New Issues and Solutions in Computer-aided Design of MCTI and Distractors Selection for Bulgarian

Ivelina Nikolova
Institute for Parallel Processing, Bulgarian Academy of Sciences
25A Acad. G. Bonchev Str.
1113 Sofia
iva@lml.bas.bg

Abstract

We describe a methodology for improving the generation of multiple-choice test items through the usage of language technologies. We apply common natural language processing techniques, like constituency parsing and automatic term extraction together with additional morpho-syntactic rules on raw instructional material in order to determine its key terms. These key terms are then used for the creation of fill-in-the blank test items and the selection of distractors. Our work aims at proving the availability and compatibility of language resources and technologies for Bulgarian, as well as at assessing the readiness for implementation of these techniques in real-world applications.

Keywords

information extraction, natural language processing application in e-learning

1 Introduction

Multiple-choice tests (MCT) are a common tool to assess learners achievements. They are widely proven to be efficient. During the last years MCT gained even more popularity due to the growth of the e-learning programmes. In these programmes, which are offered by universities and other educational institutions, multiple-choice questions appear to be the most frequently used evaluation tool. Multiple-choice is a form of assessment in which respondents are asked to select the best possible answer(s) out of a list of choices. We refer to the questions as *stems*, the best option as *correct answer* and the rest of the given choices as *distractors*. The demand for great quantities of such tests and the availability of already advanced learning technologies gave rise to a new research area dealing with the generation of multiple-choice test items (MCTI) and the suggestion of distractors from raw text.

The manual preparation of MCT is a time and effort consuming task. Teaching experts who prepare the tests have much broader knowledge in their field in general, compared to the specific content which is explicitly included in the particular instructional material. They have to tune the tests carefully to the knowledge of the test takers. Hence one of the most difficult subtasks during the creation of test items is to decide whether a question does really have its answer in the taught material. With an automatic extraction of test items from the instructional material, this problem is easily solved and the time for test designing is significantly reduced. An automatic extraction allows the test designers to oversee large instructional materials in a new manner, giving them a content overview and helping them to take faster decisions about the topics to be included in a test and concrete questions which could be given to the learners.

The generation of multiple-choice questions with the help of natural language processing (NLP) technologies is an active research area in which different tools for text processing are used in order to transform the facts from the instructional materials to questions for students assessment. The items produced in this way are often used in Computer Assisted Language Learning (CALL), for vocabulary [2], grammar [3, 4, 1] or language proficiency testing [11, 5], as well as in comprehension testing in specific subject areas in the native language [6]. Our aim is to produce multiple-choice test items for testing learners achievements especially in the second area - learners comprehension of specified instructional material.

We present the design of a workbench for test designers employing language technologies for generation of MCTI (stem, correct answer and distractors), which are to be wrapped as learning objects (LO) and can be loaded in an e-learning environment. The task is divided into three subtasks: *automatic keyterm extraction*; *sentence extraction and stem transformation* and *distractors selection*. In particular, we discuss our contributions to an improved methodology for keyterm and distractors selection and stem transformation.

The remainder of the article is organised as follows: Section 2 describes the state-of-the-art; Section 3 reveals the motivation of the author; Section 4 outlines the overall architecture of the workbench; Section 5 presents a detailed view of the text processing phases; Section 6 presents a discussion on tests done with the system and Section 7 gives a conclusion and issues for future work.

2 Related Work

One of the first works on our topic was presented by [3]. Fairon implemented a corpus search for finding sentences or short parts of text that match initially

Multilingual Resources, Technologies and Evaluation for Central and Eastern European Languages 2009, Bulgaria, pages 40–46

preselected linguistic patterns. Later, [5] proposed a word sense disambiguation method for locating sentences in which designated words carry specific senses, and applied a collocation-based method for selecting distractors that are necessary for multiple-choice cloze items. Our work differs from these approaches as far as we detect relevant terms automatically, henceforth called *keyterms*. Furthermore, for distractors selection we employ morpho-syntactic information.

Authors working on vocabulary testing [2] use definitions or examples given for the focal term in WordNet in order to produce a non-interrogative stem. [6] also employ WordNet, but only as a tool for distractors selection. Their approach is domain independent; furthermore, the authors report a 6-10 times speed-up in comparison with a manual test elicitation. Similarly, [11] uses a thesaurus in order to find distractors for stem, generated by replacing the verb of the chosen sentence with a blank. [4] apply standard classification methods in order to decide the position in the gap in the generation of fill-in-the-blank (FIB) test items. Other researchers who are actively working in the area include [1], who are focusing on the different types of question models with application mainly in the language learning. In our approach we extract sentences which contain the central terms for the given material in Bulgarian and produce FIB type of questions out of them. Along with that, we also suggest the correct answer and distractors.

