Estimating Sparse Events using Probabilistic Logic: Application to Word n-Grams

Djamel Bouchaffra

bouchaff@cedar.buffalo.edu

Fax: (716) 645-6176

Phone: (716) 645-6176, Ext. 138

Center of Excellence for Document Analysis and Recognition (CEDAR)

Department of Computer Science

State University of New York at Buffalo

520 Lee Entrance

Amherst, New York 14228-2567

U.S.A.

Regular Paper

Estimating Sparse Events using Probabilistic Logic: Application to Word n-Grams

Djamel Bouchaffra

Abstract

In several tasks from different fields, we are encountering sparse events. In order to provide with probabilities for such events, researchers commonly perform a maximum likelihood (ML) estimation. However, it is well-known that the ML estimator is sensitive to extreme values. In other words, configurations with low or high frequencies are respectively underestimated or overestimated and therefore nonreliable. In order to solve this problem and to better evaluate these probability values, we propose a novel approach based on the probabilistic logic (PL) paradigm. For a sake of illustration, we focuss on this paper on events such as word trigrams (w_3, w_1, w_2) or word/pos-tag trigrams $((w_3, t_3), (w_1, t_1), (w_2, t_2))$. These latter entities are the basic objects used in speech or handwriting recognition. In order to distinguish between for example: "replace the fun" and "replace the floor" an accurate estimation of these two trigrams is needed. The ML estimation is equivalent to a frequency computation of the configuration in a training corpus. Depending on the nature of the corpus, one of the two phrases is more frequent than the other. Even more, the nature of the language is such that one of them or many other configurations may not occur in the training corpus. Our method estimates probabilities for such unseen configurations using partial informations of these configurations. For example, if the number of occurrences of the trigram "replace the fun" is very low, then the new estimation is computed from occurrences of the unigrams "replace", "the", "fun", the bigrams "the fun", "replace the" and the trigram "replace the fun".

Keywords: *Probabilistic Logic, Statistical Language Model, Invariance Property, Maximum Likelihood Estimation, Sparseness Problem, Back-off Smoothing Technique, Cross-entropy.*

1 Introduction

Data Sparseness is an inherent property of any textual database. This problem arises while collecting frequency statistics on observations from any database of finite size. Without a loss of generality, we focuss in this paper on sparse data encountered in natural language processing [1]. This latter field involves speech recognition [6, 7, 8, 12, 13, 15] and handwriting recognition [3, 4]. Statistical methods compute in the first phase relative frequencies (derived from a maximum likelihood (ML) estimation) of configurations (e.g., $word_1$ followed by $word_2$ or tag_1 followed by tag_2 (part of speech tags), etc.) from a training corpus. In the second phase, the statistical methods attempt to evaluate alternative interpretations of new textual or speech samples. The problem of data sparseness arises during this second phase, when we are dealing with configurations that have never occurred in the training corpus. Therefore, the estimation of probabilities assigned to these configurations based on observed frequencies becomes unreliable. From an estimation theory standpoint, the variances of these observation frequencies are large. Therefore the frequencies computed are inaccurate estimators of the probabilities assigned to these configurations.

To overcome this chalenging problem due to the conventional ML estimation, a number of different approaches have been proposed and applied. Contributions were essentially made in the areas of speech recognition and parts of speech tagging. In fact, Brown et al. [6] compute words cooccurrence based on their distributions. They cluster words with respect to their cooccurrence distributions. Words with similar cooccurrence distributions were assigned a same word class. The probability of any pair of words is transformed into the probability of the pair of classes they belong to. Jelinek proposed the linear interpolation by smoothing the specific probabilities with less sparse probabilities [13, 15]. Katz proposed the discounting technique using the Turing-Good formula [14, 10]. Besides, in case of estimation of lexical associations on the basis of word forms a fairly large amount of training data is required, because all inflected forms of a word are to be considered as different words. A better result using the same amount of data can be achieved by analyzing *lemmata* (inflectional differences are ignored) or even semantic classes. However, even with this refinement, the sparseness problem persists. The models used in the literature circumvent this problem by assuming that there exists a functional relationship between the probability estimate for a previously unseen cooccurrence and the probability estimates for the word contained in the cooccurrence [12], while other techniques use class-based and similarity-based models to overcome the problem [8].

We propose in this paper a novel technique based on probabilistic logic as a solution to this problem. Without a loss of generality, we focuss here on a particular kind of configurations, *(word,pos-tags) pair cooccurrence*. Our approach can make inference on phrases based on sparse informations. It appears to be more general than other existing methods for three reasons:

- We can estimate the probability of a large predicate based on small *sparse predicates*. For example, we might be interested to know whether "it is the right time to buy stocks from a particular company given informations from a newspaper". This question could be based on other sparse predicates. They are for example: `INVEST(x)`, `STOCK PRICE(x)`, `RELIABILITY(x)`, `OTHER INFO(x)` where x is a name of a company. The informations conveyed by these predi-

cates are not necessarily contiguous in the stock market newspaper.

- Unlike other methods, the probabilistic logic approach involves "logical" connectors and predicates, therefore it can exploit "semantical" relationships (or concepts) between words and phrases. For example let's provide the knowledgebase with the following information: $\forall x$: `LIVING CREATURE(x)`$\equiv \forall x$: `MORTAL(x)`, (ie., the concept of life is the same as the one of mortality). Therefore if the phrase "big living creature" is sparse then its probability estimation would involve occurrences of the phrase "the mortal creature".

- All unigrams, bigrams that are part of any sparse trigram are linearly interacting with this sparse trigram.

We present in section 2 some shortcomings of the ML estimation. We describe in section 3 the back-off method and introduce the probabilistic logic paradigm in section 4. Two different approaches are developed in section 5. The first one provides with an implicit solution and the second one provides with a closed form (or explicit solution) of the problem. We show in section 6 some experiments performed on some sparse phrases extracted from the *Wall Street Journal Corpus* (WSJ). Section 7 is the object of the conclusion and future work.

2 Shortcomings of the ML Estimate

Conventional language models are based on observations such are words (or part of speech tags) bigrams or trigrams. Many low probability observations will be missing from any finite sample. Yet, the aggregate probability of all these unseen observations remain fairly high and any new sample is very likely to contain them. Because even the largest corpora available will contain only a fraction of these possible observations, it becomes impossible to use a maximum likelihood estimator for observation probabilities. The ML estimator for the probability of a trigram (w_1, w_2, w_3) is:

$$\hat{P}_{ML} = \frac{N(w_1, w2, w_3)}{N(w_1, w_2, +)}, \tag{1}$$

where $N(w_1, w2, w_3)$ is the number of occurrences of the trigram (w_1, w_2, w_3) in the training corpus and $N(w_1, w_2, +)$ is the number of trigrams that start with (w_1, w_2). However, this ratio will assign a probability equal to 0 for any unseen trigrams, which is not desirable. In a sparse sample, the ML estimator is biased high for observed events and biased low for unobserved ones.

3 Back-Off Smoothing Scheme

To correct this bias, several statistical methods dealing with data sparseness are being proposed in the literature [8, 13, 14, 15]. We focus in this paper on the interpolation technique used in the back-off smoothing scheme.

