Language Generation from DB Query

Kristina Kocijan*, Božo Bekavac, Krešimir Šojat
*Department of Information and Communication Sciences
Department of Linguistics
Faculty of Humanities and Social Sciences, University of Zagreb
{krkocijan, bbekavac, ksojat}@ffzg.hr

Abstract

This paper demonstrates how to generate natural language sentences from the pieces of data found in databases in the domain of flight tickets. By using NooJ to add context to specific customer data found in customer data sets, we are able to produce sentences that give a short textual summary of each customer, providing a list of possible suggestions how to proceed. In addition, due to the rich morphology of Croatian, we are giving special attention to matching gender, number and case information where appropriate. Thus, we are able to provide individualized and grammatically correct text in spite of the customer gender or the number of tickets bought and inquiries made. We believe that such short NL overviews can help ticket sellers get a quicker assessment of the type of a customer and allow for the exchange of information with more confidence and greater speed.

1 Introduction

Ever since we have started using computers for language processing, language generation, even in its most primitive form as canned text (Jurafsky and Martin, 2000), was an exciting thing to do. Since its early beginnings in the 1950's, we have made big steps trying to make language generation more adaptable to context i.e. to build systems that can produce a set of appropriate forms and choose the right context-dependent one (Jurafsky and Martin, 2000; Bateman and Zock, 2003; Perera and Nand, 2017; Gatt and Krahmer, 2017). In this paper we will present one such project that maps non-linguistic source into the linguistic form as described in Bateman and Zock (2003).

For this purpose we are using NooJ, a linguistic development environment software. NooJ is not new to language generation (Silberztein, 2012). Due to the power of a transducer that it uses, in collaboration with variables, it has been used in different transformational projects in a variety of languages; from paraphrasing for Portuguese-English machine translation projects (Barreiro, 2008), generating transformations from Italian frozen sentences (Vietri, 2012), or paraphrasing standard Arabic in biomedical texts (Boujelben et al., 2012) to transformation of English direct transitive sentences (Silberztein, 2016a).

This paper focuses on the generation of natural language sentences from databases with records on booking and buying flight tickets. The natural language that we deal with is Croatian, a South Slavic language with rich inflectional and derivational morphology and relatively free word order. Although Croatian is basically a SOV language, word order in sentences can vary due to extensive morphosyntactic marking of major parts of speech and rules of agreement. Agreement in gender, number and person plays an important role in the project presented here. In this paper we describe the generation of brief summaries of previous customers' inquiries and actual purchase of air flight tickets expressed in the natural language.

The paper is structured as follows: after the short introduction, in section 2 we provide the information on what is behind the scenes of the NLG system we propose. In section 3 we present some aspects concerning the usage of the system in real-life environment. In Sections 4, 5 and 6 we continue with the presentation of different parts of the system that will be accompanied with a short discussion explaining the procedures. The paper concludes with an outline of future work.

Proceedings of the Linguistic Resources for Automatic Natural Language Generation Workshop, pages 25–32, Santiago de Compostela, Spain, September 4, 2017.

2 Behind the NLG proposed system

Vayre et al. (2017) give a detailed account of procedures in the building of NLG systems and point out that it normally consist of typical stages. The procedures that are thereby applied can be divided into macro-planning and micro-planning. Macro-planning comprises content selection and document structuring, whereas micro-planning usually refers to the design of syntactic constructions, lexicalization, generation of referring expression, morphological adaptation etc. Morphological adaptation is one of the procedures applied in the design of overall surface realization. Apart from morphological modifications, this last stage also includes typographical adjustment and formatting and provides the final form of the text.

Morphological adjustment (e.g. generation of inflected forms through gender/number or verb/subject agreements) is particularly important for the NLG in our system since Croatian is a highly inflected language with numerous inflectional patterns. Paradigms for nominal parts of speech consist of 7 cases in singular and plural, whereas verbs are inflected for person, number and tense. Some verbal forms, i.e. past participles, are also inflected for gender. Morphosyntactically, NPs as subjects and verbs as predicates agree in the grammatical categories of person and number, whereas verbs determine the case of NPs as objects. NPs as subjects and verbs as predicates also agree in gender if a verbal form consists of an auxiliary verb and a past participle. We can demonstrate this with the following examples:

1. He has bought seven tickets.
   On *je kupi-**o** sedam karata.*

2. She has bought seven tickets.
   Ona *je kupi-**la** sedam karata.*

3. They have bought two tickets.
   Oni *su kupi-**li** dvije karte.*

4. They have bought two tickets.
   One *su kupi-**le** dvije karte.*

As these examples show, the endings of verbal participles are modified according to the subject's number and gender. The subjects in sentences 3 and 4 are the same in English, but they differ in Croatian (in sentence 3 the subject can refer only to masculine and masculine and female gender, whereas the subject in 4 refers solely to feminine).