3 Motivation

The fact that we are not familiar with any related work for learning materials in Bulgarian (except for previous work of the author [8]), together with the presence of sophisticated language technologies for Bulgarian, which allow for complex text analysis strongly inspired us to work out the practical potential of our ideas. Moreover, the growing interest in the field, which is due to its significant practical importance, was a motivating factor to concretise our aims and more precisely to apply the developed technology for e-learning purposes.

4 Workbench Outline

The system is designed in a way that it accepts instructional material from the test designer in form of raw text and produces draft learning objects - MCTI of FIB type with their correct answer and possible distractors.

Our approach is based on the assumption that the learner knowledge is tested over the terms, central to the learning materials. As shown in Fig. 1, once the text is submitted, a list of generated FIB questions, concerning keyterms from the instructional material, is presented to the test designer. At this moment, she can modify all MCTI components and then save or export them as a learning object or as a plain text document.

The list of FIB stems serves as a cross-reference to the whole text and facilitates for the test-designer in summarising the learning topics.

5 Data Processing

This section describes the processing of the data from the user input to the output of the draft learning objects. Well-established language technologies, like parsing and automatic term extraction are employed. Additionally, linguistic assumptions are taken into consideration. An overview of the data processing chain is shown in Fig. 2.

The input of the test designer is plain text instructional material, which has to be parsed in order to extract lexico-syntactic features from the text. Due to the importance of parsing as a basic source of information used later on for the test items generation, we have picked a statistical parser which reports state-of-the-art results and has been tuned to work with Bulgarian - the Berkeley parser [9, 10]. The parser was trained on BulTreeBank[1]. Parsing texts from the same domain as the training corpus gave highly satisfactory results.

After reformatting the parsed text, we extract from it all nouns and noun phrase structures as well as names. From the tools offering fast structure querying for our purposes the most appropriate turned out to be the CLaRK system[2]. As it is based on Xpath expression querying it is fully configurable. In contrast with the NP extractor *Morena* we used earlier, CLaRK allows for the manual specification of sequences of constituents.

In order to overcome the language inflection, the extracted morpho-syntactic structures are stemmed using BulStem [7] and organised in an internal representation format, where each stem[3] maps to all NPs having the same stem. Here is an example of a partial record for the stem закон (law):

*[stem value="*закон*" occurences=51*
type="N" isKeyterm=false]
*[instance value="*закон*"]*
*[instance value="*законите*"]*
*[instance value="*закона*"]*
.....

In this case, the stem is закон (law). It has a total of 51 occurences in the document. Some of them are *закон* (law), *законите* (the laws), закона (the law) and it is type *noun*. Other NP types are *np-A-N* - NP composed of adjective and noun, *np-N-PP* - NP composed of noun and prepositional phrase, *NE-loc* - name of the type location, *NE-org* - name of the type organisation, *NE-Pers* - name of the type person, *NE-other* - name of the type other). For each wordform (phrase) corresponding to the stem, only one instance is generated. If the wordform appears at least twice, only the counter *occurences* is incremented, but no new instance is created. The attribute *isKeyterm* is initially set to *false* for all stems. After the keyterm threshold is set, it is turned to *true* for the terms which belong to the keyterms list. This representation is the starting

[1] A HPSG-based Syntactic Treebank of Bulgarian (BulTreeBank), `http://bultreebank.org/`.
[2] CLaRK - an XML Based System for Corpora Development, `http://bultreebank.org/clark/index.html`.
[3] A stem is the common prefix of several wordforms.

