

# Hidden Markov Models with Affix Based Observation in the Field of Syntactic Analysis

Marcin Pietras<sup>(✉)</sup>

Computer Science and Information Technology,  
West Pomeranian University of Technology, Żołnierska 49, Szczecin, Poland  
mpietras@wi.zut.edu.pl

**Abstract.** This paper introduces Hidden Markov Models with N-gram observation based on words bound morphemes (affixes) used in natural language text processing focusing on the field of syntactic classification. In general, presented curtailment of the consecutive gram's affixes, decreases the accuracy in observation, but reveals statistically significant dependencies. Hence, considerably smaller size of the training data set is required. Therefore, the impact of affix observation on the knowledge generalization and associated with this improved word mapping is also described. The focal point of this paper is the evaluation of the HMM in the field of syntactic analysis for English and Polish language based on Penn and Składnica treebank. In total, a 10 HMM differing in the structure of observation has been compared. The experimental results show the advantages of particular configuration.

**Keywords:** Hidden Markov Models · N-gram · Syntactic analysis · Natural Language Processing · Treebank · Part-Of-Speech · Data clustering

## 1 Introduction

The mathematically rich Hidden Markov Models (HMM) are widely used in natural language processing, especially in Part-Of-Speech (POS) tagging. In recent years, various HMMs designated for this purpose has been presented [1–3]. In order to solve NLP challenges, the researchers introduced N-grams analysis [4] with the assumption that text or language structure (such as grammar or syntax) can be recognized by using occurrence probabilities for particular words (i.e. unigram - single word) or sequences of words (i.e. N-gram - sequence of N consecutive words) in the text. However, with increasing complexity of the N-gram observations (associated with length of words sequence), much more training data has to be provided to the model in order to obtain statistically significant learning results [5]. Nonetheless, in the most cases training data will still be insufficient to cover all possible sequences of words, which implies that the model may misinterpret the meaning of the observed text. Furthermore, certain expressions (in database) may be used only occasionally and be heavily dependent on the current geo-language trends and situation. Therefore, the HMM based system should also support an observation's uncertainty handling. The study presented in [6, 7] addressed the problem of new instances classifications which were not included in the training corpus. The database deficiency to some extent can be overcome by the

smoothing methods presented well in [8]. In addition, in the [9] are shown the advantages of morphological features extraction in order to handle unknown information. In contrary to the whole words N-gram based observation the study [10] introduces more advanced method that is based on morphological analysis and is dividing the word into morphemes (morpheme-word representation). However, in this paper presented concept put the main focus on the bound morphemes features detection, which is an intermediary approach between morphological analysis and dictionary words mapping. Therefore, the observation model includes mainly the word affixes, with the naive assumption of its constant length. This assumption leads to creation of the observation window (both for Prefix and Suffix of the word), which can be extended (to give a more accurate observation) or reduced (which leads to a statistical generalization). Applying this modification allows to adjust the observation window individually for every gram of observation. This approach was evaluated in the field of syntactic tagging which involves text classification and disambiguation depending on the characteristics of the recognized expression and its surroundings [11]. Performance of the HMM for text syntactic analysis is strongly dependent on the observation complexity and training database. As previously mentioned, database that would cover all words combination is an enormous challenge. The HMM creation and training based on limited treebank [12, 13] database enforce an alternative (to the dictionary word mapping methods) approaches in order to increase proper text syntactic analysis. Another issue related with observations and model structure complexity is significantly larger resources utilization. Consequently, usage of model at some rate of complexity would simply be impractical for many applications because of too time-consuming computation effort. As a one of possible solution a special “affix oriented” observation is introduce in order to reduce model complexity (by strongly decreasing the emission matrix) and to improve unknown expression recognition.

## 2 Methodology

This paper does not assume any restrictions concerning the Markov Models. All states are by default fully connected and the transitions probabilities are established by the evaluation of training database. The transitions between states are carried out at regular discrete intervals. Process of transition between states is defined by probabilities of states occurrence obtained from training database. In general, the transitions probabilities may depend on the whole process so far. For the first order HMM states probabilities are reduced to the transition from a previous state only. Formally, HMM is described with the same notation as in [14].

The information related to the model structure, the observation type as well as the textual data preparation and the HMM training process for each HMM presented in this paper are described in details.