Sentences 3 and 4 also demonstrate another feature that must be taken into account in the linguistic design of NLG component of our system. Synchronically, the number categories in Croatian are singular and plural. However, earlier stages of language development are manifested in noun forms for plural when quantifiers are numbers two, three and four, and all the other numbers ending in these digits (e.g. 52, 23, 134 etc.). Although these nouns are in the plural, their inflected forms are similar to genitive singular. In these cases there is an evidence of paucal number. For example:

5. He has bought **one ticket.**
   *On je kupio **jednu kartu**.*

6. He has bought **two / three / four tickets.**
   *On je kupio **dvije / tri / četiri karte.***

7. He has bought **five tickets.**
   *On je kupio **pet karata.***

These linguistic issues were taken into consideration in the morphological and syntactic component of our NLG system. A more detailed account is given in section 4.

In the building of the system described in this paper, we were also guided by four major choices that NLG systems must or should make, as defined in Jurafsky and Martin (2000) and Reiter and Dale (2000):

- Content selection – in this case, our content is already provided for the system (*the system is used by ticket sellers only, and ticket buyers have no access to it*);

- Lexical selection – system is choosing a lexical item provided in the set-up pool of items depending on the value of available fields;

- Sentence structure – system produces smaller chunks that are combined into full sentences with appropriate referring (*gender of pronoun referring*) and syntactic features (*tense, number, case*);

- Discourse structure – system combines multiple sentences providing coherent structure (*introducing conjunctions to produce smooth and continuous text*).

In order to deal with one of the main problems of NLG, i.e. control of choosing among the provided alternatives of generated text (Bateman and Zock, 2003), we have found the possibility of using the NooJ linguistic environment coupled with Angular JavaScript Framework as the workable option for our domain scenario.

3 Practical usage

Applications that incorporate NLG systems can significantly speed up the usage of data stored in various databases. The importance of attending to the presentation of such information to the end user and how it can influence the user's cognitive load is well justified by Vayre et al. (2017). As mentioned, the NLG system discussed here is used by sales agents employed by a travel agency. Number of information items and their formatting should not work against them, but rather help them do their job better, faster and with more confidence. One of the ways to help them in that endeavor is to decrease the linguistic complexity of the text that is automatically generated by the system.

The data about customers who buy air tickets either online, by telephone or e-mails, are stored in the database. Since the interpretation of unprocessed data is difficult and time-consuming, there is a significant risk of poor quality of service and a potential loss of clients. Agents dealing with a large number of customers on a daily basis need a straightforward representation of their previous activities in order to improve their productivity and to maintain high quality of service. Thus, a system capable of summarizing and presenting relevant data from databases in an easily understandable form is crucial for the overall improvement of agent-customer relationship.

During the processing of customers' requests, the system automatically recognizes and classifies clients into four categories – golden, silver, bronze and regular defined in the [*Recommendations*] subgraph (Figure 1). This categorization is based on their previous activities (booked and / or purchased tickets, intervals, years, amount of money spent etc.). On the basis of these data the system provides information as to whether a customer is entitled to air tickets at reduced prices or completely free of charge.

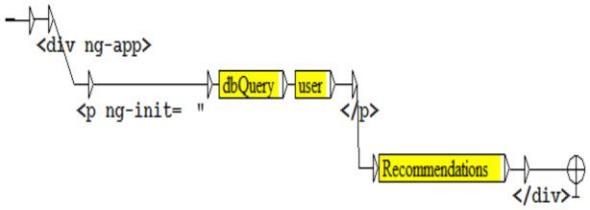

Figure 1: The main grammar

Overviews of previous activities and actual purchases, i.e. short summaries of customers' activities and status as described above, comprise four or five simple and unambiguous sentences in Croatian. These sentences contain all the data relevant for various discounted or special offers for clients, both regular and occasional. The design of the system is discussed in the next section.