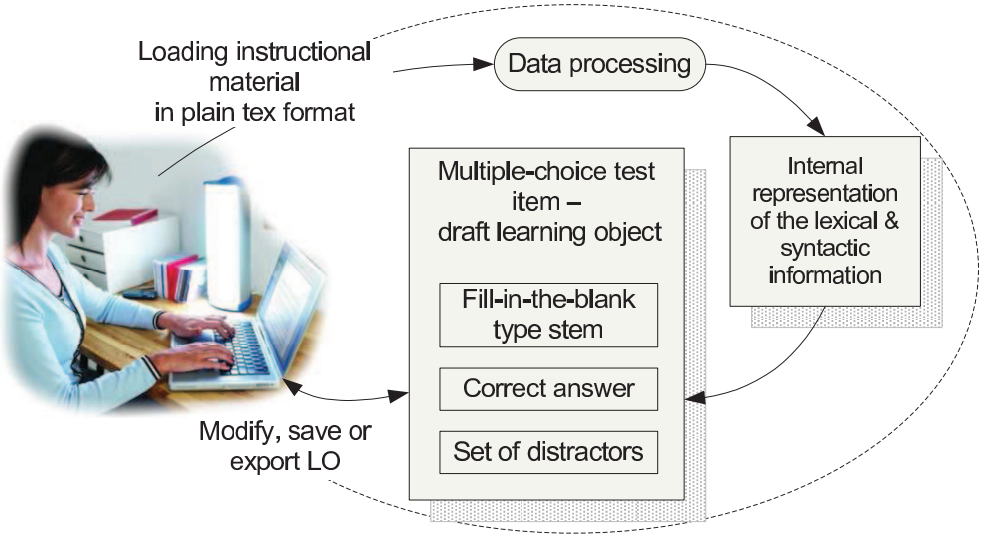

Fig. 1: *Workbench supporting the development of multiple-choice test items.*

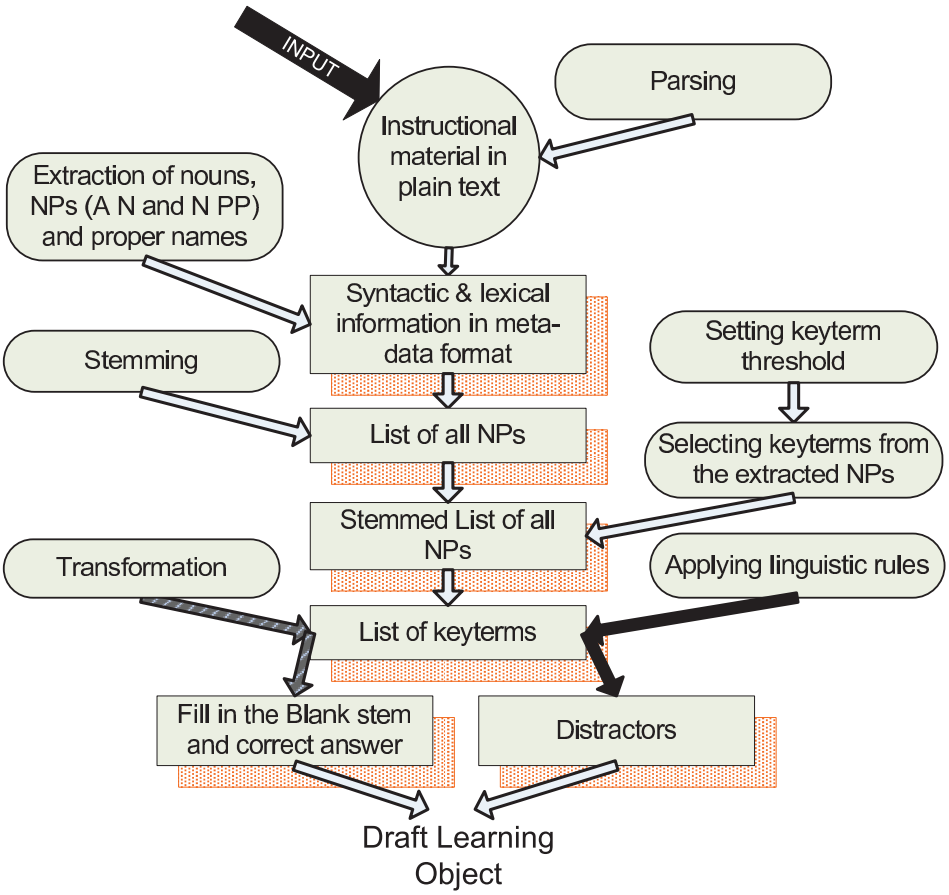

Fig. 2: *Data processing.*

point for any further processing in order to generate the test item stem and distractors.

The *occurences* attribute of the stem field is used later on for calculating the threshold for important terms. The instances are used in order to expand the stem and query the text for sentences containing the exact keyterms. In the extracted sentences, we replace the keyterm with a blank and offer it as an FIB item together with the keyterm as its correct answer. Then, applying some linguistic rules on the keyterm yields a set of distractor suggestions.