3.1 Proportion of Unseen Words

After collecting statistics from a 1,500,000 word corpus, Jelinek et al. [13] reported that about 25% of trigrams in average are new when using a test corpus of only 300,000 words. Similarly, Rosenfeld [18] reports that on the Wall Street Journal corpus, with 38 million words of training data, the rate of new trigrams in test data (taken from the same distribution as the training data) is 21% for a 5,000 vocabulary, and 32% for a 20,000 vocabulary. Setting a zero probability value to the new trigrams is inconsistent since it gives to the corpus a very "poor" cross-entropy $\mathcal{H}$. This measure will be defined further in this section. The "back-off smoothing" technique involves terms of lower complexity (unigrams and bigrams) in the estimation of a trigram. This latter quantity is estimated as:

$$\begin{cases} P((w_i,t_i) \mid (w_{i-1},t_{i-1}) \wedge (w_{i-2},t_{i-2})) \simeq \lambda_1^h \hat{P}(w_i,t_i) + \lambda_2^h \hat{P}((w_i,t_i) \mid (w_{i-1},t_{i-1})) + \\ \qquad\qquad \lambda_3^h \hat{P}((w_i,t_i) \mid (w_{i-1},t_{i-1}) \wedge (w_{i-2},t_{i-2})) \\ \sum_{i=1}^{i=3} \lambda_i^h = 1 \\ \lambda_i^h \geq 0 \ \ \forall i = 1,..,3 \end{cases} \tag{2}$$

where $\hat{P}$ is the maximum likelihood (ML) estimator of P and λ_i^h are constant which depend on the history $h = (w_{i-1},t_{i-1}) \wedge (w_{i-2},t_{i-2})$. If we assume that most of the time we do have bigrams and that they yield a more accurate computation of the probabilities than trigams or unigrams, then λ_2^h should be higher than λ_3^h and λ_1^h. In order to determine an optimal set of λ_i (i=1,3), we need to define a model evaluator measure.

3.2 Cross-entropy as a Model Evaluator

Definition 1 *Roughly speaking, the cross-entropy (or perplexity) function enables us to measure in some scale the power of prediction of the model we have proposed. The third-order cross-entropy of a set of n random variables* $(W,T)_{1,n} = (w,t)_{1,n} =$ $((w_1,t_1),(w_2,t_2),...,(w_n,t_n))$, *(a word/tag path) where the correct model is* $P((w,t)_{1,n})$ *but the probabilities are estimated using the model* $P_M((w,t)_{1,n})$, *is given by:*

$$\mathcal{H}((W,T)_{1,n}, P_M) = -\sum_{(w,t)_{1,n}} P((w,t)_{1,n}) \times \log P_M((w,t)_{1,n}). \tag{3}$$

Because we very often assume that the information source (the language model (L, P_M) that puts out sequences of words is *ergodic*, (meaning that the statistical properties can be completely characterized in a sufficiently long sequence of words that the language puts out) therefore, we use the following cross-entropy definition:

Definition 2 *The third-order cross-entropy of a language L (captured by a textual corpus of length n) can be defined (or estimated) as:*

$$\hat{\mathcal{H}}(L, P_M) \simeq -\lim_{n\to\infty}\left[\frac{1}{n} \times \log P_M((w,t)_{1,n})\right]. \tag{4}$$

It is reasonable to assume that a pair (w_i, t_i) in a phrase or a sentence depends only on the two previous word/tag pairs (w_{i-1}, t_{i-1}) and (w_{i-2}, t_{i-2}). Using this assumption and an induction process of the Bayes' rule, we can write:

$$\begin{aligned} P_M(w,t)_{1,n} = P_M\left(\bigwedge_{i=1}^{i=n}\{(w_i,t_i)\}\right) = \\ P_M(w_1,t_1) \times P_M((w_2,t_2) \mid (w_1,t_1)) \times \prod_{i=3}^{i=n} P_M\left((w_i,t_i) \mid (w_{i-1},t_{i-1}) \wedge (w_{i-2},t_{i-2})\right). \end{aligned} \tag{5}$$

The first term of the second line of Equation 5 takes care of the probability of the first two word/tag pairs (since we do not have two previous word/tag upon which to condition the probability). For simplification purpose, we make the assumption that there are two *pseudo word/tag pairs* (w_{-1}, t_{-1}) and (w_0, t_0) (which could be some beginning-of-text marks) on which we can compute conditional probabilities. Finally, we can write:

$$P_M((w,t)_{1,n}) = \prod_{i=1}^{i=n} P_M((w_i,t_i) \mid (w_{i-1},t_{i-1}) \wedge (w_{i-2},t_{i-2})),$$

and the estimated cross-entropy can be written as:

$$\hat{\mathcal{H}} = -\frac{1}{n}\left[\sum_{i=1}^{i=n} \log P_M((w_i,t_i) \mid (w_{i-1},t_{i-1}) \wedge (w_{i-2},t_{i-2}))\right]. \tag{6}$$

The "deleted interpolation" [13] technique attempts to determine the parameters λ_i^h (i=1,3) such that the estimated cross-entropy $\hat{\mathcal{H}}$ is minimum. We first split the original corpus into (K-1) samples of text (representative of the language L) using a resampling plan technique such as *Leaving-one-out* [9, 16]. This set of samples is used during a training phase and only 1 sample is left for a testing phase. This process of producing training and test samples is repeated K times. During each training phase, we estimate the probabilities involved in Equation 6 and compute the initial values of λ_i^h (i=1,..,3). We evaluate the cross-entropy of the model for each test sample and depending on whether we decreased the cross-entropy, we accept or reject the initial λ_i^h values. If we reject these values, we try other values computed on another training sample and we perform the same operations as with the previous values of λ_i^h (i=1,..,3). The Leaving-one-out method enables us to exploit all K samples of a corpus and therefore try K different values of λ_i^h (i=1,..,3).

4 Solving Sparseness through Probabilistic Logic Paradigm

Classical methods dealing with the sparseness problem assumes that the probability estimate for a previously unseen cooccurrence is a function of the probability estimates for the word/tag pairs in the cooccurrence. For example, in the trigram models that we analyze in this paper, the probability $P((w_i, t_i) \mid (w_{i-1}, t_{i-1}) \wedge (w_{i-2}, t_{i-2}))$ of a conditioned word/tag (w_i, t_i) that has never been encountered during training following the conditioning word/tag pairs $((w_{i-1}, t_{i-1}), (w_{i-2}, t_{i-2}))$ is computed from the probability of the word/tag pair (w_i, t_i) [12]. This technique is based on an independence assumption on the cooccurrence of the pairs $((w_{i-1}, t_{i-1}), (w_{i-2}, t_{i-2}))$ and (w_i, t_i).

To circumvent this independence assumption, our approach uses informations captured by all bigrams that have "a logical impact" on the estimation of $P((w_i, t_i) \mid (w_{i-1}, t_{i-1}) \wedge (w_{i-2}, t_{i-2}))$. In fact, unlike the "back-off smoothing" technique which involves a set of one unigram, one bigram and one trigram, our method involves a set $\mathcal{M}$ constituted of three unigrams, two bigrams and one trigram. This set is:

$$\begin{aligned}\mathcal{M} = \{&(w_i, t_i), (w_{i-1}, t_{i-1}), (w_{i-2}, t_{i-2}), (w_i, t_i) \mid (w_{i-1}, t_{i-1}), (w_{i-1}, t_{i-1}) \mid (w_{i-2}, t_{i-2}),\\ &(w_i, t_i) \mid (w_{i-1}, t_{i-1}) \wedge (w_{i-2}, t_{i-2})\}\,.\end{aligned} \tag{7}$$

In other words, we try to capture all n-grams (n = 1,2) that logically entail the quantity $P((w_i, t_i) \mid (w_{i-1}, t_{i-1}) \wedge (w_{i-2}, t_{i-2}))$. Probabilistic logic paradigm enables us to express a *linear relationship* between probabilities of the elements contained in the set $\mathcal{M}$.

