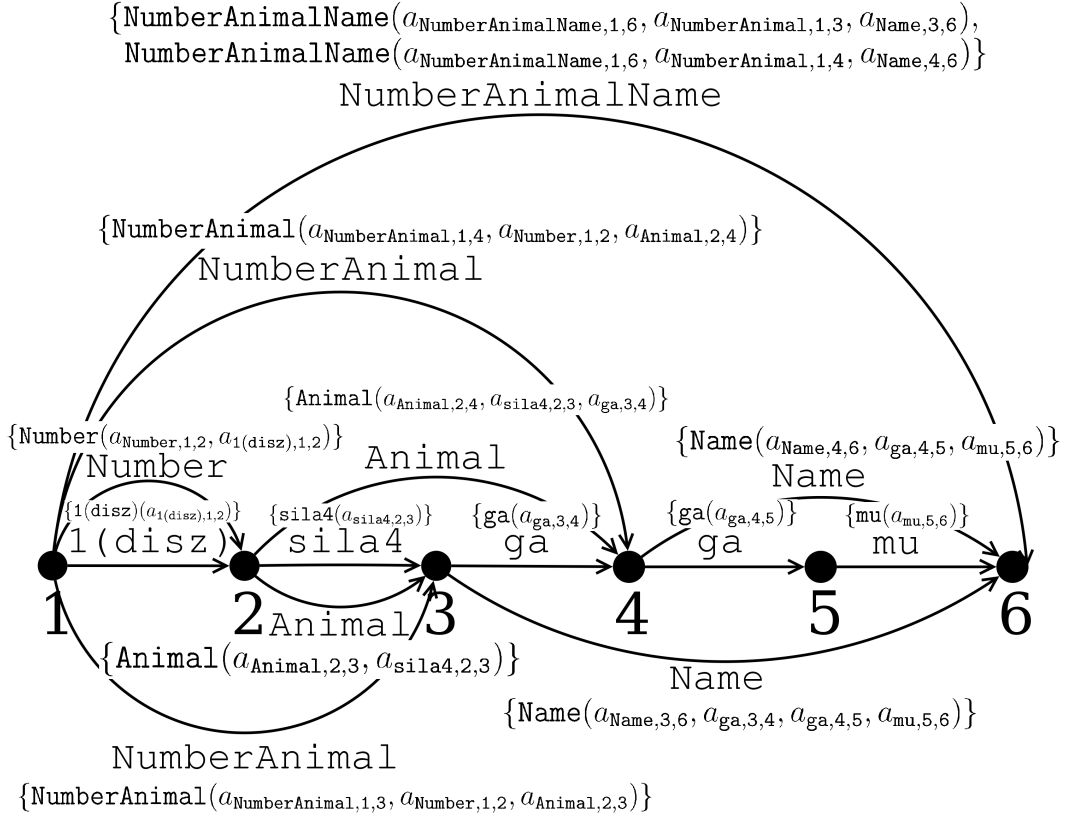


Figure 5: Semantic values of grammar symbols

$$\begin{aligned}
 \llbracket \text{Name} \rrbracket_{4,6} &= \text{Name}(a_{\text{Name},4,6}, a_{\text{ga},4,5}, a_{\text{mu},5,6}) \wedge \llbracket \text{ga} \rrbracket_{4,5} \wedge \llbracket \text{mu} \rrbracket_{5,6} = \\
 &= \text{Name}(a_{\text{Name},4,6}, a_{\text{ga},4,5}, a_{\text{mu},5,6}) \wedge \text{ga}(a_{\text{ga},4,5}) \wedge \text{mu}(a_{\text{mu},5,6}) \\
 \llbracket \text{NumberAnimalName} \rrbracket_{1,6} &= \\
 &(\text{NumberAnimalName}(a_{\text{NumberAnimalName},1,6}, a_{\text{NumberAnimal},1,3}, a_{\text{Name},3,6}) \wedge \llbracket \text{NumberAnimal} \rrbracket_{1,3} \wedge \llbracket \text{Name} \rrbracket_{3,6}) \vee \\
 &\vee (\text{NumberAnimalName}(a_{\text{NumberAnimalName},1,6}, a_{\text{NumberAnimal},1,4}, a_{\text{Name},4,6}) \wedge \llbracket \text{NumberAnimal} \rrbracket_{1,4} \wedge \llbracket \text{Name} \rrbracket_{4,6})
 \end{aligned}$$

On Fig. 4 we present an example a chart. The names of predicates associated with the grammar symbols are usually identical to their names. The first argument of each predicate is a constant that represents an entity described by the phrase. Indices of this constant identify the edge into which this phrase was parsed.