5.1 Keyterms Extraction

In order to extract the terms which are central for the instructional material and to filter out the less important ones we establish some requirements a keyterm should adhere to:

– keyterms are nouns and noun phrases from the text, which frequency is higher than a set threshold;

– keyterms are the noun phrases, which contain a keyterm with frequency higher than the set threshold;

– all names are keyterms.

The first step in this respect is to extract all potential keyterms. In previous research, we have concentrated on extracting nouns, noun phrases of the type $np - A - N$ and names, but now, in order to extend the list of valuable keyterms, we have inserted an additional type of noun phrases: $np - N - PP$. In domains like Law, where the specific terms tend to be longer, exactly this structure greatly helps in detecting keyterms. After all potential keyterms are extracted they are stemmed and the two lists of terms – the stemmed and the original one – are arranged in the internal representation shown above.

As we have determined in previous research and is reported also by other authors [6], in instructional materials the keyterms are often repeated in order to make the learner remember them. That is why simple term frequency is a better measure than TF-IDF, which tends to lower the score of the most often used words. We store the frequencies of our terms (f_{ti}) in the *occurences* attributes for each stem. We sort these frequencies and calculate the number of words having equal frequency ($r_{f_{ti}}$). Then we set the threshold as follows:

$$threshold = max f_{ti} - 2, \{r_{f_{ti}} \geq f_{ti}\} \quad (1)$$

We have established this procedure for threshold setting empirically, by observing manually prepared test items and analysing the keyterms used in questions and answers.

Once the threshold is set, we initialise the list of keyterms by adding to it all nouns or noun phrases[4] that have frequency higher than the *threshold*. In the second step, we add to the keyterm list all noun phrases that contain a keyterm, without regarding their occurrence frequency. In a third step, we add all names as keyterms. For example: Along with the stem of the noun право (Right/Law) all these noun phrase stems will be added to the list of keyterms:

право на жалби (right of complaint)
право на живот (right of living)
право на законодател инициатив (right of legislation initiative)
международ прав (International Law) etc.

All NPs containing stop words are removed. In our case stop words most often appear to be personal and possessive pronouns.

The belonging of each stem to the keyterm list is implemented by turning the value of the attribute *isKeyterm* to *true* in the internal representation shown above.

5.2 Stem Extraction

We aim to produce a MCTI, which is of FIB type, and along with it, to suggest a correct answer and possible distractors. Seen from this perspective, our task may be thought of as vocabulary testing where optional answers are available. Taking into account the constraints we have put on the extraction of keyterms, we decided to relax the syntactic restrictions about the position of the keyterm, resp. the blank. As only requirement in this respect, we set the extraction of well-formed sentences. In terms of the grammar we use, these are sentences wrapped in a VPS constituent[5].

We extract all sentences from the text which contain at least one of the keyterms. For each of the keyterms in a sentence, we check whether it is a part of a longer keyterm:

– if it is not, we replace it with blank and save the so-produced stem;

– if it is contained in a longer keyterm, then we replace with blank the longest keyterm it is a part of and then save the stem.

Consider the following example. A sentence containing a keyterm *право* (law) is:

Външната политика на Република България се осъществява в съответ-ствие с принципите и нормите на международното *право*.

(The foreign policy of Republic of Bulgaria is realised in conformity with the International Law.)

The longest sequence of words containing the keyterm *право* (right/Law) and being a keyterm is *международното право* (International Law). That is why the system catches *международното право* and replaces it with a blank and produces the stem:

Външната политика на Република България се осъществява в съответ-ствие с принципите и нормите на ..... .

Respectively in the following sentence:

Чужденците и чуждестранните юридически лица не могат да придобиват право на собственост върху земя освен при наследяване по закон.

[4] All string comparisons are done in stemmed fashion.

[5] VPS -head-subject verb phrase for full definitions - HPSG-based Syntactic Treebank of Bulgarian (BulTreeBank), BulTreeBank Project Technical Report 05. 2004, http://bultreebank.org/TechRep/BTB-TR05.pdf

(The foreigners and foreign legal bodies cannot acquire land property rights except for the case of inheritance by law.)

the longest sequence containing the term *право* and being a keyterm is the phrase "*право на собственост върху земя*" (land property right). Hence the system will replace this keyterm with blank and will produce the following FIB stem (instead of considering for a keyterm *право* only).