Since the quantity $\hat{P}((w_i, t_i) \mid (w_{i-1}, t_{i-1}) \wedge (w_{i-2}, t_{i-2}))$ is not reliable, then it should be treated separately. The truth value of this quantity is built from all unigrams and bigrams in $\mathcal{M}$ that entail it. A constant term K^h (that depends on the history h) is present in the linear combination that we develop using probabilistic logic. This constant term is expressed as a function of the nonreliable quantity $\hat{P}((w_i, t_i) \mid (w_{i-1}, t_{i-1}) \wedge (w_{i-2}, t_{i-2}))$. Doing so, we can view our approach as an extension of the back-off-smoothing model referenced by Equation 2. Our goal is to compute an optimal estimation for $P((w_i, t_i) \mid (w_{i-1}, t_{i-1}) \wedge (w_{i-2}, t_{i-2}))$. Finally our problem consists of determining an optimal (with respect to some sense) solution for the coefficients λ_i^h (i=1,..,5) such that:

$$\begin{cases} P((w_i, t_i) \mid (w_{i-1}, t_{i-1}) \wedge (w_{i-2}, t_{i-2})) \simeq \lambda_1^h \hat{P}(w_i, t_i) + \lambda_2^h \hat{P}(w_{i-1}, t_{i-1}) + \lambda_3^h \hat{P}(w_{i-2}, t_{i-2}) \\ +\lambda_4^h \hat{P}((w_i, t_i) \mid (w_{i-1}, t_{i-1})) + \lambda_5^h \hat{P}((w_{i-1}, t_{i-1}) \mid (w_{i-2}, t_{i-2})) + K^h \end{cases}. \tag{8}$$

Each probability involved in this decomposition is computed from a training corpus using the ML estimation, and K^h is the constant that depends on the history h of the configuration.

We will show in the next section how we use probabilistic logic tools to determine an estimation of $P((w_i, t_i) \mid (w_{i-1}, t_{i-1}) \wedge (w_{i-2}, t_{i-2}))$ by computing uniquely the parameters λ_i^h (i=1,..,5) and the constant K^h. We propose two different approaches to solve this problem. The first one is implicit, in which we compute only an estimation of $P((w_i, t_i) \mid (w_{i-1}, t_{i-1}) \wedge (w_{i-2}, t_{i-2}))$ without providing the linear relationship of Equation 8. Using Theorem 1 and Theorem 2, we propose a topological approach that bounds the set of solutions. Finally, using the entropy maximization criterion, we select one unique solution and provide with th the second approach that is explicit, in which we emphasize the linear relationship.

(W_i, T_i)	0 0 0 0 1 1 1 1
(W_{i-1}, T_{i-1})	0 0 1 1 0 0 1 1
(W_{i-2}, T_{i-2})	0 1 0 1 0 1 0 1
$(W_i, T_i) \wedge (W_{i-1}, T_{i-1})$	0 0 0 0 0 0 1 1
$(W_{i-1}, T_{i-1}) \wedge (W_{i-2}, T_{i-2})$	0 0 0 1 0 0 0 1
$(W_i, T_i) \wedge (W_{i-1}, T_{i-1}) \wedge (W_{i-2}, T_{i-2})$	0 0 0 0 0 0 0 1

Table 1: Logical-truth table assigned to the set of predicates. Lower cases are used to represent the joint random variable (W,T).

4.1 Probabilistic Logic Paradigm

To compute an estimation of the probability $P((w_i, t_i) \mid (w_{i-1}, t_{i-1}) \wedge (w_{i-2}, t_{i-2}))$, we propose a novel approach that explores the probabilistic logic (PL) paradigm [17, 5]. We make an analogy between word/tag configurations (w_i, t_i) and logical predicates. The predicate (w_i, t_i) is true if it is present in the corpus and false otherwise. The presence of (w_i, t_i) means that the word w_i with the particular part of speech tag t_i has been encountered or observed in the training corpus. If we were interested only in words and not with their parts of speech tags, then these predicates are equal 1 if the words are present in the corpus.

4.1.1 Mapping Between Worlds and Predicates

In order to compute this logical entailment, we need to pass from the conditional events to the joint events. We first define a linear relationship that involves joint probabilities and afterwards we express this relationship into conditional probability terms. Table 1 shows the logical truth-table assigned to this set of predicates.

As outlined previously, a predicate $\{(w_i, t_i)\}$ is either true or false. It is true if the word/tag pair (w_i, t_i) is observed (or encountered) in a corpus and false whenever it is absent. Eachtime, we obtain two worlds v_i, (i=1,2) that are assigned to a predicate and the values of the predicate are opposite in these two worlds. We define a probability distribution over the sets of possible worlds associated with the different sets of possible truth values of predicates [17]. This probability specifies for each set of possible worlds what is the probability $P(v_i)$ that the actual world is contained in v_i. The probabilities $P(v_i)$ are subject to the constraint $\sum_i P(v_i) = 1$ since the set of possible worlds are mutually exhaustive and exclusive. A mapping from the space of possible worlds to the space of predicates is built. A probability of any predicate (π_j) is the sum of the probabilities of the worlds (v_i) where the predicate is true. The linear equation mapping possible world probabilities and predicate probabilities is:

$$C \times P = \Pi, \tag{9}$$

$$C = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}, \tag{10}$$

Table 2: The consistent matrix assigned to the set of predicates.

where the consistent matrix whose columns are the possible worlds (v_i) represents this linear mapping between the space of possible worlds and the space of predicates. The vectors P and Π are defined as:

$$\begin{gathered} P = [P(v_1), P(v_2), ..., P(v_8)]^T, \\ \Pi = [P(w_i, t_i), P(w_{i-1}, t_{i-1}), P(w_{i-2}, t_{i-2}), P((w_i, t_i) \wedge (w_{i-1}, t_{i-1})), P((w_{i-1}, t_{i-1}) \wedge (w_{i-2}, t_{i-2})) \\ P((w_i, t_i) \wedge (w_{i-1}, t_{i-1}) \wedge (w_{i-2}, t_{i-2}))]^T, \end{gathered} \tag{11}$$

where "T" stands for the transpose form vector.

4.2 A Numeric Computation

Our goal is to find a numerical estimation for $P((w_i, t_i) \mid (w_{i-1}, t_{i-1}) \wedge (w_{i-2}, t_{i-2}))$. We first disregard the last row of the consistent matrix C and we add a tautology (all components = 1) as a first row of the matrix C. We thus transform the matrix C into C_d and Π into $\Pi_d = [1, P((w_i, t_i)), P(w_{i-1}, t_{i-1}), P(w_{i-2}, t_{i-2}), P((w_i, t_i) \wedge (w_{i-1}, t_{i-1})), P((w_{i-1}, t_{i-1}) \wedge (w_{i-2}, t_{i-2}))]^T$. The matrix C_d is illustrated by Table 3. The problem now consists to find P such that:

$$C_d \times P = \Pi_d.$$

This linear problem is illustrated by Table 4.

From the linear sytem of Table 4, we can extract the following constraints:

$$\begin{cases} -1 \leq -P(w_{i-1}, t_{i-1}) + P((w_{i-1}, t_{i-1}) \wedge (w_{i-2}, t_{i-2})) - P(w_{i-2}, t_{i-2}) \leq 1 \\ -1 \leq P((w_i, t_i) \wedge (w_{i-1}, t_{i-1})) - P(w_i, t_i) - P(w_{i-1}, t_{i-1}) \leq 1 \\ 0 \leq P(w_{i-1}, t_{i-1}) - P(w_{i-2}, t_{i-2}) \leq 2 \\ 0 \leq P((w_i, t_i) \wedge (w_{i-1}, t_{i-1})) \leq 2 \end{cases}. \tag{19}$$

These inequations enable us to obtain consistent constraints and positive solutions for the probabilities p_i (i=1,..,8) and *a-fortiori* a solution for our problem.

$$C_d = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \end{pmatrix}, \tag{12}$$

Table 3: The consistent matrix without the last row and with a tautology.

$$\begin{aligned} p_1 + p_2 + p_3 + p_4 + p_5 + p_6 + p_7 + p_8 &= 1 && (13) \\ p_5 + p_6 + p_7 + p_8 &= \hat{P}((w_i, t_i)) && (14) \\ p_3 + p_4 + p_7 + p_8 &= \hat{P}((w_{i-1}, t_{i-1})) && (15) \\ p_2 + p_4 + p_6 + p_8 &= \hat{P}((w_{i-2}, t_{i-2})) && (16) \\ p_7 + p_8 &= \hat{P}((w_i, t_i) \wedge (w_{i-1}, t_{i-1})) && (17) \\ p_4 + p_8 &= \hat{P}((w_{i-1}, t_{i-1}) \wedge (w_{i-2}, t_{i-2})) && (18) \end{aligned}$$

Table 4: The linear system of equations providing $p_i = P(v_i)$.