On Fig. 5 we show semantic values for some of the edges for chart on Fig. 4. For example the semantic value of the edge  $\text{Name}_{4,6}$  (denoted as  $\llbracket \text{Name} \rrbracket_{4,6}$ ) is calculated to be the formula  $\text{Name}(a_{\text{Name},4,6}, a_{\text{ga},4,5}, a_{\text{mu},5,6})$ , because the constant  $a_{\text{ga},4,5}$  refers to the edge  $\text{ga}_{4,5}$  and  $a_{\text{mu},5,6}$  to  $\text{mu}_{5,6}$ .

Ambiguous phrases may be parsed to a num-

ber of predicates. Each predicate generated for a given subsequence is a possible description of an entity. That is why we point that entity by the same constant in each predicate (consider for example semantic value of an edge  $\llbracket \text{NumberAnimalName} \rrbracket_{1,6}$ ).

We connect different possible subsequence interpretations using disjunction. Domain knowledge provides constraints on relations. These constraints may, for example, make the disjunction mutually exclusive.

We represent the semantic value of a symbol generated by an accumulation rule as a graph, whose vertices are constants that are arguments of the predicate. Each path from the first to the last

vertex in such a graph represents a list of predicate arguments. The predicates are connected by disjunction.

Semantic values of grammar symbols are constructed using semantic attachments of grammar rules. Semantic attachments are functions that compose semantics of greater objects out of the semantics of smaller ones. Predicate sets associated with edges of chart are arguments and values for these functions.

Upon the end of the parsing process we obtain an edge labelled by the start symbol of the grammar. Its semantic value is a formula which contains every possible translation of the entire text into the meaning representation language.

There are three types of rules that occur during the corpus processing:

- Rules that make up the lexicon
 

```
Name ::= ur mes
      Name(varName, varur, varmes)
```
- Rules that describe the document structure
 

```
Person ::= Name Job
      Person(varPerson, varName, varJob)
      Person ::= Name
      Person(varPerson, varName)
      Person ::= Job
      Person(varPerson, varJob)
```
- Rules that introduce domain knowledge (numbers, year names etc)
 

```
YearGod ::= d inanna
      {AS04(varYearGod), AS05(varYearGod)}
      Year ::= mu en YearGod Verb
      {p(varYear) : p ∈ valYearGod ∩ valVerb}
```

*var*<sub>Name</sub> is a variable, whose index is an edge labelled by the symbol Name. *val*<sub>YearGod</sub> is a set of predicate names connected with the symbol YearGod.

## 7 Incomplete ontology and damaged documents

In every corpus, there is a number of phrases that cannot be expressed using concepts typical for that corpus. These phrases are rare, irregular and devoid of characteristic contexts.

In order to obtain completely parsed documents we must develop a semantic representation for documents we partially do not understand. We close the ontology with the **Other Information** concept, which is treated as a transaction attribute and refers to information not included in the rest of concepts.

Parser recognises phrases it does not understand using the following heuristics: Boundaries

of transaction attributes are correlated with verse boundaries. If the parser does not recognise the content of a verse it decides that this verse contains **Other Information**.

Apart from being not understood the texts are often damaged. There are several types of damages in documents.

When a single sign is illegible, it is denoted in the document as *x*. The parser considers the *x* symbol as a wildcard that may be matched with any terminal symbol.

When a part of a verse is broken, it is denoted in the document by [ . . . ]. We estimate the number of signs in the broken verse counting the number of signs in the other verses in the document. Then we replace the [ . . . ] symbol by a sequence of *x* symbols.

We do not estimate precise contents of broken phrases, we only determine their role in the document and ontological category. This allows us to avoid combinatorial explosion while parsing damaged documents.