Чужденците и чуждестранните юридически лица не могат да придобиват ..... освен при наследяване по закон.

5.3 Suggestion of Distractors

In well designed multiple-choice questions, the distractors are semantically close to the correct answer, as well as to each other in a sense. On the basis of our previous work and observations over manually prepared tests we suppose that distractors are close to each other if they look alike, too. Very often in tests for beginners, the distractors are noun phrases which contain the same noun as the one in the correct answer but with a different modifier or the other way round - the same modifier, but different noun. Our approach was based mainly on this assumption so far and now we want to extend the idea to the following:

– distractors of NPs of the type $np - N - PP$ are only NPs of the same type and the same head noun;

– distractors of NPs of the type $np - A - N$ are only NPs of the same type, which contain the same noun and different modifier or contain the same modifier and different noun;

– distractors of nouns are all NPs containing the given noun;

– distractors of names are names of the same type (for ex. for a keyterm which is a name of the type *Org* all names of the type *Org* are distractors).

The distractors are matched with the keyterms in a stemmed fashion too. Later on they are expanded to their full form and they are offered to the test-designer. Given the previously shown examples we may produce the following distractors:

Stem: Външната политика на Република България се осъществява в съответ-ствие с принципите и нормите на международното *право*. (The foreign policy of Republic of Bulgaria is realised in conformity with the International Law.)

Keyterm/correct answer: *международното право* (the International Law)

Type: np-A-N

Possible distractors: вътрешното право (the Domestic Law); избирателно право (franchise)

Consider the case when the keyword is a part of np-N-PP phrase:

Stem: Чужденците и чуждестранните юридически лица не могат да придобиват право на собственост върху земя освен при наследяване по закон. (The foreigners and foreign legal bodies cannot acquire land property rights except in case of inheritance by law.)

Keyterm/correct answer: *право на собственост върху земя* (land property right)

Type: np-N-PP

Possible distractors: *право на адвокат-ска защита* (right of advocate defense), *право на жалби* (right of complaint), *право на живот* (right of life), *право на законодателна инициатива* (right of legislation initiative), *право на лична свобода* (right of personal freedom), *право на ползване* (right to use), *право на строеж* (right of construction), *право на труд* (right to work).

When the keyterms are names, all names of the same type are distractors to each other. For example when processing the Bulgarian Constitution the names of the type NE-Loc are only *София* (Sofia) and *България* (Bulgaria) and they are treated always as options to each other. As Sofia is a city and Bulgaria a country they hardly assimilate each other and unfortunately we can not cope with this issue yet as we do not rely on any external resources which could deliver us this additional information. Consider the following problematic example where a question complying with this rule would look like.

Stem: Столицата на Република България е град София. (The capital of Republic of Bulgaria is Sofia.)

Keyterm/correct answer: *София* (Sofia)

Type: NE-Loc

Possible distractors: *България* (Bulgaria)

We want to make clear that this is not the general figure but only a specific case. Our observation is that with the additional rule for distractors selection of names, the performance of the system significantly increases. Of course the recognition of the names is a matter of good parsing, but as the parser model is trained on BulTreeBank, which contains annotation for named entities, we rely on comparatively high rate of recognition at least in the domains in which the parser was trained.

6 Testing and Evaluation

6.1 Assessment of Results Obtained from the Bulgarian Constitution

As a first try of evaluation of our system, we run it over an extracted part of the Bulgarian Constitution and asked several experts to evaluate the quality of the resulting MCTI and the features of the system. The Law domain is characterised with its relatively long terms and also the high frequency of the terms and importance of most of the sentences. The linguistic patterns we chose for extracting keyterms and distractors turned out to suit very well the keyterms in the

Law domain. This is apparent from the results shown in Fig 3. 75% of the input sentences were extracted as MCTI. The high number of selected FIB stems means that a high percentage of sentences in the text contain keyterms. This is explicable with the nature of the laws where the redundant information is reduced to a minimum.

The average number of suggested distractors is between 2 and 3. The number of distractors for the examined texts varies between 0 and 7 and often all of the distractors are good suggestions. For about one third of the resulting MCTI the system could not offer distractors, which is due to the fact that the distractors are selected only from the submitted instructional materials and no external sources are used.

