
TEST AND ANALYSIS OF EXPERIMENTS FOR THE APPLICABILITY OF A SEMANTIC ANALYZER FOR NATURAL LANGUAGE INTERFACE TO THE DATABASE

Maria Zhekova

University of Plovdiv “Paisii Hilendarski”, Bulgaria, mimijekowa@abv.bg

Abstract: The complexity of formalization in the Bulgarian language leads to the need to develop models and theories for grammatical modeling in order to understand natural language text in natural Bulgarian. The study presents experiments on a semantic analyzer that interprets natural human language by processing and analyzing data obtained from a request in the user's natural language. Undoubtedly semantic analysis plays an important role in analyzing and organizing textual content and is a key element in natural-language dialog systems. The proposed parser is based on concepts of generative grammar and is composed of a semantic analyzer that can be refined over time and a specific lexical corpus for the subject area under consideration. The purpose of the study is to compare the semantic structure contained in the question with the corresponding values of table names and database attributes containing correspondence templates (rules), as well as interpretation procedures that determine the relationships and relations between the constituent elements. The emphasis in the development is on the presentation of quality declarative rules, limited to a correct subset of the area under consideration, to represent the links in the database to retrieve more facts. They are flexible, manageable and understandable for both machines and humans. Following recognition and classification using these annotations, a database query is created and the information retrieved is retrieved. Semantic analyzer and query processing make natural language interface systems more useful to users than traditional dialog systems that provide direct responses by accepting GUI parameters.

This article provides basic SQL query templates that correspond to user questions in natural language. Experimental tests have been developed to demonstrate the algorithm and structure of the information database. The proposed system architecture and algorithm, unlike the rest, do not restrict the user to word choice and syntax. The combination of natural language processing and the proposed method for creating a SQL query to the knowledge base effectively enhances the human-computer communication process and is a means by which ordinary users can obtain useful information in their native language. Aim for future development is the development of a modern dialogue system for word processing in Bulgarian, which makes the previous rule-based models more automated, improving the quality of the rules and returning correct and accurate answers from the system.

Keywords: semantic analyzer, natural language processing, natural language interface, dialog system

ПРОВЕЖДАНЕ НА ТЕСТОВЕ И АНАЛИЗ НА ЕКСПЕРИМЕНТИТЕ ЗА ПРИЛОЖИМОСТ НА СЕМАНТИЧЕН АНАЛИЗАТОР ЗА ЕСТЕСТВЕНЕЗИКОВ ИНТЕРФЕЙС КЪМ БАЗА ДАННИ

Мария Жекова

Пловдивски университет “Паисий Хилендарски”, България, mimijekowa@abv.bg

Резюме: Сложността за формализация на българския език води до необходимостта от разработване на модели и теории за граматическо моделиране с цел естественоезиково разбиране на текст на естествен български език. Изследването представя експерименти върху семантичен анализатор, който интерпретира естествен човешки език, като обработва и анализира данни, получени от заявка на естествения език на потребителя. Безспорно семантичният анализ играе важна роля за анализа и организирането на текстово съдържание и е ключов елемент в естественоезиковите диалогови системи. Предложеният парсер се основава на концепции за генеративна граматика и е съставен от семантичен анализатор, който може да се усъвършенства във времето и специфичен лексикален корпус за разглежданата предметна област. Цел на изследването е да се съпостави семантичната структура съдържаща се във въпроса със съответни стойности от имена на таблици и атрибути от базата данни, съдържащи шаблони за съответствия (правила), както и процедури за интерпретация, определящи връзките и релациите между съставлящите елементи. Акцент в разработката е представянето на качествени декларативни правила, ограничени до коректно подмножество на разглежданата област, за представяне на връзките в базата данни с цел извличане на повече факти. Те са гъвкави, управляеми и разбираеми както за машините, така и за хората [Жекова‘2019]. След разпознаване и

класификация използвайки тези анотации, следва създаването на заявка към базата данни и извличане на търсената информация. Семантичният анализатор и последващите обработки на заявката, правят системите с естественоезиков интерфейс по-полезни за потребителите от традиционните диалогови системи, които предоставят директни отговори, чрез прием на параметри от графичен потребителски интерфейс.