Since the solution of this problem is not unique (6 equations with 8 unknown variables), we select from among the possible solutions for P that P^* which maximizes the Shannon entropy S of the distribution P (or minimally discriminate among solutions) [11].
Our problem now consists of determining p_i $(i = 1, ..., 8)$ such that:

$$\begin{cases} \max\limits_{p_i}[S(p_i)] \\ C_d \times P = \Pi_d \end{cases} . \tag{20}$$

We are dealing with a nonlinear optimization problem with 6 linear constraints. The natural constraint $\sum_{i=1}^{i=8} p_i = 1$ is expressed in the first row of the matrix C_d.

Using *Lagrange multipliers* technique, the optimal solution can be written:

$$P^* = arg \max_{p_i} \left[-\sum_{i=1}^{i=8} p_i \times \log p_i - (\alpha_1 - 1) \left(\sum_{i=1}^{i=8} p_i - 1 \right) - \sum_{j=2}^{j=6} \alpha_j (L_j P - \pi_j) \right], \tag{21}$$

where L_j are the row vectors of the matrix C_d and $P(v_i) = p_i = P$ and α_j (j=1,..,6) are the Lagrange multipliers.

After differentiating this function with respect to p_i and setting the result to 0, we obtain the following optimal solution:

$$p_i^* = \exp^{(-\alpha_1)} \prod_{j=2}^{j=6} \exp^{(-\alpha_j \times u_{ji})} \quad (i = 1, .., 8), \tag{22}$$

where u_{ji} is the i-th component of the j-th row vector of C_d.
setting

$$x_j = \exp^{(-\alpha_j)}, (j = 1, .., 6) \tag{23}$$

Using these change of variables, we express the components p_i^* (i =1,...,8) as a function of x_j (j=1,...,6):

$$\begin{cases} p_1^* = x_1 \\ p_2^* = x_1 x_4 \\ p_3^* = x_1 x_3 \\ p_4^* = x_1 x_3 x_4 x_6 \\ p_5^* = x_1 x_2 \\ p_6^* = x_1 x_2 x_4 \\ p_7^* = x_1 x_2 x_3 x_5 \\ p_8^* = \prod\limits_{i=1}^{i=6} x_i \end{cases} . \tag{24}$$

The problem is transformed into a resolution of the nonlinear system of 6 equations illustrated in Table 5. The terms in the second members of this nonlinear system of equations are estimated from a corpus using a maximum likelihood criterion. Form Equation 9, we can write:

$$P((w_i, t_i) \mid (w_{i-1}, t_{i-1}) \wedge (w_{i-2}, t_{i-2})) = \frac{P((w_i, t_i) \wedge (w_{i-1}, t_{i-1}) \wedge (w_{i-2}, t_{i-2}))}{P((w_{i-1}, t_{i-1}) \wedge (w_{i-2}, t_{i-2}))} \simeq \frac{p_8^*}{p_4^* + p_8^*}. \tag{31}$$

$$x_1 + x_1x_4 + x_1x_3 + x_1x_3x_4x_6 + x_1x_2 + x_1x_2x_4 + x_1x_2x_3x_5 + x_1x_2x_3x_4x_5x_6 = 1 \quad (25)$$

$$x_1x_2 + x_1x_2x_4 + x_1x_2x_3x_5 + x_1x_2x_3x_4x_5x_6 = \hat{P}((w_i, t_i)) \quad (26)$$

$$x_1x_3 + x_1x_3x_4x_6 + x_1x_2x_3x_5 + x_1x_2x_3x_4x_5x_6 = \hat{P}((w_{i-1}, t_{i-1})) \quad (27)$$

$$x_1x_4 + x_1x_3x_4x_6 + x_1x_2x_4 + x_1x_2x_3x_4x_5x_6 = \hat{P}((w_{i-2}, t_{i-2})) \quad (28)$$

$$x_1x_2x_3x_5 + x_1x_2x_3x_4x_5x_6 = \hat{P}((w_i, t_i) \wedge (w_{i-1}, t_{i-1})) \quad (29)$$

$$x_1x_3x_4x_6 + x_1x_2x_3x_4x_5x_6 = \hat{P}((w_{i-1}, t_{i-1}) \wedge (w_{i-2}, t_{i-2})) \quad (30)$$

Table 5: Nonlinear system of equations extracted after the entropy maximization.

In this numerical approach, it is difficult to express some relationship between the second members of the nonlinear system of equations of Table 5.

4.3 A Symbolic Computation: Linear Interpolation:

So far, we have proposed a numerical approach for computing an estimation of $P((w_i, t_i) \mid (w_{i-1}, t_{i-1}) \wedge (w_{i-2}, t_{i-2}))$. This computation requires to resolve a nonlinear equation system for each unencountered trigram. Our goal now is to determine a linear relationship between all possible word/pos n-grams. We first propose a geometrical approach of the solution.

4.3.1 A Bounded Set of Solutions using a Geometric Approach

In order to express a *logical linear relationship* between all n-grams (n=1,2,3), we need a closed (or symbolic) form of the solution. We propose a method that is based on the following topological results:

Theorem 1 *Let* $\mathcal{P} = \{P = [p_1, p_2, ..., p_n]^T$ *such that* $\|P\|_1 = \sum_{i=1}^{i=n} p_i = 1\}$*, let* U *be a linear operator in* R^n*, and* $D = (d_{ij})$ *its associated matrix relative to the classical base of* R^n*. If* $\sum_i d_{ij} = n, \forall j$*, and* $d_{ij} \geq 0$*, then we can write the following assertions:*

(1) $U(\mathcal{P}) \subset n \times \mathcal{P}$*,*

(2) $\|\Pi\|_1 = n \times \|P\|_1 = n$*,*

where $\|.\|_1$ *is the* L_1 *norm (sum of components).*

$$D = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 7 & 6 & 6 & 4 & 6 & 5 & 4 & 1 \end{pmatrix} . \tag{32}$$

Table 6: The matrix D results from adding the two vectors $\mathcal{A}_1$ and $\mathcal{A}_2$ as two last rows.

Proof. $\|D \times P\|_1 = \sum_{i=1}^{i=n} \sum_{j=1}^{j=n} d_{ij} \times p_j = \sum_{j=1}^{j=n} p_j \sum_{i=1}^{i=n} d_{ij} = n \times \sum_{j=1}^{j=n} p_j$, and both of the relations (1) and (2) are proved.

To determine a closed form using Theorem 1, we add an *artificial row* $\mathcal{A}$ in the consistent matrix C such that the sum of the components for each column is equal to the dimension of the space which is 8 in this problem, see Table 6. The vector Π is therefore extended with two components $\mu^h(\mathcal{A}_1)$ which is a tautology and $\mu^h(\mathcal{A}_2)$ corresponding respectively to the row vectors $\mathcal{A}_1 = [1,1,1,1,1,1,1,1]^T$ and $\mathcal{A}_2 = [7,6,6,4,6,5,4,1]^T$. By adding a tautology which expresses exclusivity and exhaustivity, we made this two vector extension unique. The predicate vector Π is now extended and called Π_e. For a sake of brevity, $\mu^h(\mathcal{A}_2)$ is now called $\mu^h(\mathcal{A})$.