In order to create and train HMMs served “HMM-Toolbox” application developed specifically for this project. The HMM-Toolbox is equipped with, among others, the GUI to allow manual states tagging and HMM parameters modifications. The core algorithms associated with HMM computation (such as Viterbi path, Forward-Backward, Baum-Welch and HMM factory) are provided by Jahmm [15] library.

## 2.1 Treebank Database

The treebank as a parsed and tagged text corpus that annotates syntactic sentence structure provides ready to use database for the purpose of HMM training and further verification. For this reason all states in HMM are convergent with tags occurred in treebank database and moreover states path and observations sequence are also extracted from it. Two treebanks were utilized: Penn Treebank for English language and Składnica treebank for Polish language. In both treebanks lower level syntactic tags (Parts-Of-Speech) and higher level syntactic classes (phrase/dependency classes) can be distinguishing. Hence, two types of HMMs are designed for the each treebank.

## 3 Complexity of M-order Model with N-gram Observation

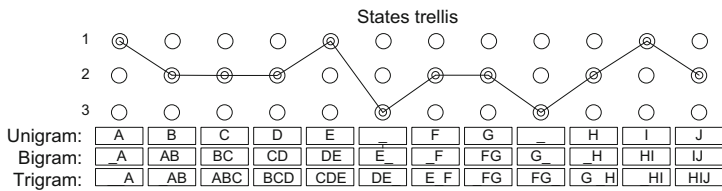
The independence assumption states that the output observation at time  $t$  is dependent only on the current state and is conditionally independent of previous observations. A number of studies show that this assumption becomes a significant deficiency of the HMM [16]. Nonetheless, the model could be improved in terms of Expectation Maximization for used data by applying observation extension. This technique is well known in computational linguistic and refers to N-gram modification [17]. The N-gram observation is created by combining N past observations together and shifting them in queue with length equal to N.

Let us consider a HMM ( $St = 3$ ,  $Obs = 11$ ) with a state sequence equal: *122213223212*, where (1-vowel, 2-consonant, 3-pause) and corresponding observation sequence equal: *ABCDE\_FG\_HIJ*. Then the bigram (2-gram) observation will look: *\_A, AB, BC, CD, DE, E\_, \_F, FG, G\_, \_H, HI, IJ*. Analogically the trigram (3-gram) observation will look: *\_\_A, \_AB, ABC, BCD, CDE, DE\_, E\_F, \_FG, FG\_, G\_H, \_HI, HIJ*. Figure 1 visualizes states path and corresponding observations for first order HMM. Certainly, by N-gram operation the observation is enhanced by occurrence of past N expressions. Still, state sequence did not change. Nevertheless, N-gram conversion affects number of observation and observation probability distribution, thereby to estimate HMM parameters at least  $O \cdot N$  more training data is required to obtain similar statistical validity in comparison to basic HMM. Similar operation can be performed in terms of states sequence. This technique is well known and refers to M-order Markov model [18]. The M-order chain can be created by combining M past states together. Let us consider the second order HMM ( $St = 9$ ,  $Obs = 11$ ), then to create a first order Markov chain each state will include combination with other state to allow precedent state occurrence memory. So basics first order states (1-vowel, 2-consonant, 3-pause) will be transformed in second order model by full states combination: *11, 12, 13, 21, 22, 23, 31, 32, 33*, (where e.g. state *12* correspond to vowel-consonant; state *22* correspond to consonant-consonant). Regardless to the order of Markov model the observation sequence stay the same:

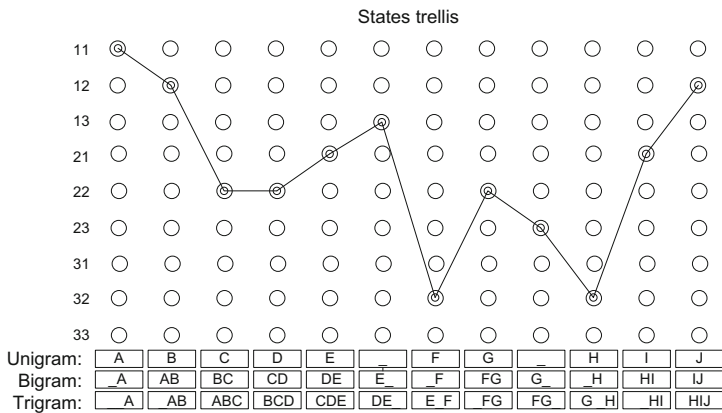
*11-A, 12-B, 22- C, 22- D, 21- E, 13-\_, 32- G, 23- \_, 32- H, 21-I, 12-J.*