The criteria for evaluating the results were the following:
Quality of the question (1-3):
1 - the question is not proper for testing learners on this material; 2 - the question is unclear; 3 - the question is a well formed sentence, concerning terms which are central for the instructional material.
Quality of the answer (1-3):
1 - the answer is not central for the instructional material; 2 - the answer is central for the instructional material but more specific or general than the desired answer; 3 - the answer is a central for the instructional material and concrete enough.

Fig. 3 shows the trends in average scores given by the experts when assessing the quality of the stems and correct answers (keyterms) of the selected questions (shown on the X-axis), according to the criteria given above (shown on the Y-axis). Both the stems and answers received most often the highest mark (3). Half of the evaluators have given the highest score (3) to all of the questions and the other half have given different marks maximum to the half of the sentences. This means that keyterms are correctly chosen by the algorithm and they have high importance for the material. Given this fact, we can explain the high score given to the stems by the fact that in Law the sentences structure is very compact. Almost each sentence represents a separate rule and they are often independent from each other. In this respect the legal texts differ significantly from texts in humanities like Geography and History where references are often used and terms are not that strictly defined. The high scores given for question quality could be also explained with the fact that the questions are directly extracted from the text, and thus their grammatical well-formness is preserved.

6.2 Discussion

Our aim with this work was to explore the field of automatic generation of MCTI and to prove the availability and compatibility of the language resources and technologies for Bulgarian as well as to assess the readiness for the implementation of these techniques in real-world applications. We were attracted by the high level of the work done in this area for English and we wanted to check whether it is possible to build a working prototype using some existing tools and to make inferences about the directions in which language technologies (LT) development for Bulgarian should take. Given the fact that the state-of-the-art in LT for English and for Bulgarian is incomparable, we wanted to point out concrete steps which must be taken in consideration to help the development of the next generation LT for Bulgarian.

From this experiment we can clearly point out several decisive factors, whose improvement will lead to more satisfactory results and overall progress in the area. First of all, as a fundamental basis of all further processing, improvements in parsing will result in more correct extraction of target morpho-syntactic structures. In the experiment we noticed that for documents from the same domains as the ones in the training corpus the parsing performs with very high precision which is comparable to the state-of-the-art results declared by the parser-developers, but for other documents, however, the precision drops dramatically. This is due to the fact that the parser is fully statistical and does not accept any additional POS input with the parsing string as some other parsers do. Improved syntactic analysis would mean more correct keyterm extraction and better distractors selection.

Our work so far, although employing several different language processing techniques is strongly dependent on the parsing results and limited to the lexemes available in the instructional material. A complementary resource like a Thesaurus of any kind would give us the options to go beyond the limits of the processed text and will extend the capabilities of our system. Dictionary of synonyms/antonyms, dictionaries of names in Bulgarian will also be of great help for defining better possible distractors and go one step further and form a question-like stem instead of a FIB one. For this purpose in future we intend to integrate BalkaNet [6] as a component in the described system.

The lack of additional resources for conceptual processing in Bulgarian is tangible. A terminological dictionary would set a common terminological frame for the analysed materials and would facilitate the keyterm and distractors selection; dictionaries of names would be of great help in defining better possible distractors as well, variety of annotated corpora in different domains would improve the parser performance.

When talking about resources we must mention as well the quality of the input resources. From the processing algorithm it is clear that some kinds of texts are hard to analyse. For example, tabular data just transformed to plain text format will would not constitute good sentences. Mathematical or chemical formulae will hardly fit in any of the patterns adapted for other domains. The input used for similar systems should be carefully adjusted for the specific needs.

Stemming seems to be satisfactory enough. There is no need for applying lemmatisation on extracted terms. In the observed samples, we have not found examples of overstemming or understemming which would be better solved by lemmatisation. We explain this with the fact that after stemming we work mostly with phrases and then inflexional ambiguity is much lower which makes the technique for transforming the wordforms to a single one (stem/lemma) less significant.

[6] Multilingual lexical database comprising of individual WordNets for the Balkan languages

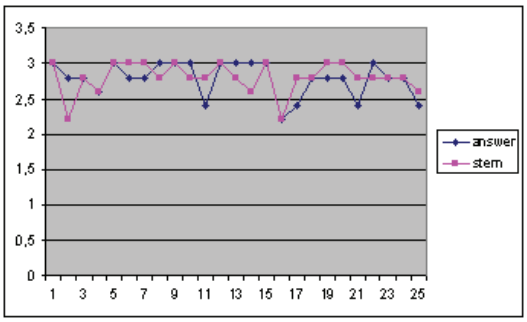

Fig. 3: *Average scores given to stems and answers.*

The chosen morpho-syntactic categories prove to be efficient and catch most of the terminology available in the instructional materials. We are especially satisfied with the addition of the noun phrases of the type np-N-PP which tend to match keyterm phrases in domains with comparatively longer terms like Law. We notice that even more categories coul be added (like the ones satisfying the regular expression A^+N). In comparison with previously reported work we noted that the new approach in distractor suggestion gains significant improvement from filtering useless distractors. Our expectation is that in a large-scale evaluation, the distractors, which are names, would contribute significantly to the overall efficiency of the system.