В статията са предложени основните шаблони на SQL-заявки, които кореспондират на потребителските въпроси на естествен език. Разработени са експериментални тестове за демонстриране на алгоритъма и структурата на информационната базова основа. Предлаганата архитектура на системата и алгоритъм за анализ, за разлика от останалите, не ограничават потребителя до избор на думи и синтаксис. Комбинирането на естественоезиковата обработка и предложения метод за създаване на SQL заявка към база знания по ефективен начин подобрява процеса на комуникация между човек и компютър и е средство, чрез което обикновените потребители могат да получат полезна информация на родния си език. Цел за бъдещо развитие е разработване на съвременна диалогова система за обработка на текст на български език, която прави предишните модели, базирани на правила по-автоматизирани, подобрявайки качеството на правилата и връщане на коректни и точни отговори от системата.

Ключови думи: семантичен анализатор, естественоезиков интерфейс, диалогова система

1. ВЪВЕДЕНИЕ

Лингвистичните процесори, наричани още парсери са действащи или експериментални компютърни модели, които имат различно предназначение - за експертна оценка, машинен превод, изкуствен интелект, автоматизирано аотиране и др. Семантичният анализатор е инструмент за извличане на структурирани данни (факти) от текст на естествен език. Фактите се извличат с помощта на безконтекстни граматика и речници на ключови думи [Колковска'2005]. Той дава обща представа за това как се съставят граматика и речници, как се конфигурира и стартира анализаторът. Описание на синтаксиса на граматиката, списъци с използвани етикети, таблици с използвани правила са разгледани в предишни наши проучвания и изследвания в областта.

За експериментите, описани в това изследване са използвани две топологии. Едната, която е ключов момент в анализатора, е лексикалният корпус, който съдържа подробна морфологична, синтактична и семантична информация, селективни ограничения, свойства, валентности и характеристики, които могат да повлияят на анализа. Втората е йерархичен фреймов модел на взаимно свързани обекти от областта, организирани и свързани помежду си посредством правила за съответствие, факти и функционални ограничения. Техниките за интерпретиране на съдържанието варират [Стайкова'2015] от създаването на таблици съпоставящи понятията от предметната област и лингвистичния ресурс на анализатора, до различни по сложност концептуални модели.

2. ГРАМАТИЧЕСКО МОДЕЛИРАНЕ

Обработката на естествен език е област от изкуствения интелект, в която компютърните системи обработват естествен език [Sowah, 2018]. Обработката на сложни въпроси включва идентифицирането на семантични структури от фрейми. В теоретичния фреймов модел като елементи на разглежданата фреймова система участват различни понятия с техните връзки, описани чрез факти и правила от съответната предметна област [Жекова'2019]. В структурно отношение той представлява системно обвързани понятия с различно равнище на абстрактност и е първообраз на бъдещата база данни. Предложения семантичен анализатор, базиран на корпуса от свързани помежду си фрейми ще може да улавя въпросите и по този начин анализирайки първоначалните данни от съдържанието на запитването и сравнявайки го с наличните топологии ще може да отговори на потребителско запитване. Първоначално е конструиран лексикален речников ресурс (лексикон) изграден на основата на фрейми, който описва с термините на компютърната лингвистика думите и фразите в българския език. Акцентът тук е представянето на качествени декларативни шаблони (правила), ограничени до коректно подмножество на разглежданата област, за представяне на връзките в базата данни с цел извличане на повече факти. Те са гъвкави, управляеми и разбираеми както за машините, така и за хората. И тук стои въпросът – как ефективно да се включат декларативните правила в предложението фреймов модел за описание на предметната област, за да се направи коректно извличането на информация от базата данни [Sowah'2018].

Цел в проучването е да се разработи система за справяне със задачата за извличане на отговор от база знания, чрез потребителски зададен въпрос на естествен човешки език, в случая български език.

Предложеният парсер се основава на концепции за генеративна граматика и е съставен от семантичен анализатор, който може да се усъвършенства във времето и специфичен лексикален корпус за разглежданата предметна област. Процедурата по анализа е еднократно преминаване и токенизация на отделните лексеми

като се започне от началото на входящия текст, сегментира се всеки токен (намират се търсените параметри), анализират се сегментите (намират се търсените таблици), категоризира се типа на въпроса и се идентифицира типа на SQL заявката, която отговаря на потребителското търсене.

Нашата цел е да се съпостави семантичната структура съдържаща се във въпроса, със съответни таблици и атрибути от базата данни, съдържащи правила за съответствия, както и процедури за интерпретация, определящи връзките и релациите между съставящите елементи.