From equation $D \times P = \Pi_e$, we can deduce:

$$\mu^h(\mathcal{A}) = 7p_1 + 6(p_2 + p_3 + p_5) + 4(p_4 + p_7) + 5p_6 + p_8 \ . \tag{33}$$

Using Theorem 1, we obtain the following closed form:

$$\begin{aligned} P((w_i,t_i) \wedge (w_{i-1},t_{i-1}) \wedge (w_{i-2},t_{i-2})) = 7 - \mu^h(\mathcal{A}) - \hat{P}(w_i,t_i) - \hat{P}(w_{i-1},t_{i-1}) \\ -\hat{P}(w_{i-2},t_{i-2}) - \hat{P}((w_i,t_i) \wedge (w_{i-1},t_{i-1})) - \hat{P}((w_{i-1},t_{i-1}) \wedge (w_{i-2},t_{i-2})). \end{aligned} \tag{34}$$

Using Bayes' rule, we finally express the conditional probability as:

$$P((w_i,t_i) \mid (w_{i-1},t_{i-1}) \wedge (w_{i-2},t_{i-2})) \simeq \frac{7-\mu^h(\mathcal{A})-\hat{P}(w_i,t_i)-\hat{P}(w_{i-1},t_{i-1})-\hat{P}(w_{i-2},t_{i-2})-\hat{P}((w_i,t_i)\wedge(w_{i-1},t_{i-1}))}{Q} - 1 \ , \tag{35}$$

where the quantity $Q = \hat{P}((w_{i-1},t_{i-1}) \wedge (w_{i-2},t_{i-2}))$ is an estimation of the probability assigned to the history of the configuration. This latter quantity has to be different from 0.

Given this latter equation, we can identify K^h as:

$$K^h = \frac{(7-\mu^h(\mathcal{A}))}{Q} - 1 \ . \tag{36}$$

In order to emphasize the coefficients λ_i^h (i=1,...,5) and to obtain *a generalized form of the back-off technique*, we write Equation 35 into the following:

$$\begin{array}{l} P((w_i,t_i) \mid (w_{i-1},t_{i-1}) \wedge (w_{i-2},t_{i-2})) \simeq \frac{-1}{Q}\Big[\hat{P}(w_i,t_i) + \hat{P}(w_{i-1},t_{i-1}) + \hat{P}(w_{i-2},t_{i-2}) \\ +\hat{P}((w_i,t_i) \mid (w_{i-1},t_{i-1})) \times \hat{P}(w_{i-1},t_{i-1})\Big] + 0 \times \hat{P}((w_{i-1},t_{i-1}) \mid (w_{i-2},t_{i-2}))+ \\ \frac{7-\mu^h(\mathcal{A})-Q}{Q\times\hat{P}((w_i,t_i)|(w_{i-1},t_{i-1})\wedge(w_{i-2},t_{i-2}))} \times \hat{P}((w_i,t_i) \mid (w_{i-1},t_{i-1}) \wedge (w_{i-2},t_{i-2})) \end{array} \ . \tag{37}$$

We finally can write all the coefficients of the interpolation of Equation 8 in the case where Q is different from 0:

$$\begin{cases} \lambda_i^h = \frac{-1}{Q} \quad \forall i = 1,..,3 \\ \lambda_4^h = \frac{-\hat{P}(w_{i-1},t_{i-1})}{Q} \\ \lambda_5^h = 0 \\ K^h = \frac{7-\mu^h(\mathcal{A})-Q}{Q} \end{cases} \ . \tag{38}$$

We can notice that the final solution depends only on the quantity $\mu^h(\mathcal{A})$ which itself depends on the optimal solution P^*. It appears also that the decomposition of the trigram probability obtained using probabilistic logic does not involve the sparse trigram computed from the training corpus. This result is predictable since this latter quantity is not reliable (using a ML estimation) and therefore needs to be disregarded. The bigram probability $P((w_{i-1},t_{i-1}) \mid (w_{i-2},t_{i-2}))$ has been simplified and does not appear in this decomposition. However, according to Equation 4, it appears that the term $\mu^h(\mathcal{A})$ depends on this bigram probability.

Remark 1 *The solution $P^* = (p_i^*)$, $(i = 1,...,n)$ obtained using Jaines MAXENT criterion on the set of constraints of Table 5 is not consistent with the additional contraint $\mu^h(\mathcal{A})$ and therefore might provide a solution for $P((w_i,t_i) \mid (w_{i-1},t_{i-1}) \wedge (w_{i-2},t_{i-2}))$ which is greater than 1.*

Therefore, a consistent solution for the vector P needs to be found. The following topological results enable us to better understand the structure of this solution.

Proposition 1 *The sets $\mathcal{P}$, $\frac{U(\mathcal{P})}{n}$, $\frac{U^2(\mathcal{P})}{n^2}$,..., $\frac{U^k(\mathcal{P})}{n^k}$,..., form a decreasing sequence with an intersection $\mathcal{D}$ different from $\emptyset$.*

Proof: $\frac{U^k(\mathcal{P})}{n^k}$ is a decreasing sequence when applying Theorem 1, so any finite intersection of the sequence is different from $\emptyset$. Besides, $\forall k$, U_k is a continuous function since it is a linear operator in a finite dimension space. Therefore, $\frac{U^k(\mathcal{P})}{n^k}$, and $\bigcap_k \frac{U^k(\mathcal{P})}{n^k} = \mathcal{D} \neq \emptyset$.

Proposition 2 *The following assertions are verified:* $U(\mathcal{D}) = \mathcal{D}$ *and for each probability vector* $P \in \mathcal{D}$, $U(P) \in \mathcal{D}$.

Proof: If $P \in \mathcal{D}$, $\forall k \geq 0$, $P \in \frac{U^k(\mathcal{P})}{n^k}$, therefore $U(P) \in \frac{U^{k+1}(\mathcal{P})}{n^k}$, i.e., $U(P) \in \mathcal{D}$. If $Q \in \mathcal{D}$, by the definition of $\mathcal{D}$, $\forall k \geq 0$, $\exists P_k \in \frac{U^k(\mathcal{P})}{n^k}$ such that $Q = U(P_k)$. Since the set $\mathcal{P}$ is compact, then the sequence P_k has at least one adherence value P, then $P \in \mathcal{D}$, since $P_k \in \bigcap_k \frac{U^i(\mathcal{P})}{n^i}$ with $i \leq k$, and the $\frac{U^i(\mathcal{P})}{n^i}$ are closed and decreasing. Finally, as $U(\mathcal{D}) \subset \mathcal{D}$ and $\mathcal{D} \subset U(\mathcal{D})$, therefore $U(\mathcal{D}) = \mathcal{D}$.

Theorem 2 *If* $D(k) = (d_{ij}(k))$ *is the matrix associated with* $\frac{U^k}{n^k}$, *relative to the canonical base of* $\Re^n$, *then a strictly increasing sequence,* k_1, k_2,..., *of natural numbers exists so that:* $\forall (i,j) \in N^2$ *(natural numbers), the sequence* $(d_{ij}(k_p)) \longrightarrow \Lambda = (\Lambda_{ij})$ *as* $p \longrightarrow \infty$, *and* $\Gamma(\mathcal{P}) = \mathcal{D}$, *where* Γ *is the limit linear operator associated with the matrix* $\Lambda = (\Lambda_{ij})$.

