CONDITIONAL RANDOM FIELD IN SEGMENTATION AND NOUN PHRASE INCLINATION TASKS FOR RUSSIAN

Kudinov M. S. (m.kudinov@samsung.com)

Samsung R&D Institute Russia, Moscow, Russia;
Dorodnitsyn Computing Center RAS, Moscow, Russia

Romanenko A. A. (a.romanenko@samsung.com)

Samsung R&D Institute Russia, Moscow, Russia;
Moscow Institute of Physics and Technology, Moscow, Russia

Piontkovskaja I. I. (p.irina@samsung.com)

Samsung R&D Institute Russia, Moscow, Russia

We propose solutions of several NLP problems for Russian making use of the conditional random fields (CRF) framework, including: shallow parsing (chunking), temporal expressions extraction and noun phrase inflection. Each of the three problems are important in speech generation, data mining and spoken dialogs systems design. The purpose of shallow parsing is to extract from the text syntactically related word forms (e.g. noun phrases) without full parsing. It may be useful in data mining applications. Temporal expressions extraction is important for natural language understanding modules of spoken dialog systems. Usually rule-based methods are used to address this problem. Noun phrase inflection is needed for speech generation modules. The main problem is to detect word forms for inflection. For all three problems statistical approach was taken. We use simple version of CRF named linear-chain CRF. In shallow parsing and time expressions extraction state-of-the-art results were achieved. In noun phrase inflection, the level of F_1-measure exceeded 95.

Key words: NLP, conditional random field, CRF, chunking, shalow parsing, temporal expressions extraction, noun phrase inflection

1. Introduction

It has been shown that a number of popular NLP tasks can be considered as the sequence labeling problem with certain output vocabulary. The typical ones are POS tagging, shallow parsing (chunking), temporal expression extraction and co-reference resolution. In bioinformatics and natural language processing the sequence labeling problem can be stated as finding the optimal mapping between an input sequence in an alphabet D onto a an output sequence in an alphabet L. During the past decades a number of both rule-based [1] and statistical [14] methods for solving this problem have been proposed. There is much evidence (e.g. [13]) of higher

performance demonstrated by linear-chain conditional random fields (L-CRF) as a statistical tool for machine learning algorithms. L-CRF is a discriminative model and in this aspect it resembles the popular Maximum entropy Markov model (MEMM). However, it was demonstrated ([3,7]) that MEMM has a considerable flaw named *label bias*. The problem is that the learning algorithm of MEMM causes the model to be more likely to choose hidden sates with lower entropy of transition probability distribution. For instance in POS tagging task MEMM will tend to choose those tags which prefer fewer types of followers. L-CRF was successfully applied for POS tagging in [7]. It was also successfully applied for shallow parsing in 13 and for co-reference resolution in [5].

Application of L-CRF to Russian is observed in [2]. In this paper POS tagging, co-reference resolution and sentiment analysis tasks have been considered.

In the present paper we propose a solution of three NLP tasks applied to Russian: temporal expression extraction, shallow parsing and inflection of noun phrases. We show that all these tasks may be reduced to sequence labeling problem and solved by means of L-CRF model. The remainder of the paper is organized as follows. In the next paragraph we give a short mathematical description of the L-CRF model; the next paragraphs are dedicated to the problems enumerated above. Each part includes description of the task, description of the datasets and the experimental results.

2. Linear-Chain CRF

CRF is a discriminative probabilistic graphical model defined induced by a non-oriented graph. The vertices of the graph correspond to random variables and edges correspond to probabilistic relations between them. In fact CRF is a graphical representation of a joint distribution $(Y_1, \ldots, Y_s)$ conditioned on observed data $(t_1, \ldots, t_s)$, where ti is a vector of observed features. Let Y_1^s and t_1^s be label and observation vector sequences from 1 to s correspondingly. Then we write the distribution of hidden label sequence conditioned on the observed feature vector sequence: $P(Y_1^s \mid t_1^s)$.

Consider the model in detail. The corresponding graph is given in Fig.1.