3. ПОСЛЕДОВАТЕЛНОСТ ОТ ДЕЙСТВИЯ

За проведените тестове, експерименти и оценки използваме ограничено количество данни за предметната област, специфични за задачата, фокусирани само върху темата за изследването, а именно – електронен Университет.

Има две предизвикателства при семантичен анализатор на текст на естествен език. Първото е използването на различен термин за обозначаване на едно и също понятие и второто е, че едно и също понятие се намира в различни таблици в базата данни, имайки нееднозначно значение [Sun'2005]. За да се справим с тези проблеми използваме семантичен анализатор, за да обединим различни семантични представи и за да намерим синоними и семантично свързани понятия.

Чрез предложената последователност от действия [Zhekova'2019], информационната система превежда потребителски въпрос на естествен език до изпълнима SQL заявка, като диференцира и параметризира частите на речта и достига до правилата и релациите, до които те водят.

- Потребителски въпрос – важен въпрос тук е дали информацията, кодирана във входящия граматичен модел се запазва или част от нея се губи. Извършва се предварителна лингвистична обработка като от въпросителния низ се премахват символи като – точка, запетая и т.н.
- Токенизация – превръщането на потребителския въпрос в списък от токени, или извършване на предварителна лингвистична обработка.
- Разпознаване на понятия – думите в зададения потребителски въпрос на естествен език са от значение, ако те са такива, които потребителя възприема като съдържащи стойностна информация.
- Разпознаване на базови понятия – проверяват се всички получени токени и от тях се стига до основните понятия, на които думите в потребителския низ са синоними.
- Открива се и се извлича свързващата информация, а именно на всяко понятие от входния низ се намират всички срещания в базата данни като имена на таблици и атрибути, с които понятието е релевантно.
- Идентифициране на SQL шаблон – измежду всички възможни комбинации на разпознатите базови понятия се избират тези, които отговарят на условието таблиците да бъдат свързани, за да се направи коректна заявка към базата данни.
- Идентифициране на SQL заявка – след като е изследван типа на атрибутите се съставя заявка търсейки отговор на потребителския въпрос.
- Изпълнение на SQL заявката и връщане на отговор в потребителския интерфейс.

Табл.1 Съответствия между понятия, таблиците и полетата, с които се асоциират и в които се намира свързващата ги информация

I	Понятие	Таблица	Атрибути
1	студент	Student	PersonalData.StudentID; PaymentFee.StudentID; ExamProtocol.StudentID
2	преподавател	Teacher	PersonalData.TeacherID; ExamProtocol.TeacherID
3	специалност	Specialty	Student.SpecialtyID; uPlan.SpecialtyID; ExamProtocol.SpecialtyID;
4	дисциплина	Discipline	Teacher.DisciplineID; uPlanContent.DisciplineID; ExamProtocol.DisciplineID
5	факултетен номер	Student	Student.FacNumber
6	факултет	Department	Teacher.DepartmentID
7	избираема	Discipline	IsOptional
8	националност	PersonalData	Nationality
9	ЕГН	PersonalData	EGN
10	Научна степен	EduDegree	Specialty.EduDegreeID
11	Форма на обучение	EduForm	Specialty.EduFormID; uPlan.eduFormID

Табл.2 Списък със съответствия между понятия и техните синоними

Основно понятие	Синонимен ред
студент	обучаван, обучаващ, възпитаник, учащ, учи, посещава, изучава
преподавател	ръководител, лектор, водещ
университет	висше учебно заведение, учебно заведение, ВУЗ, академия
преподава	води
декан	завеждащ факултет
факултетен номер	номер, факултетен номер
дисциплина	предмет

4. SQL ШАБЛОНИ НА ЗАЯВКИ КЪМ БД

4.1. Simple select

SQL оператора се използва за извличане на информация само от една таблица от БД.

4.2. Select with WHERE clause

WHERE клаузата се използва за извличане на записи, които отговарят на определено условие. *Condition* - израз, който се изчислява на true. Използва се за филтриране на записи.

1. С една единствена атомарна стойност (value) в израза;
2. С две или повече атомарни стойности, разделени с И / ИЛИ. Филтрират се записи въз основа на повече от едно условие.

Оператори в WHERE клаузата - =, >, <, >=, <=, <>, BETWEEN, LIKE, IN

4.3. Select with JOIN clause

Използва се в заявки, извличащи информация от няколко таблици, свързани посредством общ атрибут. Добавяне на GROUP BY и ORDER BY към съществуващ SELECT statement за групиране и подреждане по един или повече атрибути в заявка към БД.