4 Building the NLG section

Since we are preparing our results to be used in the network environment, we needed to incorporate all the html tags in our output as well. The main grammar (Error! Reference source not found.) consists of three main sections (subgraphs) that are connected in a manner to support the following logic:

1. recognize the query results and prepare them for initialization in the <p ng-init> tag [subgraph: *dbQuery*];

2. check the user's gender and generate the appropriate gender of a noun, verb and a pronoun in the user's description paragraph [subgraph: *user*];

3. check the user's gender and prepare the appropriate recommendations [subgraph: *Recommendations*].

Within the subgraph [*dbQuery*] (Figure 2) we are recognizing values that exist for the user and that are important to our evaluation of that user.

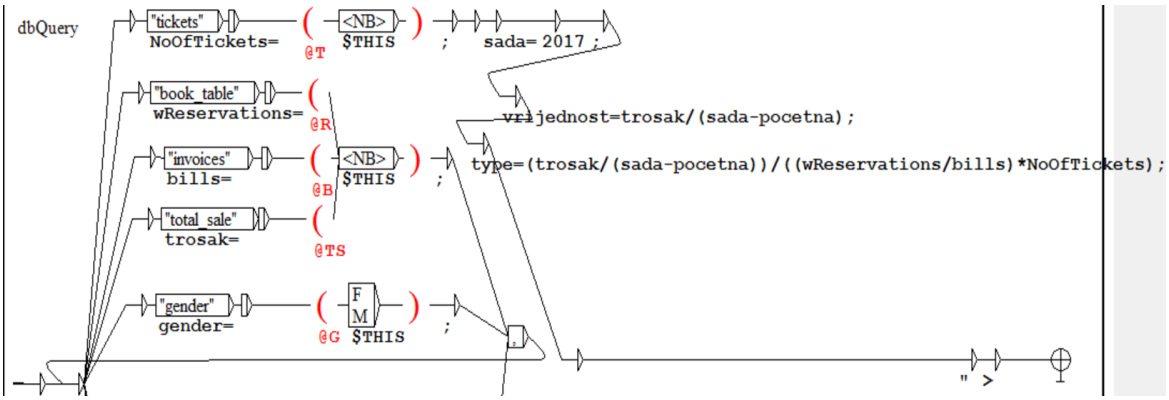

Figure 2: The *dbQuery* subgraph

For the purposes of our project[1], we were interested in the gender field {*gender*}, total amount of money spent since the first purchase {*trosak*}, total amount of invoices sent to the user since the first purchase {*bills*}, total amount of reservations made by the user via web {*wReservations*}, total number of tickets {*NoOfTickets*}.

Except for these fields, we needed to add present year {*sada*}, and formula for calculating the user's yearly average, i.e. how much s/he spends on tickets per year {*vrijednost*} and finally, formula for calculating the type of a user {*type*}. For the second formula, we considered how much money the user spends yearly, number of her/his web reservations, bills issued to the user and number of tickets actually bought. All the other database query results are recognized and annotated, but at this point, we are not using them in this project so they will not be further discussed.

In this grammar, we are using global variables (Silberztein, 2016) to ensure that our query results are available at all levels of the grammar i.e. in the main graph and also in all its subgraphs. We recognize them by the sign '@' used before the variable name. The most important one to us was the variable caring the gender value $@G since we needed this information in the following two sections to determine gender dependent forms of a noun, verb and pronoun, as we will show in the following paragraphs.

Within the subgraph [*user*] (Figure 3) we are introducing three new variables to determine the correct gender forms of a noun, verb and pronoun. The first variable $KO is given the value '*Korisnica*' (Eng. *she*-user) if the graph with the sub-grammar [F] is validated as true i.e. if the global variable $@G has the value set to feminine <$@G="F">. If the variable $@G has the value set to masculine <$@G="M"> then the variable $KO is given the value '*Korisnik*' (Eng. *he*-user).