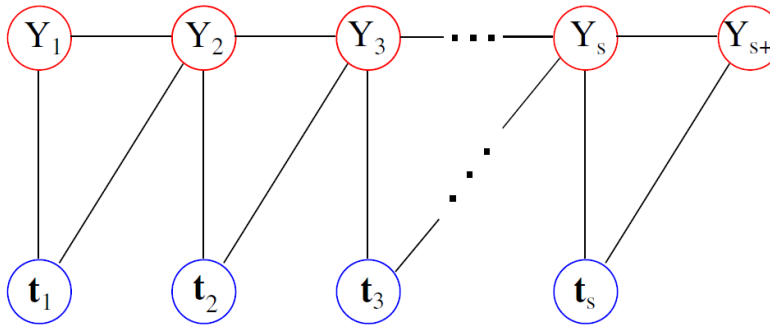

Fig. 1. The grahical model for a sentence of length *s*

Here $Y_i \in \{B, I, O\}, t_i \in \mathrm{T}$ are hidden label and observed feature vector on the i^{th} position, T stands for the set of allowable features.

In accordance with the model the sequence labelling task is formulated as finding the sequence $(\mathrm{Y}1, \ldots, \mathrm{Y}s)$, minimizing the distribution $P(Y_1^s \mid t_1^s)$. According to Hammersley-Clifford teorem such distribution may be factorized into function of arguments corresponding to the vertices of the graph specifying the CRF:

$$P(Y_1^s \mid t_1^s) = \frac{\prod_{i=1}^{s} \psi(Y_i, Y_{i+1}, t_i)}{Z} \tag{1}$$

where $Z = \sum_{Y_1^s} \prod_{i=1}^{s} \psi(Y_i, Y_{i+1}, t_i)$ is a normalizing factor or *partition function*.

$\psi(Y_i, Y_{i+1}, t_i)$ is a potential function of the graph clique calculated as:

$$\psi(Y_i, Y_{i+1}, t_i) = \exp \sum_{j=1}^{K} w_j f_j(Y_i, Y_{i+1}, t_i) \tag{2}$$

Here $f_j(Y_i, Y_{i+1}, t_i)$ is a j^{th} feature of the clique (Y_i, Y_{i+1}, t_i); K is a number of the features, $w_1 \ldots w_k$ are *feature weights* (model parameters). To find the model maximizing $P(Y_1^s \mid t_1^s)$ we maximize the sum:

$$\arg\max_{Y_1^s} \sum_{i=1}^{s} \sum_{j=1}^{K} w_j f_j(Y_i, Y_{i+1}, t_i) \tag{3}$$

To maximize (3) *Viterbi algorithm* can be used. The learning procedure is discussed in detail in [6].

3. Temporal Expressions Extraction

3.1. Task Definition

The task of temporal expressions extraction is a kind of named entity recognition task common in NLP. It is also common to normalize temporal expressions after the extraction procedure.

A temporal expression (also *time expression*, or *timex*) is a sequence of tokens (words, numbers and characters) that can denote a point in time, duration or frequency. The concept of temporal expression is not strictly defined and indistinct. Nevertheless the ISO standard "*TimeML*" for labeling and normalization of temporal expressions was developed and adopted [10].

These are some examples of temporal expressions:

Что будут показывать <TIMEX>сегодня ночью</TIMEX> по пятому каналу? /What will be on TV tonight?

<TIMEX>8 сентября 2013 года</TIMEX> состоялись выборы на пост мэра Москвы. / Mayoral elections were held on the 8th of September 2013.

<TIMEX>Через 2 недели</TIMEX> состоится встреча с руководителем. / The meeting with the chief will be held in two weeks.

Какую телепередачу показывают <TIMEX>ежедневно в 7 часов вечера</TIMEX>? / What program is on TV every day at 7 p.m.

It should to be noticed that words like "*мгновенно*" ("*instantaneously*") or "*быстро*" ("*quickly*") are not temporal expressions.