Analogically, for N-gram derivate, e.g. trigram observations:

*11-\_\_A, 12-\_AB, 22-ABC, 22-BCD, 21-CDE, 13-DE\_, 32-E\_G, 22-\_FG, 23- FG\_, 32-G\_H, 21-\_HI, 12-HIJ.*



**Fig. 1.** First order HMM states path and corresponding observation sequence.



**Fig. 2.** Second order HMM states path and corresponding observation sequence.

Figure 2 visualizes states path and corresponding observations sequence for second order HMM.

However, higher-order Markov models are able to model only fixed length dependencies and in practice are strong limited by exponential growth in number of states with every next order. This example shows that to cover all states occurrence in  $M$  order HMM at least  $S \cdot M$  more training data is required.

Combination of both  $M$ -order HMM with  $N$ -gram observation results in  $(O \cdot N) \cdot (S \cdot M)$  more training data requirements. Both approaches modify the HMM model in terms of maximizing the probability of the data and from this point of view greater model complexity and higher demanding on training data possibly pays off with better HMM fitting to the data.

## 4 HMM in the Field of Syntactic Classification

The process of determining a syntactic class for each word in the analyzed text can be achieved by using HMM, where the discreet observation represent the word while the state of the Markov Model will represent the corresponding part of speech/sentence. In order to estimate all parameters correctly, it is crucial to determine what kind of features are represent by the observation. In presented models it is assumed that the word's

affixes are more important for syntactic analysis than a stem. Hence, every word is represented only by its Prefix and Suffix, while the stem can be omitted. For the simplicity, both Prefix and Suffix has a fixed number of characters. Consequently, for short words, the values are redundantly overlapped (e.g. with - > “witith”), while for longer words middle part of the word is cut out (e.g. approachable - > “appble”). The main advantage of this approach is a simplified mechanism for inclusion of new words that were not involved in the training process. A fixed number of characters in both Suffix and Prefix results a limited number of possible combinations. Hence, the vast majority of syntactic classes can be covered even at smaller training database. In other words, affix oriented expression groups more words to the same observation (e.g. “accordable” = “accruable” or “adminstrable” = “admirable”). Words that have different, unknown stem will be categorized by their Prefix and Suffix learned from training database.

#### 4.1 N-gram Affix Oriented Observation

A basic observation is built from word’s Prefix and Suffix. The N-gram observation is created by combining N past observations together and shifting them in queue with length equal to N. However, affix oriented observations can be degraded by removing letters from the inner side of the word. Correspondingly, the total number of possible combinations significantly decreases, revealing only important statistic relations. Simple sentence case study example in Table 1 shows the main idea about affix N-gram observation and its derivations.

Because of progressive clarity reduction for past elements in the gram sequence the presented example, is limited to use up to 3 grams in the observation. The progressive character of the N-gram observation modification is here utilized to limit the information carried by the past grams. However, observation window can as well have (for all or for some selected grams) a fixed length. At this point three different affix observation

**Table 1.** Applied affix modification. Uni-, Bi- and Tri-gram observations extracted from example sentence: “those who cannot remember the past are condemned to repeat it”.

Raw	Unigram	Bigram	Trigram
those	tho-ose	___-___: tho-ose	___-___: ___-___: tho-ose
who	who-who	th_-_se: who-who	___-___: th_-_se: who-who
cannot	can-not	wh_-_ho: can-not	t_-_e: wh_-_ho: can-not
remember	rem-ber	ca_-_ot: rem-ber	w_-_o: ca_-_ot: rem-ber
the	the-the	re_-_er: the-the	c_-_t: re_-_er: the-the
past	pas-ast	th_-_he: pas-ast	r_-_r: th_-_he: pas-ast
are	are-are	pa_-_st: are-are	t_-_e: pa_-_st: are-are
condemned	con-ned	ar_-_re: con-ned	p_-_t: ar_-_re: con-ned
to	to_-_to	co_-_ed: to_-_to	a_-_e co_-_ed: to_-_to
repeat	rep-eat	to_-_to: rep-eat	c_-_d: to_-_to: rep-eat
it	it_-_it	re_-_at: it_-_it	t_-_o: re_-_at: it_-_it

structures for N-gram observation can be distinguished. Absolute progressive: every subsequent gram is degraded more than its predecessor until the loss of whole information. Offset progressive: every subsequent gram is degraded more than its predecessor until reach some level of information which is mandatory. Constant affix: here the level of observation clarity is adjusted for all grams.

