The linguistic models of industrial and insurance companies

Vladislav B. Kovchegov
Horizon Blue Cross and Blue Shield of New Jersey
vladislav_kovchegov@horizon-bcbsnj.com
vlad_kovchegov@yahoo.com

Abstract

In this paper we discuss methods of using the language of actions, formal languages, and grammars for qualitative conceptual linguistic modeling of companies as technological and human institutions. The main problem following the discussion is the problem to find and describe a language structure for external and internal flow of information of companies. We anticipate that the language structure of external and internal base flows determine the structure of companies. In the structure modeling of an abstract industrial company an internal base flow of information is constructed as certain flow of words composed on the theoretical "parts-processes-actions" language. "The language of procedures" is found for an external base flow of information for an insurance company. The formal stochastic grammar for the language of procedures is found by statistical methods and is used in understanding the tendencies of the health care industry.

Key Word: social systems, organization, automata, formal languages, formal grammars, theory of actions, semantic space

Introduction

In this article all companies are broken into two parts: industrial and purely informational companies. The industrial company (industrial part of a company's activities) is modeled as an input flow of words where any word represents assembling or disassembling actions. Any word enables us to generate the company's structure, figure out the number of employees, the movement's path, and so on. The creation of a semantic space for industries is beyond our abilities and depends on a particular industry. This article shows us only a hint of how to do this. We show that the more deeply developed languages of parts, processes, actions plus description of shapes and conditions, and technological semantic space give us the ability to build more realistic models of industrial companies by using the described method.

A pure informational company is represented as a model of input flow of an insurance company. In real life all companies have industrial and informational parts. Companies in which the industrial part is the main section are called "industrial" companies. Companies where informational business is the main topic are called "informational" companies. But for both types of companies, and in every day life, conversations between people are important functions.

For linguistic modeling of companies we use the formal grammars. The formal grammars G is given by the quadruplet <N, T, P, S> where S is the root or start symbol, T is the set of terminals symbols, N is the set of non-terminals symbols, and P is the set of grammar rules, substitutions, or productions. The alphabet V is the union of T and N, where in the general case the alphabet is a set of arbitrary symbols. The root symbol belongs to the alphabet V. Let us use the symbol V* for all words generated by the alphabet V. For instance, if V={a, b}, then V*={Em, a, b, aa, ab, bb, ba, aab, … } where the empty word set is denoted by Em. The non-terminal symbols correspond to the variables, and the terminal symbols correspond to the words of natural language. The set of grammar rules is the set of expressions **a -> b** where **a** and **b** are words from V* and the word, **a**, is not empty. The formal language generated by the grammar G is denoted by the symbol L(G).

Part 1. THE BASE MODEL OF AN INDUSTRIAL COMPANY.

For simplicity we divide all companies based on the industrial and informational type. In real life all companies have industrial and informational parts. The industrial part consists of all activities under material flow, while the informational part has business with information. However, informational parts also include material flow such as: papers, envelopes, phones, computers (PC and mainframe), communication's networks and so on. We can then divide all activities on the internal and external parts. The internal parts focus on internal technological flows in the company and the informational system has to reflect the current condition of industrial flows. This information will be used for the operational decision making process. The strategic and tactical decision is to make processes need more external information and create a model of the external word. In this article we concentrate only on the internal part of a company's activities.

For the modeling of internal industrial flows we separate two types of actions: assemble and disassemble. We can find traces of this type of action almost everywhere. For the formulization of this type of action we can use a formal language as well. If a person or groups of people want to assemble something, he/she or they need elements that will be use as parts. Some of these parts are assembled somewhere and are just labeled as parts. While other parts are done in a company and may be the result of some activities. For these parts we will use the alphabet of actions using special symbols for denoting tools and natural processes – devices. Any device can represent a main natural process, so when we say devise we mean to keep in mind some natural process and vice versa.