There are two main approaches to solving the problem of temporal expressions extraction. The first one is a rule-based approach. The main idea of this approach is searching in sentence for predefined patterns of temporal expressions [9,12]. This approach requires developing a list of patterns of timexes composed by a linguist and the resulting list is usually dependent on the domain of texts [12]. The second approach is based on machine learning [9]. In the context of statistical approach linguistic knowledge is not necessary, but large amount of labeled data is needed for training a statistical model.

3.2. Labeling Scheme and Generation of Features

The "*TimeML*" specification suggests a XML-like markup for time expressions. But XML-like scheme is verbose and redundant for simple extraction of temporal expressions. For this reason we took simple and widely-used BIO (Begin Inside Out) and IO (Inside Out) labeling alphabets.

Now we describe how to reduce task of temporal expression extraction to task of sequence labeling.

1. If i^{th} token of input sequence is the first token of temporal expression, then the i^{th} output label is *B*.
2. If i^{th} token of input sequence is included in temporal expression and is not the first token of it, then then the i^{th} output label is *I*.
3. The output label is "O", otherwise.

The set of valid features of tokens *T* contains five types of features.

1. All possible analyses of the token retrieved from the dictionary of OpenCorpora (non disambiguated) [12]. Examples of such kind of features: "noun", "verb", "dative case", "perfective aspect", etc.

2. Features based on spelling of the token: “token starts with capital letter”, “token has a digits”, etc.
3. Features describing position of token in the sentence: “token is the first token in sentence”, “token is the last token in sentence”
4. Features indicated that token is a specific for temporal expression word, i. e. trigger-word: “token is a month name”, “token is a name of day of week”, etc.
5. Previous groups features of nearest tokens.

3.3. Dataset

Currently there is no dataset with labeled temporal expressions for Russian. For this reason a small set of Russian phrases was labeled by hands in BIO labeling scheme. This small dataset contained 2000 sentences with roughly 500 temporal expressions and was used like test dataset.

Nevertheless, training machine learning algorithms require much bigger dataset. So, we used semi automatic procedure based on regular expressions for obtaining training data. Moreover, we developed rule-based base-line algorithm with help of this regular expressions.

3.4. Feature Selection

Set of valid features T has a high dimension. So, it is reasonable to apply feature selection methods. We used algorithm “Random Forest” [4] as an algorithm for feature selection. This algorithm is related to stochastic logic methods of machine learning. But “Random Forest” is also applicable to classification problem. So we used it like alternative method of temporal expressions extraction. Advantages of “Random Forest” are high generalization ability, application to data with high dimension and ability to deal with binary features.

3.5. Experiments

Subset of dataset from project OpenCorpora was taken for training and testing. This subset was labeled automatically with base-line algorithm based on patterns. Then subset was split into train (380,000 phrases, 5,000 temporal expressions) and test parts (40,000 phrases, 9,000 temporal expressions). Moreover, the subset of sentences labeled manually was used for testing (2,000 phrases, 500 expressions).

We trained two alternative algorithms: “Random Forest” and CRF. Outputs of algorithms were converted from BIO scheme to IO. Then standard quality measures were calculated (Recall R, Precision P and F_1-measure):

- $P = \dfrac{tp}{tp + fn}$

- $R = \frac{tp}{tp + fp}$
- $F_1 = \frac{2PR}{P + R}$

The results of base-line algorithm, CRF and "Random forest" are listed in the table below. It should be noticed that these results were obtained on the best set of features, i. e. on features which were selected with feature selection procedure "Random Forest".

Table 1. Experimental results. Time expressions extraction

Algorithm	*P*	*R*	F_1
Base-line	95,7	85,7	90,4
RF	96,4	87,1	91,5
CRF	96,3	89,9	93,05

4. Shallow Parsing

4.1. Task Definition

The problem of shallow parsing was formultated in [1]. The shallow parser searches in text the so-called base NPs which are the fragments of noun phrases excluding recursive parts. The common example of base-NP in English is a noun with its left adjuncts:

green colorless thoughts,
USA President Barack Obama.