Under a direct comparison, the results we obtain for Bulgarian are not as good as those obtained for English, but this discrepancy can be explained by the presence of much more sophisticated language technologies for English. The presence of such tools and resources for Bulgarian will help us to gain conceptual knowledge about the target terms, to build more semantically-grounded distractors and to better filter significant from insignificant terms and sentences. Due to the limited resources available for Bulgarian, the capabilities of our system are also limited. However, we have implemented the main idea of the automatic MCTI generation and have shown what can be done with some of the existing language resources for Bulgarian as well as we have also scatched the gaps that need to be addressed in the future.

7 Conclusion and Future Work

Our aim with this work was to explore the field of automatic MCTI generation and to prove the availability and compatibility of the language resources and technologies for Bulgarian as well as to assess the readiness for the implementation of these techniques in real-world applications.

Our ideas for future development are related to experiments with a larger variety of question types and better distractors selection by involving dependency parsing and more external resources. We are working on improvement of the user interface as it is a main issue concerning the test designers' efficiency and will allow a real-time evaluation. Deeper evaluation, including classical test theory and error analysis in order to improve the output is also one of our future goals.

References

[1] I. Aldabe, M. L. De Lacalle, and M. Maritxalar. Automatic acquisition of didactic resources: generating test-based questions. In I. F. de Castro, editor, *Proceeding of SINTICE 07*, pages 105–111, 2007.

[2] J. C. Brown, G. A. Frishkoff, and M. Eskenazi. Automatic question generation for vocabulary assessment. In *HLT '05: Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 819–826, Morristown, NJ, USA, 2005. Association for Computational Linguistics.

[3] C. Fairon. A web-based system for automatic language skill assessment: Evaling. In *Proceedings of Computer Mediated Language Assessment and Evaluation in Natural Language Processing Workshop*, pages 62–67, 1999.

[4] A. Hoshino and N. Hiroshi. A real-time multiple-choice question generation for language testing: A preliminary study. In *Proceedings of the Second Workshop on Building Educational Applications Using NLP*, pages 17–20, Ann Arbor, Michigan, June 2005. Association for Computational Linguistics.

[5] C.-L. Liu, C.-H. Wang, Z.-M. Gao, and S.-M. Huang. Applications of lexical information for algorithmically composing multiple-choice cloze items. In *Proceedings of the Second Workshop on Building Educational Applications Using NLP*, pages 1–8, Ann Arbor, Michigan, June 2005. Association for Computational Linguistics.

[6] R. Mitkov, L. A. Ha, and N. Karamis. A computer-aided environment for generating multiple-choice test items. *Natural Language Engineering*, 12.:177–194, 2006.

[7] P. Nakov. Bulstem: Design and evaluation of inflectional stemmer for bulgarian. In *Proceedings of Workshop on Balkan Language Resources and Tools (1st Balkan Conference in Informatics*, Thessaloniki, Greece, 2003.

[8] I. Nikolova. Language technologies for instructional resources in bulgarian. In K. Balogh, editor, *Proceedings of 13th Student Session at ESSLLI 2008*, pages 135–142, 2008.

[9] S. Petrov, L. Barrett, R. Thibaux, and D. Klein. Learning accurate, compact, and interpretable tree annotation. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 433–440, Sydney, Australia, July 2006. Association for Computational Linguistics.

[10] S. Petrov and D. Klein. Improved inference for unlexicalized parsing. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, pages 404–411, Rochester, New York, April 2007. Association for Computational Linguistics.

[11] E. Sumita, F. Sugaya, and S. Yamamoto. Measuring non-native speakers' proficiency of English by using a test with automatically-generated fill-in-the-blank questions. In *Proceedings of the Second Workshop on Building Educational Applications Using NLP*, pages 61–68, Ann Arbor, Michigan, June 2005. Association for Computational Linguistics.