Proof: $\forall (i,j,k) \in N^3$, we can write $0 \leq d_{ij}(k) \leq 1$; so $\forall (i,j) \in N^2$, the sequence (d_{ij}^k) possesses an adherence value (Bolzano-Wiertrass). By setting in order the finite set of indices i and j, we can by n^2 extractions of the sequence, find an increasing sequence of natural numbers (k_p) so that, $\forall (i,j) \in N^2$, the sequence $(d_{ij}(k_p)) \longrightarrow \Lambda = (\Lambda_{ij})$ (adherence value) when $p \longrightarrow \infty$. Besides, $\forall p \geq 1$, we can write using the definition of Γ: $\Gamma(\mathcal{P}) \subset \frac{U^{k_p}(\mathcal{P})}{n^{k_p}}$, therefore $\Gamma(\mathcal{P}) \subset \mathcal{D}$ since $\frac{U^i(\mathcal{P})}{n^i}$ is a decreasing sequence. Let $P \in \mathcal{D}$, $\forall p \in N$ (natural number), we can find $P_{k_p} \in \mathcal{P}$ such that $P = \frac{U^{k_p}(P_{k_p})}{n^{k_p}}$. If Q is an adherence value assigned to the sequence P_{k_p}, we prove the result by using, for example the following inequality:

$$\|\frac{U^{k_p}(P_{k_p})}{n^{k_p}} - \Gamma(Q)\| \leq \|\frac{U^{k_p}(P_{k_p})}{n^{k_p}} - \Gamma(P_{k_p})\| + \|\Gamma(P_{k_p}) - \Gamma(Q)\|. \tag{39}$$

Proposition 3 *The set* $\mathcal{P}$ *is the convex hull relative to the canonical base of* $\Re^n$, $\{e_i\}$ *and* $\mathcal{D}$ *is the convex hull relative to the image base* $\{\Gamma(e_i)\}$.

Proof: Since Γ is a linear operator (continuous function in a finite space) and $\mathcal{P}$ is a convex hull, therefore its image $\mathcal{D}$ remains convex.

Proposition 4 *Any point* $Q \in \mathcal{D}$ *can be written as a convex combination associated with* $\{\Gamma(e_i)\}$,

$$\begin{cases} Q = \sum_{i=1}^{i=n} \alpha_i \Gamma(e_i), \\ \sum_{i=1}^{i=n} \alpha_i = 1, \\ \forall i, \alpha_i \in [0..1] \end{cases} . \tag{40}$$

Proof: This is due to the fact that $\{\Gamma(e_i)\}$ are the vectors that generate the convex hull $\mathcal{D}$.

Proposition 5 *If the matrix $D(k) = d_{ij}(k)$ is multiplied by itself a large number of times, a level of stability (or attraction) is reached, any vector $\Pi_e \in \mathcal{D}$ is an image of a probability vector $P \in \mathcal{D}$.*

Proof: Because the set $\mathcal{D}$ is invariant by the endomorphism U.

4.3.2 Analysis of the Topological Results

These results enables us to bound the set of solutions for the possible world vector P^*. We now know how to determine this convex set $\mathcal{D}$ of solutions by a convergence process of the matrix D. The followings results have been obtained:

- $\Gamma(\mathcal{P}) = \mathcal{D}$:
  $\forall Y = (y_i) = \frac{U}{n}(\frac{U}{n}(\frac{U}{n}...(\frac{U}{n}(P)))) \ (P \in \mathcal{P}), \exists P' \in \mathcal{D};\ Y = P'$ (thus: $\sum_i y_i = 1$)
- $\mathcal{D} \subset \frac{U(\mathcal{P})}{n}$:
  $\forall P \in \mathcal{D}, \exists \Pi \in U(\mathcal{P});\ \Pi = n \times P$
- $U(\mathcal{D}) = \mathcal{D}$:
  $\forall \Pi = (\pi_i) \in U(\mathcal{D}),\ \exists P = (p_i) \in \mathcal{D};\ \Pi = P$ (thus $\sum_i \pi_i = 1$).

These results does not provide a unique solution for the vector P but tell us that a vector $\frac{\Pi^k}{n^k}$ (k large enough!) behaves as a probability vector verifying the natural constraint ($\sum_i \frac{\pi_i^k}{n^k} = 1$). Even if we have obtained $U(\mathcal{D}) = \mathcal{D}$ (invariance property), there is still an infinity of solutions. Besides, since $\forall P \in \mathcal{P}$, $D \times P = \Pi$, therefore by a composition process, we obtain: $\frac{1}{n^2}D^2(P) = \frac{1}{n^2}D(\Pi)$. By iterating this composition process a finite number of times, we can see that the maximum entropy function decreases at each iteration until we reach the convex set $\mathcal{D}$ where its value is minimal.

Proposition 6 *The maximum entropy is a decreasing function with respect to the composition process by the operator $\frac{U^k}{n^k}$ $(n = 8)$ and $(k = 1, 2, ...)$.*

Proof: Because $\forall P \in \mathcal{P}$, $D \times P = \Pi \Leftrightarrow P = D^{-1} \times \Pi$, (D nonsingular matrix) a component of this vector can be written as: $p_i = \sum_{j=1}^{i=8} d_{ij}^{-1}\pi_j$. Besides, since $\forall P \in \mathcal{D}$, $\exists \Pi' \in U(\mathcal{P})$;
$D \times P = \Pi' \Leftrightarrow P = D^{-1} \times \Pi'$, therefore a component of this vector P can be written: $p_i = \sum_{j=1}^{i=8} d_{ij}^{-1}\pi'_j$. However, the components π'_j that belongs to the vectors in $\mathcal{D}$ are smaller than the componemts π_j of vectors in $U(\mathcal{P})$. Therefore, the entropy $\sum_{i=1}^{i=8}(\sum_{j=1}^{i=8} d_{ij}^{-1}\pi'_j \log(\sum_{j=1}^{i=8} d_{ij}^{-1}\pi'_j))$ in this latter case is smaller.

Because we have only bounded the set of solutions for the vector P, we now explore the Jaines maximum entropy criterion for selecting a unique solution.

4.3.3 The Maximum Entropy Criterion on a Larger Set of Constraints

By considering the additional constraint whose second member is $\mu^h(\mathcal{A})$, the total number of constraints becomes 8. This set of constraints is composed of the rows of the matrix D. We maximized $S(p_i)$ under this set of 8 constraints. Using the Lagrange mutipliers technique, the optimal solution found is:

$$P^* = arg \max_{p_i} \left[-\sum_{i=1}^{i=8} p_i \times \log p_i - (\alpha_1 - 1) \left(\sum_{i=1}^{i=8} p_i - 1 \right) - \sum_{j=2}^{j=8} \alpha_j (L_j P - \pi_j^e) \right], \tag{41}$$

where L_j are the row vectors of the matrix D and α_j (j=1,...,8) are the Lagrange mutipliers. The set of solutions for p_i (i=1,...,8) as a function of x_i (i=1,...,8) is:

$$\begin{cases} p_1^* = x_1 (x_8)^7 \\ p_2^* = x_1 x_4 (x_8)^6 \\ p_3^* = x_1 x_3 (x_8)^6 \\ p_4^* = x_1 x_3 x_4 x_6 (x_8)^4 \\ p_5^* = x_1 x_2 (x_8)^6 \\ p_6^* = x_1 x_2 x_4 (x_8)^5 \\ p_7^* = x_1 x_2 x_3 x_5 (x_8)^4 \\ p_8^* = \prod_{i=1}^{i=8} x_i \end{cases} . \tag{42}$$

As in section 4.2, the problem is transformed into a resolution of a nonlinear system of 8 equations with 9 unknown variables (x_i $(i = 1, ..., 8) + \mu^h(\mathcal{A})$). The second member $P((w_i, t_i) \wedge (w_{i-1}, t_{i-1}) \wedge (w_{i-2}, t_{i-2}))$ has been replaced by its value found in the numerical approach. We solved this constrained optimization problem using MAPLE Equation Solver. Since there are 8 equations and 9 unknown variables, therefore the solution will depend on 1 free variable. However, the value of $\mu^h(\mathcal{A})$ is constant which does not depend on this free variable. We have replaced the value of $\mu^h(\mathcal{A})$ in the expression of K^h and determined the linear decomposition of the quantity $P((w_i, t_i) \mid (w_{i-1}, t_{i-1}) \wedge (w_{i-2}, t_{i-2}))$.

4.4 Sparseness Algorithm

We describe the algorithm that should be invoked in the case of sparse data. This algorithm summarizes the different points of this work. It receives as input the different n-grams (n=1,...,2) probabilities and compute an estimation of $P((w_i, t_i) \mid (w_{i-1}, t_{i-1}) \wedge (w_{i-2}, t_{i-2}))$ using the numeric and symbolic approaches.