To adapt this approch to Russian we include in base NP agreed adjectives, numerals and nouns (*Президент Ельцин*), and dependent base NPs in genitive case. Thus, according to these criteria the following phrases may be considered as base NPs:

любимый руководитель Ким Чен Ир/beloved leader Kim Jong Il

*друг отца жены инженера Лаборатории эффективных алгоритмов/*a friend of father of the Algorithm Lab engineer's wife

and so on.

It should be noticed that base NPs (or *chunks*) do not have clear linguistic interpretation. Nevertheless, it was proposed in [1] that speakers tend to make pauses on the borders of base NPs. Thus base NP appears to be a psycholinguistic entity and correlate with the term *elementary discourse item* appearing in some linguistic theories.

4.2. Solution Outline

To reduce shallow parsing to the sequence labelling problem we took the following approach:

1. If the i^{th} input token is the beginning of a base NP then the i^{th} output label is B.
2. If the i^{th} input token is insid a base NP then the i^{th} output label is I.
3. Otherwise, the output label is O.

To provide the opportunity of detection of heads of the phrase we augmented the standard BIO-alphabet with two labels: BH (token is the beginning of the base NP and is the head) and IH (token is the phrase head). The input sequence of the shallow parser was the output of the morphological analyzer. The features were: part of speech, gender, number, case (if defined), upper/lowercase. We also used features of neighboring tokens and there combinations.

4.3. Dataset

The training set was generated from the syntactically annotated corpus SynTagRus IITP RAS [8]. Every syntactic tree corpus was traversed and subtrees with noun roots were detected. Then the following edges were excluded: 1) the edges coming into tokens different from nouns, adjectives, numerals or adverbs or to nouns in case different from genitive and non-agreed with the head; 2) the edges coming to non-neighboring tokens from the base NP. Based on these criteria *BIO-BH-IH* was generated for the training and test set. The training set comprised 40976 and 6310 were chosen for the test set.

4.4. Results

We trained two models: the first one was based only on the grammatical features and the second one also used input token (wordform) as a feature. The results are given in tables 2 and 3.

Table 2. Experimental results. Base NPs

Model	P	R	F_1
SynTagRus. Tokens+	93,39	93,07	93,23
SynTagRus. Tokens-	94,52	94,27	94,39
Pereira (2003)	n/a	n/a	94.38

Table 3. Experimental results. Heads detection

Model	P	R	F_1
SynTagRus. Tokens+	95,56	95,14	95,55
SynTagRus. Tokens-	96,48	95,45	95,96

The tables demonstrate the method can effectively detect both base NPs and the corresponding heads. For comparison we also give the results achieved by Pereira for English.

5. Noun Phrases Inflection

5.1. Task Definition

The problem of inflection of noun phrases i.e. changing its case from nominative to any oblique case generally can be solved by means of special phrase inflection rules. This approach however is time consuming and error prone, so it is reasonable to make use of machine learning approach. In fact, the task reduces to the search of the head and tokens agreed with it. Then, all found target tokens are set to the proper morphological form.

5.2. Solution Outline

Using the simplest possible label inventory consisting of “1” and “0” is sufficient in this case. We output label “1” of the token should be set in a target form and “0” otherwise. Composing the input sequence we considered only tokens with morphological features of nominals (e.g. noun, adjective, numeral, participle). Neighboring non-nominal tokens were used as features. Thus, each input token had the following features: part of speech, gender, number, case and features of neighboring tokens both included and non-included in the input sequence.

For example, for the NP *медведь из леса* (a bear from forest), token *леса* would have following features: *noun, masculine, genitive, preposition_in_posistion-1, noun_in_position_-2*. We also used combination of the features.

5.3. Dataset

Corpus of noun phrases was generated from syntactic trees of SynTagRus corpus. We had to use information about edge types in the tree to generate cleaner training set. Each generated phrase was set to nominative case. We used noun phrases of the length no more than 10 tokens. Unfortunately, although we used edge labels the corpus still contained many erroneous entries. We manually cleaned a corpus of 100,000 tokens. 10,000 form them were selected for the training set.

