Using book dialogs to extract emotions from texts in Polish

Paweł Skórzewski

Adam Mickiewicz University
Umultowska 87, 61-614 Poznań, Poland
pawel.skorzewski@amu.edu.pl

Abstract

Detection of emotions from text can be very challenging, especially if no annotated corpus is given. We propose to use book dialog lines and accompanying phrases to obtain utterances annotated with emotion vectors. We describe two different methods of achieving this goal. Then we use neural networks to train models that assign a vector representing emotions for each utterance. These solutions do not need any corpus of texts annotated explicitly with emotions, because information about emotions for training data is extracted from dialogs' reporting clauses. We compare the performance of both solutions with other emotion detection algorithms.

1. Introduction

1.1. Sentiment analysis and emotion detection

Sentiment analysis can be described as the use of methods of computational linguistics to identify, extract and analyze subjective information. Emotion detection deals with extracting and analyzing emotions from data and can be viewed as a sub-field of sentiment analysis. The notion of sentiment analysis is often used in a narrower sense: as the analysis of sentiment polarity of words and sentences (positive vs. negative vs. neutral).

1.2. Related work

The importance of both sentiment analysis and emotion detection continuously increases because of their application in market analysis and opinion mining (Strapparava and Mihalcea, 2007).

Most of the research in the domain of sentiment analysis focuses on classifying texts as positive or negative (Pang et al., 2008) (De Bruyne et al., 2018). Emotion detection and analysis is rather new research field in natural language processing, but the study of emotions has a long history in psychology and sociology (De Bruyne et al., 2018). Most popular models of emotions are where an emotional state is considered as a combination of some basic emotions. Plutchik distinguished 8 basic emotions (Plutchik, 1980), while Ekman distinguished 6 basic emotions (Ekman, 1992).

Traditional emotion classification consists of assigning one or more of emotion categories to a given text fragment (an utterance, a sentence or a document). Systems for emotion classification have been developed so far for different kinds of text, like children fairy tales (Alm et al., 2005), newspaper headlines (Strapparava and Mihalcea, 2007), poems (Auracher et al., 2010) or blog posts (Aman and Szpakowicz, 2007) (Gill et al., 2008). There are also studies on extracting the intensity of emotions from text (Mohammad and Bravo-Marquez, 2017). Sometimes detection of emotions in texts is assisted by analyzing emotions in speech, which allows to use additional non-lexical features (Metze et al., 2010).

However, most of these studies are conducted for the English language. There are several studies on detecting emotions in texts for other languages, including French (emotion lexicon – Abdaoui et al., 2017), Spanish (emotion detection in tweets – Gil et al., 2013), Chinese (emotion detection in tweets – Yuan and Purver, 2015) or Japanese (text-based affect analysis – Ptaszynski et al., 2017). In this paper, we focus on Polish, which is not well studied in the field of emotion analysis, although presented methods are language-independent and could be used for other languages too, provided that adequate text corpora are available.

1.3. Emotion models: the wheel and the hourglass

Robert Plutchik in his "wheel of emotions" theory distinguished 8 basic emotions, arranged in 4 pairs of opposite emotions: *joy—sadness*, *trust—disgust*, *fear—anger*, *surprise—anticipation* (Plutchik, 1980). He claims that every emotion can be viewed as a combination of these 8 basic emotions, e.g. *love* = *joy* + *trust* or *pessimism* = *sadness* + *anticipation*. Moreover, all emotions can occur in varying degrees of intensity, e.g. *annoyance* is weak *anger* and *terror* is strong *fear*.

The hourglass of emotions is an extension of this model (Cambria et al., 2011). It uses a 4-dimensional space to represent emotions, with 4 dimensions equivalent to Plutchik's basic emotion pairs (see Table 1).

dimension	-1	$\longleftrightarrow$	$+1$
pleasantness	sadness	$\longleftrightarrow$	joy
attention	surprise	$\longleftrightarrow$	anticipation
sensitivity	fear	$\longleftrightarrow$	anger
aptitude	disgust	$\longleftrightarrow$	trust

Table 1: Four dimensions of the hourglass model of emotions.