Algorithm SparseDataPL
begin
`Compute an estimation of =` $P((w_i, t_i) \mid (w_{i-1}, t_{i-1}) \wedge (w_{i-2}, t_{i-2}))$

```
(* This computation is performed from the training corpus using a ML estimation *)
 Read the data P(w_i,t_i), P(w_{i-1},t_{i-1}), P(w_{i-2},t_{i-2}), P((w_i,t_i) ∧ (w_{i-1},t_{i-1})), and
P((w_{i-1},t_{i-1}) ∧ (w_{i-2},t_{i-2}))
 Solve the nonLinear system with respect to x_i (i=1,...,6)
 Compute p*_i (i=1,...,8)
 if (Numeric computation ==true)
  then
    Compute the PL estimation of the 3-grams: p*_8 / (p*_4 + p*_8)
 else (*if Symbolic computation == true*)
  then
   begin
    Decompose the model into a linear combination
    Compute the quantity μ^h(A) using MAXENT
    Extract λ^h_i (i=1,..,5)
    Write the linear decomposition of P((w_i,t_i) | (w_{i-1},t_{i-1}) ∧ (w_{i-2},t_{i-2}))
    w.r.t. Equation 8.
   end
end.{SparseDataPL}
```

5 Effect of the Solution on the Cross-Entropy Minimization

Let's consider a corpus of n word/tags where there are only k sparse trigrams in it. In order to use the definition of cross-entropy of Equation 6, n should be large ($n \longrightarrow \infty$). Minimizing the cross entropy is equivalent to minimizing the k terms $P((w_i, t_i) \wedge (w_{i-1}, t_{i-1}) \wedge (w_{i-2}, t_{i-2}))$ corresponding to the sparse trigram events. Using the results we have obtained, this is equivalent to maximizing $\sum_{i=1}^{i=k} \log p_8^i$, where the optimum for each trigram is obtained for $p_8^j = 1 \ \forall j = 1, .., k$ and therefore $p_i^j = 0 \ \forall i = 1, .., 7$ and $j = 1, ..k$. This latter condition implies $P((w_i, t_i) \mid (w_{i-1}, t_{i-1}) \wedge (w_{i-2}, t_{i-2})) = 1$ which is predictable for an optimum perplexity. Besides, the linear relationship presented in the the symbolic approach was possible to express because we have used the optimal solution $\hat{P}((w_i, t_i) \mid (w_{i-1}, t_{i-1}) \wedge (w_{i-2}, t_{i-2}))$ obtained from the matrix C_d of the numeric approach. The term $\mu^h(\mathcal{A})$ was computed using this solution.

6 Experiment

We have applied this technique to estimate some word/tag trigram probabilities. The underlined trigrams in the following set of phrases are sparse in the *Wall Street Journal* corpus.

---

1. we are expecting <u>the closest inspection</u>
2. except <u>for feeling mad</u> i recall nothing

3. time flies like an arrow
4. we still prefered spring

We show as an illustration how we compute $P((w_i, t_i) \mid (w_{i-1}, t_{i-1}) \wedge (w_{i-2}, t_{i-2}))$ using MAPLE Equation Solver in the first sparse trigram of the first phrase:

the closest inspection

(the,DT) (closest,ADV) (inspection,Noun) .

In the examples of sparse trigrams that have been tested, the third word/tag is the ith couple (w_i, t_i), the second is (w_{i-1}, t_{i-1}) and the first is (w_{i-2}, t_{i-2}).

6.1 Numeric Approach

In this case $P((w_i, t_i) \mid (w_{i-1}, t_{i-1}) \wedge (w_{i-2}, t_{i-2}))$ is approximately equal to $\frac{p_8^*}{p_4^* + p_8^*}$.

$eq1 := x_1 + x_1 * x_4 + x_1 * x_3 + x_1 * x_3 * x_4 * x_6 + x_1 * x_2 + x_1 * x_2 * x_4 + x_1 * x_2 * x_3 * x_5 + x_1 * x_2 * x_3 * x_4 * x_5 * x_6 = 1$
$eq2 := x_1 * x_2 + x_1 * x_2 * x_4 + x_1 * x_2 * x_3 * x_5 + x_1 * x_2 * x_3 * x_4 * x_5 * x_6 = .250$
$eq3 := x_1 * x_3 + x_1 * x_3 * x_4 * x_6 + x_1 * x_2 * x_3 * x_5 + x_1 * x_2 * x_3 * x_4 * x_5 * x6 = .200$
$eq4 := x_1 * x_4 + x_1 * x_3 * x_4 * x_6 + x_1 * x_2 * x_4 + x_1 * x_2 * x_3 * x_4 * x_5 * x_6 = .400$
$eq5 := x_1 * x_2 * x_3 * x_5 + x_1 * x_2 * x_3 * x_4 * x_5 * x_6 = .10e - 1$
$eq6 := x_1 * x_3 * x_4 * x_6 + x_1 * x_2 * x_3 * x_4 * x_5 * x_6 = .10e - 1$
`solve(eq1,eq2,eq3,eq4,eq5,eq6);`
$x_6^* = .05533063428$, $x_1^* = .2850000000$, $x_3^* = .6289198606$,
$x_4^* = .95121951222$, $x_2^* = .4285714286$, $x_5^* = .1228070175$

Substituting the solution x_i by p_j^*, we obtain:
$p_1^* = .28500000000$, $p_2^* = .27000000000$, $p_3^* = .18000000000$, $p_4^* = .01000000000$, $p_5^* = .12000000000$, $p_6^* = .12000000000$, $p_7^* = .01000000000$, $p_8^* = .00100000000$.

We can notice that the natural constraint $\sum_{i=1}^{i=8} p_i^* = 1$ is respected.
Finally, we can compute the trigram probability estimation:

$$P((w_i, t_i) \mid (w_{i-1}, t_{i-1}) \wedge (w_{i-2}, t_{i-2})) \simeq \frac{p_8^*}{p_4^* + p_8^*} = \frac{.00100000000}{.01000000000 + .00100000000} = .10000000000 \tag{43}$$

6.2 Symbolic Approach

This method requires the computation of the term $\mu^h(\mathcal{A})$. This is obtained from the solutions of The MAXENT criterion (with the 8 constraints) problem:

$eq1 := x_1 * x_8^7 + x_1 * x_4 * x_8^6 + x_1 * x_3 * x_8^6 + x_1 * x_3 * x_4 * x_6 * x_8^4 + x_1 * x_2 * x_8^6 +$
$x_1 * x_2 * x_4 * x_8^5 + x_1 * x_2 * x_3 * x_5 * x_8^4 + .001 = 1$
$eq2 := x_1 * x_2 * x_8^6 + x_1 * x_2 * x_4 * x_8^5 + x_1 * x_2 * x_3 * x_5 * x_8^4 + .001 = .250$
$eq3 := x_1 * x_3 * x_8^6 + x_1 * x_3 * x_4 * x_6 * x_8^4 + x_1 * x_2 * x_3 * x_5 * x_8^4 + .001 = .200$
$eq4 := x_1 * x_4 * x_8^6 + x_1 * x_3 * x_4 * x_6 * x_8^4 + x_1 * x_2 * x_4 * x_8^5 + .001 = .400$
$eq5 := x_1 * x_2 * x_3 * x_5 * x_8^4 + .001 = .010$
$eq6 := x_1 * x_3 * x_4 * x_6 * x_8^4 + .001 = .010$
$eq7 := x_1 * x_2 * x_3 * x_4 * x_5 * x_6 * x_7 * x_8 = .001$
$eq8 := 7 * x_1 * x_8^7 + 6 * (x_1 * x_4 * x_8^6 + x_1 * x_3 * x_8^6 + x_1 * x_2 * x_8^6) +$
$4 * (x_1 * x_3 * x_4 * x_6 * x_8^4 + x_1 * x_2 * x_3 * x_5 * x_8^4) + 5 * x_1 * x_2 * x_4 * x_8^5 + .001 = \mu^h(\mathcal{A})$
solve(eq1,eq2,eq3,eq4,eq5,eq6,eq7,eq8)
$\{x_8 = x_8,\ x_7 = 2.234567901 * x_8,\ x_3 = .6306620209 * x_8,\ \mu^h(\mathcal{A}) = 6.129000000,$
$x_2 = .4285714286 * x8,\ x_4 = .9512195122 * x_8,\ x_6 = .05227369316 * x_8,$
$x_1 = \frac{.2870000000}{x_8^7},\ x_5 = .1160220994 * x_8\}$