5.4. Results

We present the results of experiments on the search of targets of inflection:

Table 4. Experimental results. Inflection targets

P	R	F_1
99,44	99,74	99,59

Although we anticipated good results the algorithm happened to perform better than expected and the total time spent on the development less than the estimated time on the grammar preparation.

6. Conclusions

Solution of three natural language processing tasks including shallow parsing, temporal expressions extraction and noun phrase inflection have been proposed. We have shown that all these tasks can be reduced to sequence labeling problems and solved by means of linear-chain CRF statistical model. It allows replacing complex linguistic rules with a set of relevant features and data preparation. This work is often less time consuming and error prone.

References

1. *Abney S.* (1991), Parsing by chunks, Principle-based Parsing, Kluwer Academic Publishers, pp. 257–279.
2. *Antonova A. Ju., Solovyev A. N.* (2013), Conditional random field models for the processing of Russian [Ispol'zovanie metoda uslovnyh sluchajnyh polej dlja obrabotki tekstov na russkom jazyke], Computational Linguistics and Intellectual Technologies: Proceedings of the International Conference "Dialog 2013" [Komp'iuternaia Lingvistika i Intellektual'nye Tehnologii: Trudy Mezhdunarodnoj Konferentsii "Dialog 2013"], Bekasovo, pp. 39–52.
3. *Bottou L.* (1991), A theoretical approach of connectionist learning: Applications to the recognition of speech [Une approche theorique de l'apprentissage connexionniste: Applications a la reconnaissance de la parole], Doctoral dissertation [Doctoral dissertation], University of Paris XI.
4. *Breiman L.* (2001), Random forests, Machine Learning, Vol. 45, no. 1, pp. 5–32.
5. *Culotta A., Wick M., Hall R.* (2007), First-Order Probabilistic Models for Coreference Resolution, In Proceedings of HLT/NAACL, pp. 81–88.
6. *Granovskij D. V., Bocharov V. V., Bichineva S. V.* (2010), Open corpora: principles of work and prospects [Otkrytyj korpus: printsypy raboty i perspektivy], In proceddings of XIII Russian Conference "Internet and society today" [Internet i sovremennoe obshchestvo: Trudy 18 Vserossijskoj ob"edinennoj konferentsii], St. Petersburg, pp. 94–99.
7. *Lafferty J., McCallum A., Pereira F.* (2001), Conditional random fields: Probabilistic models for segmenting and labeling sequence data, Proceedings of the 18th International Conference on Machine Learning, Williamstown, Massachusetts, pp. 282–289.

8. *Nivre J., Boguslavsky M., Iomdin L.* (2008), Parsing the SynTagRus treebank of Russian, Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008), pp. 641–648.
9. *Poveda J., Surdeanu M., Turmo J.* (2007), A comparison of statistical and rule-induction learners for automatic tagging of time expressions in english, In Proceedings of the 14th International Symposium on Temporal Representation and Reasoning (TIME 2007), IEEE, pp. 141–149.
10. *Pustejovsky J., Ingria R., Sauri R.* et al. (2003), Timeml: Robust specification of event and temporal expressions in text, In Fifth International Workshop on Computational Semantics (IWCS-5).
11. *Ramshaw L. A., Marcus M. P.* (1995), Text chunking using transformation-based learning, The Third Workshop on Very Large Corpora, pp. 82–94.
12. *Reeves R. M., Ong F. R., Matheny M. E.* (2013), Detecting temporal expressions in medical narratives, I. J. Medical Informatics, Vol. 82, no. 2, pp. 118–127.
13. *Sha F., Pereira F.* (2003), Shallow parsing with conditional random fields, In Proceedings of HLT/NAACL, pp. 213–220.
14. *Sutton C., McCallum A.* (2006), An Introduction to Conditional Random Fields for Relational Learning, MIT Press.