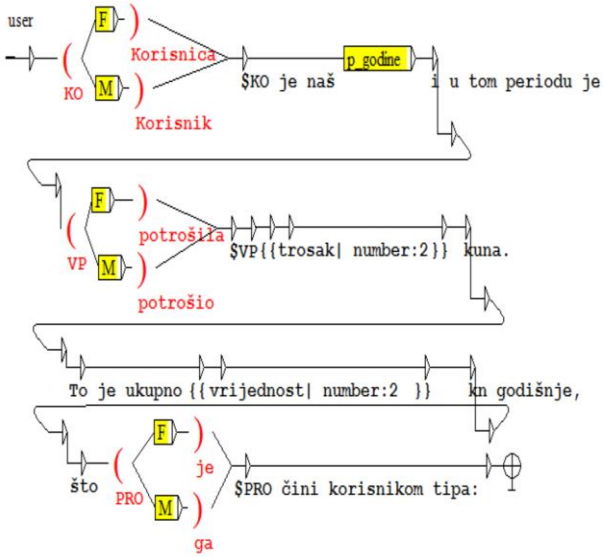

Figure 3: The subgraph *user*

The same validation is checked for the verb 'to spend' which takes the form '*potrošila*' or '*potrošio*' for the feminine and masculine user respectively, and for the accusative form of the pronouns 'she' and 'he' that become '*nju*' and '*ga*' in Croatian, depending on the gender. Since Croatian verbal past participles are gender dependent, we have used the constraint on customer's gender to produce the

[1] We believe that each agency will work with its own parameters that make up their types of different users. Parameters we chose here are for demonstration purposes only.

correct verb forms. If the constraint <$@G=”F”> is validated, NooJ takes the upper path and uses correct female forms of the main verb. Combination of gender constraints and tense operations allows us to generate correct sentences.

If all the validations check out correctly, there are two possible variants of this paragraph that can appear to the agent – one for the feminine (a) and one for the masculine user (b).

(a) ***Korisnica*** *je naš član X godina i u tom period je* ***potrošila*** *Y,00 kuna. To je ukupno Y,00 kn godišnje, što* ***nju*** *čini korisnikom tipa:* (Eng. **She-user** is our member for X years and in that period **she-spent** Y,00 kunas. That is a total of Y,00 kunas per year, which makes **her** a user of type:)

(b) ***Korisnik*** *je naš član X godina i u tom period je* ***potrošio*** *Y,00 kuna. To je ukupno Y,00 kn godišnje, što* ***ga*** *čini korisnikom tipa:* (Eng. **He-user** is our member for X years and in that period **he-spent** Y,00 kunas. That is a total of Y,00 kunas per year, which makes **him** a user of type:)

In the text, X and Y are replaced by the values calculated for each user in real time.

The *user* subgraph has one additional sub-grammar [*p_godine*] that checks for the number of years the user has been a customer (Figure 4). This check was necessary for two reasons:

- if our user is a new user, then s/he is described as a ‘*novi član*’ (Eng. new user) and we do not use the number of years to describe how long s/he has been the user. This way we have avoided awkward sentences like ‘*User has been our member for 0 years*.’ ;[2]

- for all the users that have been using the service for more than a year, we use the full number of years since s/he first used the services provided by the company. However, since the word for ‘year’ in Croatian changes its form depending on the number that precedes it, it was necessary to connect the proper number with the proper word form. Thus, if less than one year has passed since the first contact and today {*sad_poc*}, there are no years in between and we consider this person to be the new user. If the last digit is however greater than 0 and lower than 2, the word after the number {*god_clan*} takes the form ‘*godin**u***’; if it is greater than 1 and lower than 5 it takes the form ‘*godin**e***’ and if it is greater than 4 it takes the form ‘*godin**a***’. Since NooJ does not support mathematical operations, in order to check the difference between the first contact and today, we moved these calculations to the web environment, but used NooJ to prepare the ground for all the possible calculations.

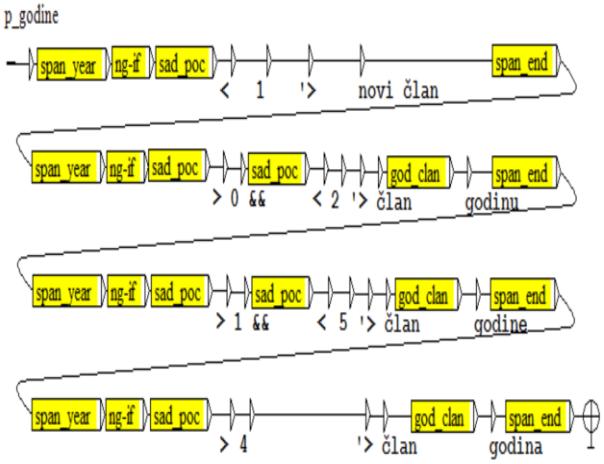

Figure 4: The subgraph *p_godine*

Similar check was performed in the final subgraph *Recommendations* (Figure 5).