So, let us denote E, the set of elements: E = {x1, … , xN}, where N is the number of parts. For instance, for a car the number of parts N equals approximately fifty thousand. We then describe the set of methods for gluing the elements and the set of human actions that do this. Let us use the symbol P for the set of gluing processes and/or devices and A for the set of actions of gluing. So, if the process is mechanical, we need parts and facilities (wrenches, bolts and so on). If the process is chemical we then have to describe the type of glues, conditions and devices. The whole assemble-action (A-action) appears as the word ((x1, x2: a1, a2, P1), x2, x4, (x5, x4: a3, a5, p2, p5); a4, a3, a7, p3), where the elements of E: the action from the set of action A, p1, … , p5 belong to the set of processes P (chemical, physical, biological and so on). The parenthesis denotes we get a new element. For instance, the word (x1, x2: a1, a2, p1) represents the new element assembled from x1, x2; and for gluing uses process p1 from P and human actions a1 and a2. But all elements must satisfy the same conditions. We can describe the set of conditions and denote this set by the symbol C. So, we say that elements xk belongs to C(xk), if xk satisfies the set of conditions C(xk). Similarly, (x1,x2:a1,a2, p1) must belong to C(x1,x2:a1,a2, p1) and so on.

Disassemble actions (D-actions) transform one element into the set of elements or fractions. For instance, D(x) ={Fx1, … , Fxn}, where x is the initial object and Fx1, … , Fxn are fractions of x. Symbol F is the first letter of the word "fraction." Occasionally it is better to use the distribution of debris sizes. Sometimes, the result of D-actions is the set of parts or parts and fractions. D-actions can be realized by using different natural and artificial processes: from explosive materials to mechanical processes. So, D-

actions can create fragments, debris and parts. Results of D-actions can be used for assembling processes. If object x consists of a biological nature, the result of D–action is a type of injury.

Thus, every element ("letter") has a shape and/or weight, material and as a description, these things (shape and so on) need to use a special language as well. For example: a description of the shape is a cylinder with diameter 2.456 feet, height 3.2

feet, depth of wall 1.2 inches; weight 1.03 ton; steel. If the element is a bit of information, we use another description: 12 millions records, length of record is 238 symbols, MS Word (extension .doc). But for our models, this information must be extended by the place of information on the semantic space (see below). If the element consists of information we need to explain what the information is about. For the modeling of a company we build the semantic space (tree) and all processes and actions must have the informational presentation on the **semantic space**. The creation of a semantic space for particular case is a very big problem. The high level of a semantic tree for a large number of tools can be expressed by the scheme "engine – transmitter – working tool - control." The typical engine transforms chemical energy into mechanical, the transmitter propagates mechanical action to the working tool, and the control system helps to control the working process. This scheme, however, is not unique and these set of schemes generate a **technological semantic space**. We will use a semantic space as a system of coordinate for all human products.

Thus following, for any process (mechanical, physical, chemical, biological, social and so on) we have to describe the condition and distance from the "normal" condition. We describe the normal condition as conditions normal for humans: physical, chemical, biological conditions (consistency of air, gravity, radiation, temperature and so on), landscape and green/animal words as well as demographic, social, and other conditions. Some conditions are considered good for processes but wrong for humans. In order to use these processes people create a **shell**, known as the **shell philosophy**. For deeper modeling of all processes we need to use the shell language. Examples of popular shells are homes, and clothing. These types of shells protect people from bad weather and decorate them. Examples of industrial shells are chemical and nuclear reactors. An ordinary vehicle consists of a combination of a few shells: the cabin of car protects the driver and the passengers. The engine creates the condition for burning fuel and transforms the chemical energy into mechanical. While a bathyscaphe protects people from high pressure and so on. For now we can rewrite the semantic scheme for tools that move and carry (cars, airplanes, ships and so on) by the description "engine – transmitter – working tool – control - shells." Chemical processes (including some metallurgical processes) use high-level semantic schemes such as "load to pot – chemical 'cooking' – unload pot," where "pot" is the shell for reactions. How can we describe a shell? We can think of a shell characterized by **condition**, **shape** and **materials**. Simultaneously, we must find a place of the shell on the semantic space.

Description of base flow of industrial company. The assembly word gives us information about the structure point of assembly, facilities and communications. We can define the physical body of a company. Parentheses represent not only objects, but points of assembly as well. There are many types of assembly locations varying from primitive to very sophisticate devices. Following this, the systems of parentheses give us information about order of actions. If a parenthesis is within another parenthesis, this object has to be done first before the second. If we have information about the assembling time for all objects, we can figure out the structure of jobs and can estimate the number simultaneously working spots.