In the hourglass model, every emotion can be represented as a vector $(P, At, S, Ap) \in [-1, 1]^4$, where P stands for *pleasantness*, At – *attention*, S – *sensitivity*, and Ap – *aptitude*. Such a vector is called a *sentic vector* by the authors. We will call it also simply an *emotion vector*.

The hourglass model allows us to represent not only basic emotions (e.g. *joy* is represented as a vector $(0.5, 0, 0, 0)$) and derivative emotions (e.g. *anxiety* = *anticipation* + *fear* is represented as $(0, 0.5, -0.5, 0)$) but also varying degrees of emotion intensity (e.g. *rage* = strong *anger* as $(0, 0, 1, 0)$) and the entire spectrum of mixed emotions (e.g. $(-0.9, 0.5, 0.1, -0.4)$). Conversely, every vector $(P, At, S, Ap) \in [-1, 1]^4$ can be seen as representing some emotional state.

1.4. Challenges in analysis of sentiment and emotions

Many methods of emotion and sentiment analysis are lexicon-based (Yuan and Purver, 2015). There are various possible ways of combining individual sentiment (or emotions) to obtain the overall sentiment of an utterance (Kim and Hovy, 2004).

However, one must keep in mind that the emotional tone of a text is mostly not a simple combination of emotions related to particular words (Mohammad, 2016). Some expressions, though of neutral sentiment itself, have a significant impact on the sentiment of a whole utterance (e.g. negation or modal verbs). The same word can carry different sentiment or emotions depending on the context. In addition, there are such phenomena as sarcasm, irony, humor, metaphors or colloquial expressions. Sometimes a sentiment of an utterance can be ambiguous even for a human recipient. There are research studies trying to deal with the aforementioned challenges, e.g. recognizing and processing humor (Dybala et al., 2017) or detecting irony (Reyes et al., 2012).

On the other hand, the important challenge for emotion classification of texts with machine learning is the lack of reliable training data (Alm et al., 2005). While finding annotated data for classic sentiment analysis (positive/negative) is relatively easy (there are many online review systems that involve some rating or scoring opinions), texts annotated explicitly with emotions are virtually impossible to find on the wild. For some types of texts, like e.g. Twitter or Facebook posts, creating an annotated corpora of sentences and emotions can be made using emoticons (Pak and Paroubek, 2010) (Yuan and Purver, 2015) or hashtags (Mohammad, 2012).

It can be implied from these examples that in order to create a useful training corpus indirectly, we should find texts containing “metadata” that can be used as an indication of emotion. Dialogs in books can be seen as a natural example of such texts with “metadata” because they are often accompanied by reporting clauses and other descriptions of the manner of speaking. In this paper, we propose a solution that uses dialogs and accompanying phrases as a source of training data to create a model of emotions in texts.

2. System description

2.1. System overview

Our system takes an utterance on the input and returns a sentic vector representing an emotional state of the input utterance. To make our system extensible and to facilitate testing and comparing different approaches, our system has a modular structure. The system can be parameterized with which algorithm should it use to calculate the emotion vector, as well as which function should it use to aggregate emotions of individual words and word meanings (not used in some of the available algorithms). The input utterance is first pre-processed, and then a chosen classifier with selected parameters is applied to obtain the sentic vector. The pre-processing stage involves tokenization, lemmatization and POS-tagging. The overview of the system is presented in the Figure 1.

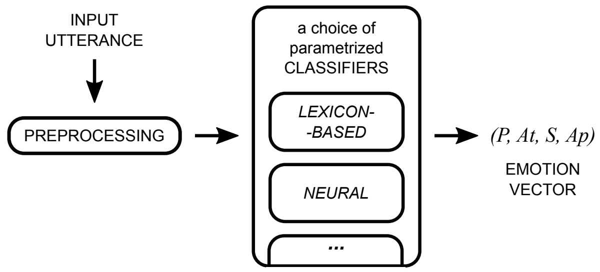

Figure 1: System overview.

2.2. Emotion representation

For assigning emotional values to particular words, we use the plWordNet-emo sentiment lexicon for Polish (Zaśko-Zielińska et al., 2015). It contains comprehensive emotion and sentiment annotations for 30 000 lexical units. For each lexical unit, understood as a triple: lemma + semantic variant + part of speech, a set of labels taken from the set of Plutchik’s basic emotions is assigned. Sometimes there are no emotion labels assigned to a given lexical unit. In other words, zero, one or more basic emotions can be assigned to each lexical unit.