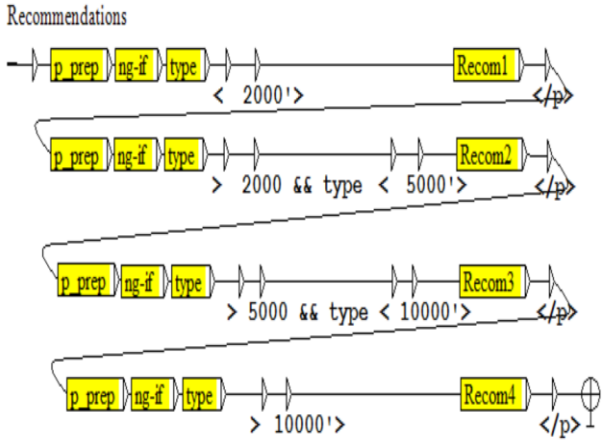

Figure 5: The subgraph *Recommendations*

In this subgraph we had to calculate the type of the user {*type*}, using the formula already prepared in the subgraph [*dbQuery*]. NooJ will again generate all four recommendations [*Recom1 .. Recom4*],

[2] Cf. section 2, examples 5,6 and 7.

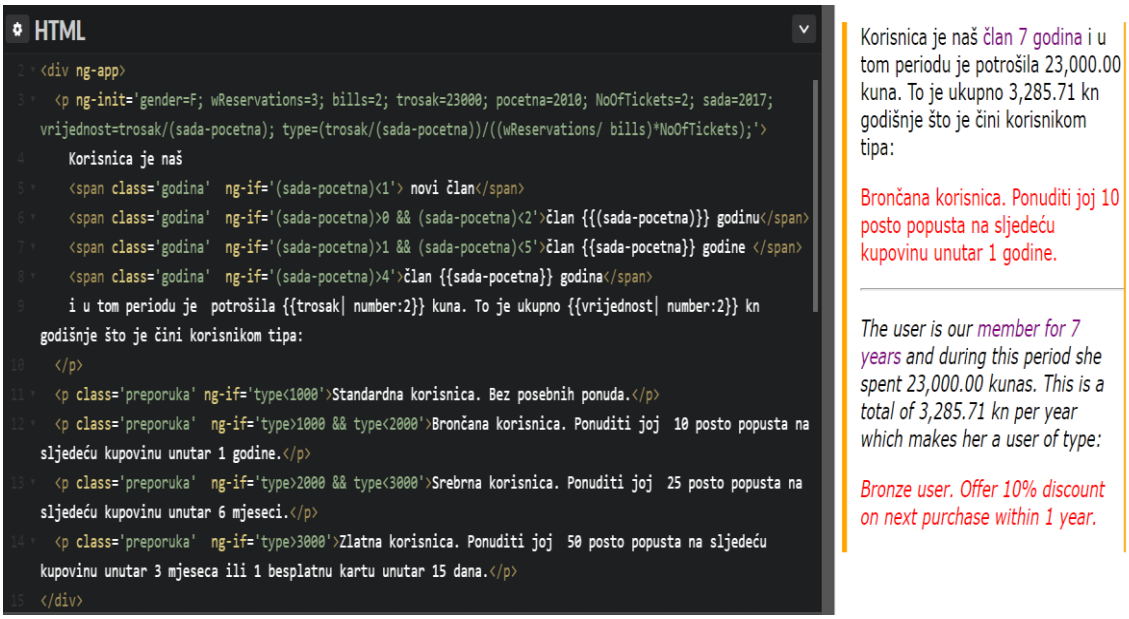

Figure 6: HTML code generated in NooJ (on the left) and its representation in a web viewer (on the right)

adopting them to the gender defined within the global variable $@G, while the final choice among offered recommendations will be performed within the web browser using the AngularJS.