Finally, the coefficients λ_i^h of the linear decomposition of $P((w_i, t_i) \mid (w_{i-1}, t_{i-1}), (w_{i-2}, t_{i-2}))$ can be written as:

$$\begin{cases} \lambda_i^h = \frac{-1}{.10e-1} = -100 \quad \forall i = 1, .., 3 \\ \lambda_4^h = \frac{-.20}{.10e-1} = -20 \\ \lambda_5^h = 0 \\ K^h = \frac{7-6.12900000000-.10e-1}{.10e-1} = 86.10000000000 \end{cases} . \tag{44}$$

We have computed an estimation of the three sparse events using the numerical and symbolic approach. The results are illustrated in Table 7 and Table 8.

7 Conclusion and Future Work

A novel approach based on probabilistic logic is proposed to solve the problem of sparse events (or configurations). This approach is general, it can be used either in data mining, natural language processing, ZIP code recognition [2], or other field where data can be seen as events (or predicates). It exploits all logical interactions between subconfigurations that compose the sparse event. Unlike other classical approaches, our technique does not involve the non reliable trigram estimated from a corpus using the ML estimation. However, the hyperplane that approximates the solution does not cross the origine point. The solution we have obtained is an approximation of the optimum. We have experimented this technique on some sparse phrases and showed how we compute the interpolation coefficients. It also appeared that the perplexity value depends on the amount of informations we have already observed on all subconfigurations (unigrams, bigrams) composing the

Sparse events:	"the closest inspection"	"for feeling mad"	"like an arrow"	"still prefered spring"
Estimated Probabilities:				
$\hat{P}(w_i, t_i)$	.250	.180	.170	.150
$\hat{P}(w_{i-1}, t_{i-1})$	.200	.250	.280	.350
$\hat{P}(w_{i-2}, t_{i-2})$	.400	.600	.450	.290
$\hat{P}((w_i, t_i) \wedge (w_{i-1}, t_{i-1}))$	.10e-1	.030	.035	.036
$\hat{P}((w_{i-1}, t_{i-1}) \wedge (w_{i-2}, t_{i-2}))$	.10e-1	.033	.030	.032
$\hat{P}((w_i, t_i) \mid (w_{i-1}, t_{i-1}) \wedge (w_{i-2}, t_{i-2}))$	.100	.060	.123	.100

Table 7: Evaluation of sparse trigram probabilities using the numeric approach.

Sparse events:	"the closest inspection"	"for feeling mad"	"like an arrow"	"still prefered spring"
coefficients λ_i^h:				
λ_1^h	-100	-30.300	-33.333	-31.254
λ_2^h	-100	-30.300	-33.333	-31.254
λ_3^h	-100	-30.300	-33.333	-31.254
λ_4^h	-20	-6.060	-9.333	-10.938
λ_5^h	0	0	0	0
K^h	86.100	32.001	31.001	26.001

Table 8: Linear decomposition of sparse trigram probabilities using the symbolic approach.

sparse trigram. Besides, we have to investigate how the second member value $\mu^h(\mathcal{A})$ of the symbolic approach varies with respect to the solution $P((w_i, t_i) \mid (w_{i-1}, t_{i-1} \wedge (w - i - 2, t_{i-2}))$ determined via the numeric approach. We need to compare the perplexity of our model with the one obtained using classical methods on a same corpus of texts. These issues are parts of our next future work.

References

[1] ARPA, "Human Language Technology", Proceedings of a Workshop Held at Plainsboro, New Jersey, Sponsored by Advanced Research Projects Agency, Information Science and Technology Office, San Francisco, California: Morgan Kaufmann Publishers, Inc., 1994.

[2] D. Bouchaffra, V. Govindaraju and S.N.Srihari, "A Methodology for Deriving Probabilistic Correctness from Recognizers" , in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (CVPR'98), Santa Barbara, California, June 23-25, 1998.

[3] D. Bouchaffra, E. Koontz, V Kripasundar and R. K. Srihari, "Integrating Signal and Language Context to Improve Handwritten Phrase Recognition: Alternative Approaches", in *Preliminary Papers of the Sixth International Workshop on Artificial Intelligence and Statistics*, Fort-Lauderdale, Florida, Jan., 4-7, 1997.

[4] D. Bouchaffra, E. Koontz, V Kripasundar and R. K. Srihari, "Incorporating diverse information sources in handwriting recognition postprocessing", in *International Journal of Imaging Systems and Technology*, Vol. 7, pp 320-329, published by John Wiley & Sons Inc., 1996.

[5] D. Bouchaffra, "Theory and Algorithms for Analysing the Consistent Region in Probabilistic Logic", in *An International Journal of Computers & Mathematics*, vol. 25, No. 3, pp. 13,25, Pergamon Press, 1993.

[6] P.F. Brown, V. J. Della Pietra, P. V. deSouza, J. C. Lai and R. L. Mercer, "Class-based n-grams models of natural language", in *Computational Linguistics*, 18(4): 467-479, 1992.

[7] K. Church and W. Gale, "Enhanced Good-Turing and Cat-Cal: Two New Methods for Estimating Probabilities of English Bigrams", in *Computer Speech and Language*, 1990.

[8] I. Dagan, F. Pereira and L. Lee, "Similarity-Based Estimation of Word Cooccurrence Probabilities", in *Proc. of the 32nd Annual Meeting of the Association for Computational linguistics*, New Mexico State University, June 1994.

[9] B. Efron, "The Jacknife, the Bootstrap, and other Resampling Plans", Philadelphia: Soc. Industrial and Applied mathematics, 1982.

[10] I.J. Good, "The population frequencies of species and the estimation of population parameters", in *Biometrika*, 40 pp. 237-264, 1953.

[11] E.T. Jaynes, "Information Theory and Statistical Mechanics", in *Physical Reviews*, 106 : 620-630, 1957.

[12] F. Jelinek, R. Mercer and S. Roukos, "Principles of lexical language modeling for speech recognition", in *Sadaoki Furui and M. Moham Sondhi, editors, Advances in Speech Signal Processing*, Mercer Dekker, Inc. pages 651-699, 1992.

[13] F. Jelinek, J. Lafferty and R. Mercer, "Interpolated Estimation of Markov Source Parameters from Sparse Data", in *Pattern Recognition in Practice*, pp. 381-397, North Holland, 1980.

[14] S. M. Katz, "Estimation of probabilities from sparse data for the language model component of a speech recognizer", in *IEEE Transactions on Acoustics, Speech and Signal Processing*, 35(3): 400-401, 1987.

[15] A. Nadas, "Estimation of probabilities in the language model of the IBM speech recognition system", in *IEEE Trans. Acoustic, Speech and Signal Processing*, vol. 32, pp. 859-861, 1984.

[16] H. Ney, U. Essen and R. Kneser, "On the Estimation of Small Probabilities by Leaving-One-Out", in *IEEE, Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, NO. 12, 1995.

[17] N. J. Nilsson, "Probabilistic Logic", in *Artificial Intelligence*, vol. 28, pp. 71-87, Elsevier, North-Holland, 1986.

[18] R. Rosenfeld, "Adaptive Statistical Language Modeling: A Maximum Entropy Approach", in *PhD thesis*, School of Computer Science, Carnegie Mellon University, 1994.