Suppose, we have A-word ((x1,x2;a1,a2, P1), x2, x4, (x5,x4; a3,a5,p2,p5); a4, a3, a7, p3) and suppose the assembling time for object one (x1,x2;a1,a2, P1) is t1, for object two (x5,x4; a3,a5,p2,p5) the assembling time is t2, and for the whole object the assembling time is t3, where t2 approximately equals 2t3, t1=3t3. In this case for a continuous job we need 3 assembly locations for object one and two assembly locations for object two. If we know the number of workers working on object one we can estimate the necessary number of employees needed to complete the task. We then have to calculate the number of managers needed and we can finish the evaluation of necessary employees for all A-, D-operations.

All of the information reflected by the current situation of A and D actions need to be gathered by managers as well. The informational part of the model of a company will be discussed later when we describe the almost pure informational company – insurance company.

If we change proportions between the assembling times (t1=x t3, t2=y t3, where x and y not integers) we can get a more complex situation and find the necessary operative control. In the general case x and y are random numbers and these objects cannot satisfy the necessary condition. As a result, our managers are given the jobs of operative control. For a formal modeling of the operational control we can use Ianov's schemes. In the general case it is not a small job to prepare the scheme. We need to describe all possible conditions and manager reactions. It is possible, however, for either an A – or D – action to do this and then combine these schemes into one big scheme and optimize it.

The shape, weight, and materials of objects in words determine which objects stay still and which objects move. The cumbersome and/or fragile and/or heavy objects are more likely stay still than be relocated for the next step of the assembly process. The maximum sizes of the objects are defined by the size of the building (the main shell of a company), and the connections between the objects define the structure of the rooms and corridors.

For us the model of an industrial company or an industrial base of a company is the flow of A- and D – words in parts – processes –actions alphabets saturated by information about the shape and conditions. From this flow we can acquire a lot of information about the company's structure, the necessary connections between the points of assembly/disassembly and the number employees. This flow then gives us information about architectural features of the building (the main "shell" of a company).

We can then proceed onto the next step in the modeling of company. We will describe the semantic space as a base for pure human actions as conversations, conflicts, and so on. For this purpose we will describe the model of restaurant.

Part 2. THE BASE MODEL OF INSURANCE COMPANY. We then do an example of modeling certain features of a pure informational company: an insurance company. One of main purposes of this modeling is to be able to find the semantic (linguistic) base of an insurance company. The model of an insurance company is done as a mathematical modeling and a computer simulation of the input flow. The keystone of this is a semantic model of the history of clients' diseases. To construct this semantic model we use the alphabet of the types of procedures performed on patients with the given chronic disease. For instance, the alphabet for diabetes contains 34 "letters." The history of disease can be represented by a short word in a given alphabet and looks like "A_ANDR S4 S2", where A_, AN, ... , S4, S2 are letters of the alphabet of the disease. The study of the information for five years shows us that the structure of short words has a tendency to change. To model this tendency we use Markov chains. The conditional probability is found from the data. Then, using a computer simulation, we calculate a set of pseudo-random "short words." The next problem shown is the generation of the set of pseudo-random "long words." The long word may look like "A_-12AN-1DR-17 S4-1S2-1", where the number that follows each "letter" is the frequency of encountering this letter. For this purpose we find the conditional probability Pr{X=" long word"/X=" short word"} and then generate a set of pseudo-random "long words." The list of "long words" and the list of "normative prices" for procedures give us the ability to calculate the mean, "harmonic", minimal and maximal prices for all diseases. This model may be used when making the forecast of an insurance company. This model is the basis for a more comprehensive model of an insurance company.

A semantic representation of external flow and statistical simulation of the cost of disease. An insurance company is an example of a company in which the main business is almost purely informational. The real process (emerging diseases, interaction of patients and doctors, diagnosis, procedures, payments, etc.) is omitted from insurance company (IC). These actions, however, or the majority of them need to be reflected by a system of informational flows. An insurance company creates a few doctor networks, hospital networks, and forms the membership by selling insurance packages (“products”). The product consists of a set of rules for the patients purchasing the package. The cheaper package prescribes the member doctors, hospitals, and so on, while the more expensive package provides patients more freedom with additional choices. The informational and reality images are not the same, and is the main reason for the search of fraudulence, auditing, and so on, all of which are an important part of IC activities.

We now describe the type of information used. Patients that stay in hospitals are called inpatients and information on these patients’ is called institutional files. Other patients are called outpatients with an associated professional file.