Thus, depending on the gender, there are again two possible sets of recommendations that may be generated - (a) for the feminine and (b) for the masculine user:

(a) ***Recom1: Standardna korisnica***. *Bez posebnih ponuda.* (Eng. ***She*-Standard user**. No special offers.)

Recom2: Brončana korisnica. *Ponuditi* ***joj*** *10% popusta na sljedeću kupovinu unutar 1 godine.* (Eng. ***She*-Bronze user**. Offer **her** 10% discount on next purchase within 1 year.)

Recom3: Srebrna korisnica. *Ponuditi* ***joj*** *25 % popusta na sljedeću kupovinu unutar 6 mjeseci.* (Eng. ***She*-Silver user**. Offer **her** 25% discount on next purchase within 6 months.)

Recom4: Zlatna korisnica. *Ponuditi* ***joj*** *50% popusta na sljedeću kupovinu unutar 3 mjeseca ili 1 besplatnu kartu unutar 15 dana.* (Eng. ***She*-Golden user**. Offer **her** 50% discount on next purchase within 3 months or 1 free ticket within 15 days.)

(b) ***Recom1: Standardni korisnik***. *Bez posebnih ponuda.* (Eng. ***He*-Standard user**. No special offers.)

Recom2: Brončani korisnik. *Ponuditi* ***mu*** *10% popusta na sljedeću kupovinu unutar 1 godine.* (Eng. ***He*-Bronze user**. Offer **him** 10% discount on next purchase within 1 year.)

Recom3: Srebrni korisnik. *Ponuditi* ***mu*** *25 % popusta na sljedeću kupovinu unutar 6 mjeseci.* (Eng. ***He*-Silver user**. Offer **him** 25% discount on next purchase within 6 months.)

Recom4: Zlatni korisnik. *Ponuditi* ***mu*** *50% popusta na sljedeću kupovinu unutar 3 mjeseca ili 1 besplatnu kartu unutar 15 dana.* (Eng. ***He*-Golden user**. Offer **him** 50% discount on next purchase within 3 months or 1 free ticket within 15 days.).

5 Dealing with the control within the Web environment

There are several calculations that our project requires (number of years between user's first contact and today, user's average spending, type of the user depending on her/his spending…) in order to generate proper sentences. Since they could not be dealt with inside the NooJ environment, we have opted for AngularJS[3] that is considered to be "the most popular JavaScript MV (model view) solution in the world today" (Smith:Introduction, 2015). Its code allowed us to extend the HTML code with some new attributes that allow for JavaScript type functionality.

For this reason, it was necessary to incorporate all the needed AngularJS code in the text generated within NooJ. This is also the reason why all the text that depended on some mathematical calculations was generated and exported to the web environment where the final choice was made based upon the calculations (Figure 6).

The left side of Figure 6 shows the entire code prepared within NooJ, but notice that on the right side, not all generated parts of sentences[4] are shown. This was made possible by AngularJS part of the text. In fact, we gave Angular control over the tag which holds our text. We constrained its scope only to this section of the page so it would not interfere with other frameworks used originally by the application.

6 Discussion and future work

We have demonstrated the procedure for a fast and straightforward recognition of customers' activities, their classification into various categories based on previous activities and the production of help messages for further interaction between a sales agent and a customer.

At this time, we have only considered situations when the user is a single private person, male or female. The problem of dealing with the company representatives still needs to be solved. But, if such a user can be distinguished within the database data, the grammar can adequately be extended with new sets of validations that will allow for the generation of new user specific descriptions and appropriate sets of recommendations.

In further work we intend to expand the algorithms used so far in order to enable predictions about future needs and desires of a customer. For example, if a customer regularly makes inquiries about flights and tickets using the web page interface, but the number of confirmed reservations is either decreasing or they are not realized at all, this can indicate that functionality of the web page is not satisfactory. This can also indicate that customers actually use web pages of other travel agencies for booking and purchase of air tickets.

Another line of research that we wish to pursue in the future is the generation of automatic reports for sales managers. These reports provide brief summarizations of all the activities recorded in the agent-customers interactions and enable quick changes or modifications of business strategies if necessary. By using NLG systems, the time required for the creation of such reports is shortened and it is possible to make quick decisions.