To represent emotions we use a 4-dimensional space model based on Cambria’s hourglass model of emotions. We assign a 4-dimensional sentic vector (P, At, S, Ap) to each of Plutchik’s basic emotion in the way shown in Table 2.

emotion	vector	emotion	vector
joy	$(1, 0, 0, 0)$	sadness	$(-1, 0, 0, 0)$
anticipation	$(0, 1, 0, 0)$	surprise	$(0, -1, 0, 0)$
anger	$(0, 0, 1, 0)$	fear	$(0, 0, -1, 0)$
trust	$(0, 0, 0, 1)$	disgust	$(0, 0, 0, -1)$

Table 2: Representation of basic emotions as vectors.

This representation not only allows us to represent every basic and derivative emotion as a 4-dimensional vector, but also works the other way round: every vector $(P, At, S, Ap) \in [-1, 1]^4$ can be seen as a representation of some emotion, e.g. $(-0.9, 0.1, 0.5, -0.1)$ is a representation of an emotional state dominated by intense sadness and some anger.

Our system can assign an emotion vector to any input word appearing on its own or in the context of a sentence. In the first case, if no information about part-of-speech is given, all lexical units from the plWordNet-emo corpus with the same lemma as a given word are taken into account. Then, for each of these lexical units, an emotion vector is calculated as the aggregate of the sentic vectors of the corresponding basic emotion labels, using the aggregation function provided as a system parameter. Subsequently, the eventual emotion vector is calculated as the aggregate of all emotion vectors for the corresponding lexical units. If the part of speech of the input word is known, only lexical units of the same part of speech are taken into account.

2.3. Lexicon-based classifiers

For reference, we prepared a classifier that is based on a simple aggregation of emotions related to particular words of a given utterance. This classifier takes a tokenized, lemmatized and POS-tagged utterance as the input. For every word (token: lemma and POS-tag) of the utterance, the corresponding sentic vector is calculated in the way described in the Subsection 2.2. Then, the aggregate of all words' sentic vectors is calculated using the selected aggregation function, and this aggregate is returned as the utterance's sentic vector. In our research we used two aggregation functions:

- arithmetic mean,
- maximum with regard to absolute value (i.e. the function that chooses the number whose absolute value is the greatest).

2.4. Neural models

We tested a few different architectures of neural networks to create the best model. The configurations differ in the number, types and dimensionality of layers, the values of dropout and recurrent dropout parameters (Gal and Ghahramani, 2016), and the number of training epochs.

The first layer is always an embedding layer that converts input utterances to vectors. The last layer is always a dense layer with 4-dimensional output and sigmoid activation function. Between them, there are some middle layers: Long-Short Term Memory (LSTM) layers (Hochreiter and Schmidhuber, 1997) and dense layers with hyperbolic tangent activation function. We used Adam optimizer (Kingma and Ba, 2015) and cosine proximity loss function. This general architecture is shown schematically in Figure 2.

The results of evaluations of different architecture variants are described in the Section 3.

We had no corpus annotated explicitly with emotions or sentic vectors to be our training data, so we prepared the training corpus the following way.

As a source for data to create a training corpus we used a part of the corpus created for the purposes of the paper (Kubis, 2019). Kubis's corpus was created on the basis of the transcripts of books from the online service Wolne Lektury[1], which collects the texts of books in Polish that belong to the public domain. It contains 1.37 million utterances (23 million tokens). Despite the fact that the source books were written in the 19th and 20th century, the corpus's language is modern Polish, because most of the books' texts were contemporized before publication, and the remaining texts have been pre-processed with a diachronic normalizer. We used this corpus because it had

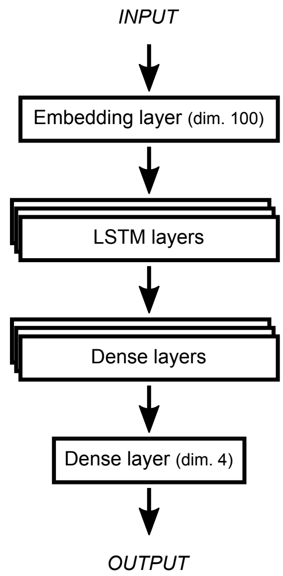

Figure 2: General architecture of the neural network.

been already pre-processed and annotated with useful semantic features, especially related to dialogs.