The five tables are an extraction from a professional file, years 1995-99, with the disease of diabetes. We use the alphabets {A_, AN, DR, E_, LP, L_, M_, S1, S2, S3, S4, SO, RD, …}, where “letter” A_ stands for procedure “TRANSPORTATION,” “letter” AN stands for procedure “ANESTHESIA,” “letter” DR stands for procedure “DRUG,” “letter” LP stands for procedure “LAB PATOLOGY,” “letter” E_ stands for procedure “DIGEST SYSTEM,” “letter” M_ stands for procedure “MED SERVICE,” “letter” S1 stands for procedure “SURGERY:INTEG,” letter S4 stands for procedure “SURGERY: CARD,” and so on. For diabetes we use an alphabets of 37 letters. Every patient has a sequence of the procedures. We have transferred the sequence of procedures into the set of “long” and “short” words. The “long” word looks like "A_-5AN-2DR-11 S4-1 S2-2", where a letter represents a procedure, a number after letter denotes the number of times. So, A_-5 means that procedure “TRANSPORTATION” is used five times. The “short” word is the “long” word without a number. The “short” word looks like "A_ANDR S4 S2.” We then calculate the frequencies of “short” words for all five years.

We can see from the complete lists that the majority of words are small length. The bigger “words” have a less probability of occurring. The first step is to generate the list of words (randomly) and then calculate the price associated with every word. For the real calculation we generate “words with frequencies”: “AN-1; DR-17; S4-2; S5-1.” These words signify that the patient had anesthesia once, surgery (digest) – twice, surgery (cardiology) – once and took drugs 17 times. Once the list of “words with frequencies” has been generated, the program calculates the price of the list.

The linguistic model and the forecast for linguistic model. We now transfer our “short words” problem into grammar problems. This means that for a list of short words (see list for 1995–99) we generate this word’s automaton grammar. Then, when we get the five grammars for all five tables, we find the general pattern for all stochastic grammar. We thus reduce our problem into one: to make a forecast for the matrix of probability. This means we have to make a prognosis for multidimensional numerical vectors. In the general case there are a lot of solutions for this problem, but this method does not allow us to make a prognosis. We start our analysis from year 1998, which contains more information.

Step 1. Grammar for table 4.

We have grammar G4=(VN, VT, P, S), where S is the start or root symbol, P is the set of substitute or deduced rules, VT is the set of terms or set of words (having all letters belong to our alphabet), VN is the set of non terms, denoted as regular grammar. This signifies that all deduced rules look like A -> aB or A-> a, where A and B are non-terms, and a is a word in the given alphabet. We only need the description of set P. Our goal is to obtain a regular grammar, which generates a given set of short words. The set of deduced rules for words from table 4 is shown below.

S-> A_, S -> A_X1, X1 -> ANX2, X2 -> DRX3, X3 -> LP, X3 -> S4, X1 -> E_, X1 -> DR, X1 ->LP, X1 -> DRX4,

X4 -> E_X5, X6 -> S4, X5 -> LP, X4 -> E_, X4 _> LP, X4 -> S1, X4 -> S4, S -> DR, S -> DRX6, X6 -> LPX7,

X7 -> RDX8, X7 -> RD, X8 -> S4, X7 -> S4, X7 -> S1, X7 -> S1X9, X9 -> S4, X6 -> RD, X6 -> SO, X6 -> S1,

X6 -> S1X7, X6 -> S2, X6 -> S4, S -> AN, S -> ANX10, X10 -> DR, X11 -> DRX12, X12 ->S4.
We can minimize the number of substitutions and calculate the probability of using the given rule. We obtain the number rule and the probability of using this rule of substitutions. In our grammar all words are generated from the table as well as additional information. But the situation is not that bad because the arbitrary table contains only partial information and short words, which have larger frequencies. In actuality, short words can be very long. For larger words the probability of occurrence is less than the probability of a shorter one.

For future application we will represent the above grammar in a hierarchal or tree form: we divide all rules into levels. The first level is the root level and contains all substitution rules with first symbol S. The second level rules are a set of rules in which the first symbol belongs to the previous set, the first level rules, and so on.
For our case the first level rules is:

S -> A_, S -> DR, S -> AN, S -> E_, S -> LP, S -> S1, S -> S4, S -> S4, S -> RD.

The second level rules is:

A_ -> AN, A_ -> DR, A_ -> E_, A_ -> LP, A_ -> S1
DR -> E_, DR -> G_, DR -> LP, DR -> RD, DR -> RD, DR -> S1, DR -> S2, DR -> S4, AN -> DR

The third level is the set:

DR -> E_, DR -> LP, DR -> S1, DR -> S4, AN -> DR, LP -> RD, LP -> S1, LP-> S4

The forth level contains the set of rules:

E_ -> LP, E_ -> S1, E_ -> S4

and so on.