When a whole verse or a number of verses is broken we assume that they have syntactic structure of a typical verse or contain **Other Information**.

## 8 Transaction Extraction

After performing the semantic analysis we know the concepts used in the document, its topic and structure. Now, we extract transactions from the documents.

Transactions extracted from our example document (Fig. 1) are presented on Fig. 6. Extracted transactions are represented as edges of chart and they compose a formula of meaning representation language. For unequivocal tablets this formula is a simple conjunction of transaction predicates, whilst ambiguous documents provide transactions which are connected by conjunction and disjunction as well.

While extracting transactions, we analyse document summaries: we correlate the number of animals with their quantity written down in the document summary. This allows us to reduce the number of ambiguities generated by document damages and specifics of writing system.

We extracted 68619 transactions from the selected subcorpus with precision 86% and recall 90%.

Figure 6: Transactions described in the text (already with resolved **Year** ambiguities).

Date	Number	Animal	Supplier	Receiver
23-11-AS05	1	silā4	ur-mes ensi2	ab-ba-sa6-ga
23-11-AS05	1	silā4	da-da dumu lugal	ab-ba-sa6-ga
23-11-AS05	1	silā4	id-da-a	ab-ba-sa6-ga

## 9 Applications

Our methodology proved its usefulness in several applications.

The first, simple, but powerful and needed application is document retrieval. The system allows us to retrieve documents that satisfy queries concerning concepts and structure of the texts.

It is worth mentioning that due to the script polyvalence, text damages and editor records mixed with Sumerian text, matching a sequence of signs in a document is not a trivial task.

Using the transaction based knowledge representation we found pairs of documents that contain information about the same transaction. This task is important due to the fact that this is the way to follow the distribution process and administration activity.

The system provides help also in the tasks of: identification of Sumerian officials (prosopography); determining relations between Sumerian officials in terms of number of commodities that are transferred between them; classification of types of documents generated by Sumerian administration; research on Sumerian language itself.

## 10 Conclusions

We presented an ontology driven methodology for processing Sumerian economic documents. We applied it with a success to sumerological research problems.

We plan to extend the ontology, grammar and lexicon so that the system could process the whole corpus of Neo-Sumerian economic documents. It will allow us to construction of the model of Sumerian economics. We will estimate the contents of unavailable documents by means of statistic based approximation and machine learning methods.

Language characteristics have little influence on the system construction, so nothing prevents the system from being applied to narrow subject matter corpora written in other languages. Especially the problems of incomplete ontologies and partial data (caused for example by damages done to doc-

uments) are independent from the language characteristics.

## Acknowledgements

I would like to thank Jerzy Tyszkiewicz for the inspiration in my work. I would also like to thank Marek Stępień for the support with his knowledge of Sumerian language.

The research has been supported by the grant N-N206-400234 from Ministry of Science and Higher Education of the Republic of Poland.

## References

- Copetake, A., D. Flickinger and A. Sag. 1999. *Minimal recursion semantics: An introduction*. CSLI, Stanford University.
- Cuneiform Digital Library Initiative, The. 2006. <http://cdli.ucla.edu>.
- Jaworski, W. 2006. Learning Compound Decision Functions for Sequential Data in Dialog with Experts. In: S. Greco et al. (Eds.) *Proc. RSCTC 2006*, LNAI 4259, Springer, pp. 627-636.
- Jurafsky, D., J. H. Martin. 2000. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. Prentice-Hall.
- Labat, R., F. Malbran-Labat. 1988. *Manuel d'épigraphie akkadienne Signes, Syllabaire, ideogrammes*, 6th edition. Geuthner (Librairie Orientaliste Paul Geuthner S.A.).
- Sharlach, T.M. 2004. *Provincial Taxation and the Ur III State*. Leiden-Boston.
- Steinkeller, P. 1987. The Administrative and Economic Organization of the Ur III State: The Core and the Periphery. In: R.D. Biggs, Mc.G. Gibson (Eds.) *The Organization of Power: Aspect of Bureaucracy in the Ancient Near East*. SAOC 46, Chicago, s.19-41
- Stępień, M. 1996. *Animal Husbandry in the Ancient Near East: A Prosopographic Study of Third-Millennium Umma*. Bethesda, Md. : CDL Press.