Kubis's corpus contains texts that have been split into paragraphs, tokenized and lemmatized. Each token (lemma) is tagged with part of speech and morphological information. There are also annotations about which parts of text contain named entities, which parts are dialog lines, which parts indicate speaker and which parts indicate the manner of speaking. Those last annotations were especially helpful for us to create a training corpus annotated with emotions.

For example, the corpus contains the following utterance: – *Wszystko jak najlepiej – wykrzyknął wesoło lekarz.* (' "Everything is fine", the doctor exclaimed cheerfully.'). The particular phrases and words of this utterance are annotated as follows:

- the phrase *wszystko jak najlepiej* ('everything is fine') is annotated as "dialog line",
- the phrase *wykrzyknął wesoło lekarz* ('the doctor exclaimed cheerfully') is annotated as "reporting clause",
- the phrase *wesoło* ('cheerfully') is annotated as "manner of speaking".

The idea is to use emotions extracted from the manner of speaking ('cheerfully'), or from the whole reporting clause ('the doctor exclaimed cheerfully'), to describe the emotional state related to the dialog line ('everything is fine').

We used two different methods to obtain a reliable sentic vector for training utterances, both of them based on the properties of dialogs in books. Therefore, to create a training dataset, we selected only utterances that were dialog lines in the books.

In the first method ("reporting clauses model"), we used dialogs and corresponding reporting clauses to create a training dataset. In each dialog line where a reporting

[1] https://wolnelektury.pl

clause was present, we took the proper dialog line as the input utterance, and the reporting clause as the basis for calculating sentic vector. The sentic vectors were calculated in the way described in Subsection 2.3. but using the reporting clause as the input. This way we obtained 72 198 training examples.

In the second method ("manners model"), we used dialogs and corresponding expressions describing the manner of speaking to create a training dataset. In each dialog line where a manner of speaking was identifiable, we took the proper dialog line as the input utterance, and the expression indicating the manner of speaking as the basis for calculating sentic vector. The sentic vectors were calculated in the way described in Subsection 2.3. but using the expression indicating the manner of speaking as the input. This way we obtained 11 033 training examples.

Each of these two methods resulted in obtaining a corpus of utterance—sentic vector pairs, that was used in turn as training data for the neural network. Note that the number of obtained training examples was significantly lower than the size of the original Kubis's corpus due to the fact that only a fraction of original utterances were dialog lines and met the required conditions.

In both cases we trained the neural network described above for several epochs, obtaining a model for predicting an emotion vector for a given input utterance.

3. Experiments and results

For testing purposes, we prepared an annotated corpus of 598 utterances and corresponding sentic vectors. The corpus was created by means of manual annotation. Each utterance has been annotated by at least four independent annotators. The annotators were volunteer students interested in machine learning. Each annotator was presented an utterance and had to point one or more of 8 Plutchik's basic emotions, or choose "neutral". Each basic emotions chosen by the annotator was converted to a sentic vector in the way shown in Table 2; "neutral" answer was converted to the sentic vector $(0, 0, 0, 0)$. All sentic vectors corresponding to these annotations have been averaged to one sentic vector, to represent the complete emotional state the annotator associates with this utterance. Then, all the sentic vectors corresponding to annotations for a given utterance have been averaged. This way we obtain 894 pairs of utterances and corresponding emotion vectors.

We evaluated different variants of both neural models ("manners model" and "reporting clauses model") and, for comparison, two lexicon-based models (with arithmetic mean and with maximum as the aggregation function), as described in Section 2. Additionally, we compared our results to the "zero baseline" model that returned zero vectors $(0.0, 0.0, 0.0, 0.0)$ for all input utterances.

We used two metrics for evaluation purposes:

- root-mean-square error (RMSE), defined as:

$$\mathrm{RMSE} = \sqrt{\frac{1}{m}\sum_{i=1}^{m}\left|\left|v^{(i)} - v_{\mathrm{ref}}^{(i)}\right|\right|^2},$$

- mean cosine distance (MCosD), defined as:

$$\mathrm{MCosD} = \frac{1}{m}\sum_{i=1}^{m}\left(1 - \frac{v^{(i)} \cdot v_{\mathrm{ref}}^{(i)}}{\left|\left|v^{(i)}\right|\right| \cdot \left|\left|v_{\mathrm{ref}}^{(i)}\right|\right|}\right),$$

where m is the total number of utterances in the testing corpus, $v^{(i)}$ is the obtained sentic vector for ith utterance, $v_{\mathrm{ref}}^{(i)}$ is the reference sentic vector for ith utterance.