In the future we use the following notation for rules. The notation A -> B for rules of level N signifies that (a) there does not exist a rule of level N+1 that begins with symbol B; (b) the calculation process prints out a word and goes to the beginning of the calculation process. The notation A -> B* means that there exists a next level rule which starts from symbol B. We write A -> B, A -> B*, when we want say that there exists both a modification and a non-zero probability.
Frequencies for Grammar 4 (1998). 1st level (S is the starting point, followed by first level rules)

S

A_	A_*	DR	DR*	AN	AN*
.14348	0.05219	.43117	.2581	.010523	.007252

S

E_	LP	LP*	S1	S4	RD
.00384	.0506	.00597	.01735	.01948	.0000

2nd level (rules for the second level are used only for letters with asterisks: A_*, DR*, AN*, and LP*)

S -> A_

AN*	DR	DR*	E_	LP	S1
0.1110	.2597	.36769	.180376	.015857	.01189

S -> DR

E_	G_	G_S4	LP	LP*	RD	SO
.00716	.0088	.00551	.30689	.1901	.0099	.0066

S -> DR

S1	S1S4	S2	S4
.1041	.01267	.0066	.34159

S -> AN

DR	DRS4
0.78431	0.21569

Note. In this paper we do not present events with small frequencies. Symbol LP* occurs with a small empirical probability (frequency equal 0.00597), so we ignore the next level.

3rd level (use third level rules for AN*, DR*, LP*and ignore off events)

S -> A_ -> DR

E_	E_*	LP	LPS4	S1	S4
0.34837	.18797	.12030	.04762	.070175	.155388

S -> A_ -> AN

DR	DR*
.5089	.49107

S -> DR -> LP

RD	RDS4	S1	S1S4	S4
0.03478	.02898	.08985	.03188	.8145

4th level (use fourth level rules for E_*, DR* and ignore off events)

S -> A_ -> DR -> E_

LP	LPS4	S1	S4
0.25333	0.2400	0.1333	0.37333

Predictive model for base process.

The grammar can now generate the list of short words that contain more words than included real data. We must, likewise, describe the grammar frequencies for all tables and then create a general **universal grammar** that generates all tables. It is easy to demonstrate that the above-described grammar is our universal grammar. What is the difference between these grammars? The difference is just the Frequencies for Grammars and a forecast only needs to be done for this. We should be aware that our frequencies are in numerical format. We will show below the frequencies for all cases, only for level two only. All frequencies below are frequencies for a universal grammar.

2nd level (1995-99)

S -> A_

AN*	DR	DR*	E_	LP	S1
0.999	0.0001	0.0003	0.0003	0.0001	0.0002
0.5149	0.1940	0.1940	0.0970	0.00004	0.00006
0.3533	0.2827	0.1943	0.1696	0.0000	0.0000
0.1110	0.2597	0.36769	0.180376	0.015857	0.01189
0.0000	0.3898	0.4550	0.1552	0.0000	0.0000

The problem of finding a solution is standard and there are a lot of different ways. The easiest method is to interpolate all values for all columns, make negative numbers equal zero, find the sum of rows and normalize the rows by dividing by the sum. If all variables are independent, we can use the next prognosis formulas (2nd level, S -> A): x1=0;

x2=(-168.54121 + .08451*year)/(-470.3754355 + .2358393*year)
x3=(-218.045815 + .109309*year)/(-470.3754355 + .2358393*year)
x4=(-78.396752 + .0393176*year)/(-470.3754355 + .2358393*year)
x5=(-3.115175 + .0015597*year)/(-470.3754355 + .2358393*year)
x6=(-2.280141 + .001143*year)/(-470.3754355 + .2358393*year)

These formulas give us the ability of calculating the probability to use substitutions or deduction rules for the second level, case S -> A_ for an arbitrary time point.

Conclusion

In this article we constructed a stochastic grammar ("linguistic code") for the external flow of an insurance company. We believe that this method is applicable to all industries and that similar linguistic codes can be found for other major kinds of companies for further research. This would help in creating universal linguistic models of companies in all industries.

Reference.

Kovchegov, V.B., 2000, *Computer Simulation of an Insurance Company* // Proceeding of Society for Chaos Theory in Psychology and Life Sciences Conference, July 22-24, Philadelphia