## 4.2 Model Observation Evaluation

When using an affixes based observation a question about the accuracy of word mapping arises. If length of affixes is large enough all words are directly mapped, which means that even very similar words are distinguished from each other. However, if the length is too small, too many different words will be grouped together and the

**Table 2.** Dictionary coverage regarding to different Prefix-Suffix characters length.

Nr	Prefix	Suffix	Observations for English	English dictionary coverage [%]	Observations for Polish	Polish dictionary coverage [%]
1	1	1	685	1.07	967	1,03
2	1	2	5191	8.12	5879	6,29
3	1	3	18152	28.40	18952	20,29
4	1	4	35291	55.22	36264	38,82
5	1	5	48180	75.39	54232	58,06
6	2	1	4518	7.06	6681	7,15
7	2	2	17710	27.71	20469	21,91
8	2	3	36086	56.46	39447	42,23
9	2	4	49322	77.17	56025	59,98
10	2	5	56689	88.70	70834	75,84
11	3	1	20479	32.04	25185	26,96
12	3	2	38939	60.93	46657	49,95
13	3	3	51251	80.19	62991	67,44
14	3	4	57689	90.27	74076	79,31
15	3	5	61065	95.55	83169	89,04
16	4	1	40348	63.13	49335	52,82
17	4	2	52309	81.85	68175	72,99
18	4	3	58367	91.33	78218	83,74
19	4	4	61268	95.87	84347	90,31
20	4	5	62773	98.22	89550	95,88
21	5	1	52345	81.90	67430	72,19
22	5	2	58654	91.78	80607	86,30
23	5	3	61535	96.28	86370	92,47
24	5	4	62784	98.24	89962	96,32
25	5	5	63423	99.24	92882	99,44
All words:			63906	100	93396	100

accuracy of syntactic recognition will suffer or even go to zero. Table 2 presents dictionary coverage regarding to different Prefix-Suffix character length [19, 20].

As it is shown in Table 2 the length of the affixes affect word representation and word grouping. For Prefix and Suffix with the length of 2 characters each, the clarity of word representation is about 28 % (17710 different affixes represents 63906 words), while for affixes with single character each gives 1 % of clarity of word representation (685 different affixes represents 63906 words). Similar situation is for Polish language.

For completely new expression (misspelled or not included in the training database) some additional precautions method may be applied. Primarily by detecting unknown expression a Levenshtein distance [21] can be measured in order to find closest observation that approximately match this expression. For N-gram observation the Levenshtein distance should be calculated for each word observation in N-word sequence and each obtained value should be weighted with multiplicative inverse of N. Depends on the analyzed language the Levenshtein distance can be modified in order to mitigate Suffix or Prefix impact as it is shows in [22]. If the expression does not fit to any representation then such expression is classified as unknown (\_\_\_\_).

### 4.3 HMM Preparation

Presented process of preparation assumes that the HMM is based on affix observations which are statistically significant so that HMM will be able to determine the membership of the word to the one of the syntactic classes. At HMM initialization a number of states is established based on syntactic tags found in treebank database.

The database for Polish language (Składnica treebank) consist more than twenty thousand sentences with more than 200 thousand syntactically tagged words. The database for English language (based on Penn treebank) consist more than ten thousand sentences with more than 200 thousand syntactically tagged words. The set of all possible affix observations for given language is obtained from the proper treebank database. However, the initialization for emissions probabilities is based on Wordnet database. All counters related to observation occurrence are initialized with the value corresponding to the membership level of a given affix to the given Part-Of-Speech (Noun, Verb, Adjective or Adverb) in Wordnet. For example, counter for affix “re-al” will be initialized with values (4-strong, 1-weak): 4 for Noun class, 3 for Adjective class, 2 for Verb class and 1 for Adverb class. To overcome data sparseness problem and to improve probabilities estimation for unseen observations the additional Laplace smoothing is also applied.