The obtained RMSE and MCosD for selected models are shown in Tables 3 and 4.

4. Conclusion

Different neural models performed very similar in terms of mean cosine distance. RMSE values were more diverse, but still outperformed by simple averaging sentic vectors for particular words in the utterance.

The fact that models trained on dialogs' reporting clauses performed not as well as simple averaging emotion vectors of particular words may indicate that the emotional state of the utterance is more closely related to the words it consists of than to the words it is described by. It may also suggest that authors tend to describe manners of speaking explicitly only if it cannot be inferred from a dialog line itself.

5. Future work

There is still room for improvement of our system's performance, at least in three directions. First, collecting more training data (dialogs with reporting clauses and words determining the manner of speaking) should improve the performance of the neural models. Secondly, we can find ways to use other meta-information about utterances that could be more useful to determine the emotions related to these utterances. Last, we plan to investigate other neural network architectures that may be more suitable to learn emotion vectors.

6. References

Abdaoui, Amine, Jérôme Azé, Sandra Bringay, and Pascal Poncelet, 2017. FEEL: a French expanded emotion lexicon. *Language Resources and Evaluation*, 51(3):833–855.

Alm, Cecilia Ovesdotter, Dan Roth, and Richard Sproat, 2005. Emotions from text: machine learning for text-based emotion prediction. In *Proceedings of the Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*. Vancouver: Association for Computational Linguistics.

Aman, Saima and Stan Szpakowicz, 2007. Identifying expressions of emotion in text. In *International Conference on Text, Speech and Dialogue (TSD)*. Springer.

Auracher, Jan, Sabine Albers, Yuhui Zhai, Gulnara Gareeva, and Tetyana Stavniychuk, 2010. P is for happiness, N is for sadness: universals in sound iconicity to detect emotions in poetry. *Discourse Processes*, 48(1):1–25.

Cambria, Erik, Andrew Livingstone, and Amir Hussain, 2011. The hourglass of emotions. In *Cognitive Behavioural Systems - COST 2102 International Training*

model	middle layers configuration	RMSE	MCosD
reporting clauses	no middle layers	1.000	0.971
reporting clauses	dense (64)	1.479	0.981
reporting clauses	LSTM (64)	1.590	0.969
reporting clauses	LSTM (32) + dense (32)	1.460	**0.968**
reporting clauses	LSTM (64) + dense (64)	1.490	0.969
reporting clauses	LSTM (100) + dense (100)	1.525	0.969
manners	no middle layers	**0.627**	0.977
manners	dense (64)	0.809	0.973
manners	LSTM (64)	1.388	0.969
manners	LSTM (32) + dense (32)	1.247	0.969
manners	LSTM (64) + dense (64)	1.346	0.969
manners	LSTM (100) + dense (100)	1.294	0.969

Table 3: The evaluation results for selected neural models. The second column shows all layers except the first one (embedding layer) and the last one (dense layer with 4 units). The dimensions of middle layers are given in parentheses. All LSTM layers have dropout value of 0.5 and no recurrent dropout.

model	RMSE	MCosD
lexicon-based arithmetic mean	**0.492**	**0.880**
lexicon-based max. w. r. t. abs.	1.524	0.937
zero baseline	0.496	N/A

Table 4: The evaluation results for lexicon-based models.

School, Dresden, Germany, February 21-26, 2011, Revised Selected Papers. Springer, pages 144–157.

De Bruyne, Luna, Orphée De Clercq, and Véronique Hoste, 2018. LT3 at SemEval-2018 task 1: a classifier chain to detect emotions in tweets. In *Proceedings of The 12th International Workshop on Semantic Evaluation (SemEval-2018)*. Association for Computational Linguistics.