Further, such reports facilitate a better distribution of manpower, i.e. travel agents can direct their attention toward an individual client and her/his particular needs. For example, if the same customer makes online inquiries about flights without confirmation of reservation over several days, the system should alert a travel agent about these activities.

On the basis of these data, a sales agent can automatically generate an offer according to the parameters of the customer's search, using predefined textual samples. The intervention of sales agents in such cases would be minimal or even not necessary, since the system should be able to automatically make decisions and create offers in the form of short texts using the data stored in the database.

To sum up, a quality customer relationship management system nowadays should predict customers' wishes and needs and enable appropriate, efficient and quick actions.

7 Conclusion

This project presents the first steps in the natural language generation for Croatian in the domain of flight tickets. On the basis of data from a database

[3] https://angular.io/docs

[4] The English translation provided below the Croatian text is given here only for demonstrational purposes and is not part of the original project.

query, we are able to generate a text that gives an agent a quick summary of a customer with possible suggestions on how to proceed in her/his conduct. Such a quick insight should help agents make multi-criteria decisions faster and with more confidence, but within the business approved parameters. By producing natural language text that reduces the cognitive effort, agents can provide better service to their customers and thus upgrade the business results.

Acknowledgments

The authors wish to thank Travel Management Company d.o.o. for providing needed training data for this project.

References

Anabela Barreiro. 2008. ParaMT: a paraphraser for machine translation in *Lecture Notes in Computer Science,* vol. 5190, Springer-Verlag, pp 202-211.

John Bateman and Michael Zock. 2003. Natural Language Generation in *The Oxford Handook of Computational Linguistics* (ed. Ruslan Mitkov), Oxford University Press, pp 284-322.

Ines Boujelben, Slim Mesfar, Abdelmajid Ben Hamadou. 2012. Transformational Analysis of Arabic Sentences: Application to Automatically Extracted Biomedical Symptoms in *Automatic Processing of Various Levels of Linguistc Phenomena* (eds. K. Vučković, B. Bekavac, M. Silberztein), Newcastle upon Tyne: Cambridge Scholars Publishing, pp 182-194.

Albert Gatt and Emiel Krahmer. 2017. *Survey of the State of the Art in Natural Language Generation: Core tasks, applications and evaluation*. `https://arxiv.org/abs/1703.09902`. Date Accessed: May 27, 2017.

Daniel Jurafsky and James H. Martin. 2000. *Speech and Language Processing: An introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*, Prentice Hall, Upper Saddle River, New Jersey.

Rivindu Perera and Parma Nand. 2017. Recent Advances in Natural Language Generation: A Survey and Classification of the Empirical Literature. *Computing and Informatics,* Vol 36, No 1 (2017), pp 1-32.

Ehud Reiter and Robert Dale. 2000. *Building Natural Language Generation Systems*. Cambridge UP, Cambridge, UK.

Max Silberztein. 2012. Automatic Transformational Analysis and Generation in *Automatic Processing of Various Levels of Linguistc Phenomena* (eds. K. Vučković, B. Bekavac, M. Silberztein), Newcastle upon Tyne: Cambridge Scholars Publishing, pp 221-231.

Max Silberztein. 2016a. *Joe loves Lea:* Transformational Analysis of Direct Transitive Sentences in *Automatic Processing of Natural-Language Electronic Texts with NooJ. NooJ 2015* (eds. T. Okrut, Y. Hetsevich, M. Silberztein, H. Stanislavenka). Communications in Computer and Information Science, vol 607. Springer, Cham, pp 55-65.

Max Silberztein. 2016b. *Formalizing Natural Languages: The NooJ Approach,* Wiley. USA.

Chris Smith. 2015. *Angular Basics*. `http://www.angularjsbook.com/`. Date Accessed: May 11, 2017.

Jean-Sébastien Vayre, Estelle Delpech, Aude Dufresne and Céline Lemercier. 2017. Communication Mediated through Natural Language Generation in Big Data Environments: The Case of Nomao. *Journal of Computer and Communications,* 5, `https://doi.org/10.4236/jcc.2017.56008`, pp 125-148.

Simonetta Vietri 2012. Transformations and frozen sentences in *Automatic Processing of Various Levels of Linguistc Phenomena* (eds. K. Vučković, B. Bekavac, M. Silberztein), Newcastle upon Tyne: Cambridge Scholars Publishing, pp 166-180.