## 5 Experiments and Results

Classification of words syntactic category has been made by calculating the Viterbi path. Correctness of syntactic class recognition (expressed in percentage) is represent by average accuracy of all HMM states. For affix observation structure two digits are assigned, the first determines the number of letters for the Prefix and the second digit determines the number of letters for the Suffix. The verification was conducted on test

database. A part of unknown expression occurred in test database is also pointed out. Table 3 lists all 10 HMMs included in the experiment.

The HMM for Polish Dependency Types [23] classification and HMM for English Phrase Chunk classification are hierarchical HMMs [24] based on word's affix observation and on POS class recognized previous by HMM for Part-Of-Speech classification for a given language.

## 6 Discussion

Results presented in Table 3 for unigram HMMs conforms that by using affix oriented observation HMM performs well. Even for single letters affixes more than 75 % of class were recognize correctly. The differences in the observation number may reach decimal of percents. This is due to combinatorial limitations of the model where a smaller number of letters is subject of observation. The biggest advantage of HMMs with simple and small set of observations is the computation time. For many applications (where computation latency/time is essential) complex HMM compute to slow and despite the higher accuracy more useful is a smaller model even with some classification deficiency.

**Table 3.** Classification results for affix based HMM in the field of syntactic analysis.

HMM description	Number of states/Observations	Size of Prefix-Suffix	Accuracy [%]	Unknown [%]
Polish Part-Of-Speech	40/836	1-1	75,89	0.07
Polish Part-Of-Speech	40/4791	1-2	83.69	0.67
Polish Part-Of-Speech	40/14703	2-2	88.40	2.87
Polish Part-Of-Speech	40/36196	3-3	93.63	10.31
Polish Dependency Types	28/11579	2-2 +POS state	82.54	7.34
English Part-Of-Speech	47/767	1-1	80.02	0.13
English Part-Of-Speech	47/3994	1-2	90.20	0.93
English Part-Of-Speech	47/9055	2-2	91.20	2.69
English Part-Of-Speech	47/15546	3-3	95.26	5.48
English Phrase Chunk	24/12915	2-2 +POS state	82.70	3.89



By affix orientation modification the knowledge generalization takes place at the observation level. Nonetheless, the clarity of the observation should be chosen carefully. Insufficient level of accuracy may lead to the loss of its statistical significance and the observation becomes useless.

Obtained results for unigram HMMs are very promising especially for Polish language. The accuracy of Part-Of-Speech classification is comparable with the results presented in [25]. Although, the Dependencies Types classification is much more difficult task still unigram HMM were able to achieve more than 80 % of classification correctness.

Nevertheless, classification results presented in this paper refers only to unigram HMMs. Hence, the further research will concentrate on examination of HMMs based on N-gram affix oriented observations. Presumably, N-gram variant should significantly improve the classification accuracy.

## 7 Conclusions

In the paper affix oriented observations was introduce and comprehensively described. The analysis was performed to examine the correlation between clarity of word representation (affix size) and HMM classification accuracy. In addition, a general example for M-order HMM model N-grams was presented for concept better understanding and to point out further application of affix oriented observation in order to decrease model complexity.

Evaluated Hidden Markov Models based and trained on Penn Treebank (for English) and on Składnica Treebank (for Polish) database prove that HMMs are suited to the language processing in the field of syntactic tagging even by limited clarity of observations. The number of unknown words has significantly decreased for HMMs that less accurate observation. Furthermore, accuracy of recognizing the syntactic class remained at a similar level in comparison to models with exact observation. Depending on the requirements of the programs which utilize a syntactic analysis, different HMMs derivatives can be useful. If very high accuracy is required, a complex N-gram HMM with additional support for unknown words recognition should be applied. However, if processing speed is a priority and accuracy plays a secondary role then HMM based on simplified observation (single character affixes) will be more appropriate.