Dybala, Pawel, Motoki Yatsu, Michal Ptaszynski, Rafal Rzepka, and Kenji Araki, 2017. Towards joking, humor sense equipped and emotion aware conversational systems. In *Advances in Affective and Pleasurable Design*, volume 483 of *Advances in Intelligent Systems and Computing*. Springer, pages 657–669.

Ekman, Paul, 1992. An argument for basic emotions. *Cognition & Emotion*, 6(3-4):169–200.

Gal, Yarin and Zoubin Ghahramani, 2016. A theoretically grounded application of dropout in recurrent neural networks. In *Advances in Neural Information Processing Systems*.

Gil, Gonzalo Blázquez, Antonio Berlanga de Jesús, and José M Molina Lopéz, 2013. Combining machine learning techniques and natural language processing to infer emotions using Spanish Twitter corpus. In *International Conference on Practical Applications of Agents and Multi-Agent Systems*. Springer.

Gill, Alastair J, Robert M French, Darren Gergle, and Jon Oberlander, 2008. Identifying emotional characteristics from short blog texts. In *30th Annual Conference of the Cognitive Science Society*. Cognitive Science Society Washington, DC.

Hochreiter, Sepp and Jürgen Schmidhuber, 1997. Long short-term memory. *Neural Computation*, 9(8):1735–1780.

Kim, Soo-Min and Eduard Hovy, 2004. Determining the sentiment of opinions. In *Proceedings of the 20th International Conference on Computational Linguistics*. Association for Computational Linguistics.

Kingma, Diederik P and Jimmy Ba, 2015. Adam: A method for stochastic optimization. In *The International Conference on Learning Representations (ICLR)*. San Diego.

Kubis, Marek, 2019. Quantitative analysis of character networks in Polish XIX and XX century novels. In *Proceedings of Digital Humanities Conference 2019 (DH 2019)*.

Metze, Florian, Anton Batliner, Florian Eyben, Tim Polzehl, Björn Schuller, and Stefan Steidl, 2010. Emotion recognition using imperfect speech recognition. In *Eleventh Annual Conference of the International Speech Communication Association*.

Mohammad, Saif, 2012. #Emotional tweets. In **SEM 2012: The First Joint Conference on Lexical and Computational Semantics – Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012)*. Montréal, Canada: Association for Computational Linguistics.

Mohammad, Saif M., 2016. Sentiment analysis: Detecting valence, emotions, and other affectual states from text. In Herb Meiselman (ed.), *Emotion Measurement*. Elsevier, pages 201–237.

Mohammad, Saif M. and Felipe Bravo-Marquez, 2017. Emotion intensities in tweets. In *Proceedings of the Sixth Joint Conference on Lexical and Computational Semantics (*Sem)*. Vancouver, Canada.

Pak, Alexander and Patrick Paroubek, 2010. Twitter as a corpus for sentiment analysis and opinion mining. In *LREc*, volume 10.

Pang, Bo, Lillian Lee, et al., 2008. Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval*, 2(1–2):1–135.

Plutchik, Robert, 1980. A general psychoevolutionary theory of emotion. In *Theories of emotion*. Elsevier, pages 3–33.

Ptaszynski, Michal, Pawel Dybala, Rafal Rzepka, Kenji Araki, and Fumito Masui, 2017. ML-Ask: Open source affect analysis software for textual input in Japanese. *Journal of Open Research Software*, 5(1):1–16.

Reyes, Antonio, Paolo Rosso, and Davide Buscaldi, 2012. From humor recognition to irony detection: The figurative language of social media. *Data & Knowledge Engineering*, 74:1–12.

Strapparava, Carlo and Rada Mihalcea, 2007. SemEval-2007 task 14: Affective text. In *Proceedings of the 4th International Workshop on Semantic Evaluations (SemEval-2007)*. Prague.

Yuan, Zheng and Matthew Purver, 2015. Predicting emotion labels for Chinese microblog texts. In M. Gaber et al. (ed.), *Advances in Social Media Analysis*. Springer, pages 129–149.

Zaśko-Zielińska, Monika, Maciej Piasecki, and Stan Szpakowicz, 2015. A large wordnet-based sentiment lexicon for Polish. In Galia Angelova, Kalina Bontcheva, and Ruslan Mitkov (eds.), *International Conference Recent Advances in Natural Language Processing. Proceedings*. Hissar, Bulgaria.