## References

1. Kupiec, J.: Robust part-of-speech tagging using a hidden Markov model. In: Computer Speech and Language, pp. 225–242 (1992)
2. Goldwater, S., Griffiths, T.: A fully Bayesian approach to unsupervised part-of-speech tagging. In: Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics, Prague, Czech Republic, pp. 744–751. Association for Computational Linguistics, June 2007
3. Gao, J., Johnson, M.: A comparison of Bayesian estimators for unsupervised hidden Markov model pos taggers. In: Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing, pp. 344–352 (2008)

4. Lioma, C.: Part of speech n-grams for information retrieval. Ph.D. thesis, University of Glasgow (2008)
5. Brants, T.: TnT — A statistical part of speech tagger. In: Proceedings of the 6th Applied NLP Conference (ANLP-2000), pp. 224–231 (2000)
6. Thede, S.M.: Predicting part-of-speech information about unknown words using statistical methods. In: Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics - v.2, pp. 1505–1507 (1998)
7. Nakagawa, T., Kudoh, T., Matsumoto, Y.: Unknown word guessing and part-of-speech tagging using support vector machines. In: Proceedings of the Sixth Natural Language Processing Pacific Rim Symposium, pp. 325–331 (2001)
8. Jurafsky, D., Martin, J.H.: Speech and Language Processing. Prentice Hall, Upper Saddle River (2000)
9. Tseng, H., Jurafsky, D., Manning, C.: Morphological features help POS tagging of unknown words across language varieties. In: Proceedings of the Fourth SIGHAN Bakeoff (2005)
10. Luong, M.T., Nakov, P., Ken, M.Y.: A hybrid morpheme-word representation for machine translation of morphologically rich languages. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP 2010), Cambridge, MA, pp. 148–157 (2010)
11. Adler, M.: Hebrew morphological disambiguation: an unsupervised stochastic word-based approach. Ph.D. thesis, Ben-Gurion University of the Negev, Israel (2007)
12. Taylor, A., Marcus, M., Santorini, B.: The Penn Treebank: An Overview (2003)
13. Hajnicz, E.: Lexico-semantic annotation of składnica treebank by means of PLWN lexical units. In: Proceedings of the Seventh Global WordNet Conference, Tartu, Estonia, pp. 23–31 (2014)
14. Rabiner, L.R.: A tutorial on hidden Markov models and selected applications in speech recognition. *Proc. IEEE* **77**(2), 257–286 (1989)
15. Jahmm, Java implementation of HMM related algorithms (2009)
16. Layton, M.: Augmented Statistical Models for Classifying Sequence Data (2006)
17. Langkilde, I., Knight, K.: The practical value of n-grams in generation. In: Proceedings of the Ninth International Workshop on Natural Language Generation, Niagara-on-the-Lake, Ontario, pp. 248–255 (1998)
18. Lee, L.-M., Lee, J.-C.: A study on high-order hidden Markov models and applications to speech recognition. In: Ali, M., Dapigny, R. (eds.) IEA/AIE 2006. LNCS (LNAI), vol. 4031, pp. 682–690. Springer, Heidelberg (2006)
19. Fellbaum, C.: WordNet: An Electronic Lexical Database. MIT Press, Cambridge (1998)
20. Maziarz, M., Piasecki, M., Szpakowicz, S.: Approaching plWordNet 2.0. In: Proceedings of the 6th Global Wordnet Conference, Matsue, Japan (2012)
21. Levenshtein, A.: Binary codes capable of correcting deletions, insertions and reversals. *Sov. Phys. Dokl.* **10**(8), 707–710 (1966)
22. Pietras, M.: Sentence sentiment classification using fuzzy word matching combined with fuzzy sentiment classifier. *Electrical Review - Special issue, Poland* (2014). doi:[10.15199/48.2015.02.26](https://doi.org/10.15199/48.2015.02.26)
23. Wróblewska, A.: Polish dependency parser trained on an automatically induced dependency bank. Ph.D. dissertation, Institute of Computer Science, Polish Academy of Sciences, Warsaw (2014)
24. Fine, S., Singer, Y., Tishby, N.: The hierarchical hidden Markov model: analysis and applications. *Mach. Learn.* **32**, 41–62 (1998)
25. Kobyliński, L.: PoliTa: a multitagger for Polish. In: Proceedings of the Ninth International Conference on Language Resources and Evaluation, Iceland, pp. 2949–2954 (2014)

Hard and Soft Computing for Artificial Intelligence,  
Multimedia and Security

Kobayashi, S.-y.; Piegat, A.; Pejas, J.; El Fray, I.;  
Kacprzyk, J. (Eds.)

2017, XXV, 368 p. 151 illus., Softcover

ISBN: 978-3-319-48428-0