Допълнителни клаузи

- **Select with GROUP BY clause** – използва се за извличане на информация, чрез групиране на записите по определен критерий.
- **Select with ORDER BY clause** – използва се за извличане на информация, чрез сортиране на записите във възходящ ред по подразбиране.
- **Select with MIN() and MAX() functions** – връщат най-малката или най-голямата стойност от търсения атрибут.
- **Select with COUNT(), AVG() and SUM() functions** – COUNT() връща броя на записите, AVG() връща средната стойност на числов атрибут, SUM() връща сума на числов атрибут.

Табл.3 SQL шаблони на въпроси в диалогова система с естественоезиков интерфейс

SQL шаблони	Синтаксис на шаблона
<i>Основни</i>	
Simple select	SELECT attr1, attr2, FROM table1
Select with WHERE clause	SELECT attr1, attr2, FROM table1 WHERE condition1 AND/OR condition2 ...
Select with JOIN clause	SELECT attr1, attr2, FROM table1 INNER JOIN table2 ON join condition WHERE condition
<i>Допълнителни</i>	
Select with GROUP BY clause	SELECT attr1, attr2, FROM table1 WHERE condition GROUP BY attr1
Select with ORDER BY clause	SELECT attr1, attr2, FROM table1 WHERE condition ORDER BY attr1, attr2
Select with MIN() function	SELECT MIN(attr1), attr2 FROM table1
Select with MAX() function	SELECT attr1, MAX(attr2) FROM table1 WHERE condition
Select with SUM() function	SELECT SUM(attr1), attr2 FROM table1
Select with AVG() function	SELECT AVG(attr1), attr2 FROM table1
Select with COUNT() function	SELECT COUNT(attr1), attr2 FROM table1

5. ПРОВЕЖДАНЕ НА ТЕСТОВЕ И ЕКСПЕРИМЕНТИ

Пример 1: Кои студенти изучават дисциплината „Облачни технологии“ с преподавател „Емилия Христова“?

1. Извършва се предварителна лингвистична обработка. Нормализира се входящия текст. Полученият списък с токени е следния:

Токен [0] – Кои	Токен [1] – студенти
Токен [2] – изучават	Токен [3] – дисциплината
Токен [4] – „Облачни технологии“	Токен [5] – с
Токен [6] – преподавател	Токен [7] – „Емилия Христова“

2. В Списъка на съответствията между понятията и техните синоними (вж. Табл.2) се проверяват всички токени и чрез тях се стига до основните понятия, на които думите са синоними.
3. Премахват се излишните думи, които не се съдържат в лексикалния корпус – кои, с.
4. От Таблицата на съответствия между понятия, таблиците и полетата, с които се асоциират (вж. Табл.1) се извлича свързващата информация, а именно на всяко понятие от входния низ се намират всички срещания в базата данни като имена на таблици и атрибути, с които понятието е релевантно.
5. Получават се следните съответствия:

Понятие	Таблица	Атрибут
Студент	Student-A1 PaymentFee-A2 ExamProtocol-A3	StudentID
Дисциплина	Discipline-B1 uPlanContent-B2 Teacher-B3	DisciplineID
Преподавател	Teacher-C1 ExamProtocol-C2	TeacherID

От тях са възможни следните комбинации: (A1, B1, C1); (A1, B2, C1); (A1, B3, C1); (A1, B1, C2); (A1, B2, C2); (A1, B3, C2); (A2, B1, C1); (A2, B2, C1); (A2, B3, C1); (A2, B1, C2); (A2, B2, C2); (A2, B3, C2); (A3, B1, C1); (A3, B2, C1); (A3, B3, C1); (A3, B1, C2); (A3, B2, C2); (A3, B3, C2);

От възможното множество комбинации се избират тези, които отговарят на условието, таблиците да бъдат свързани, за да се направи коректно заявката към базата данни. => (A1, B1, C1) и (A1, B3, C1) отговарят на условието.

A1, B1 и C1 са три таблици, а B3 е атрибут-релация, свързваща две от таблиците. Следователно шаблона, който ще се приложи, за създаване на SQL заявката към базата данни ще бъде от третия вид – Select with JOIN clause.

Изследват се типа на атрибутите. Обръща се внимание и на премахнатите излишни думи. Търси се отговор на израза, чиято стойност е известна, за да се определи условието за търсене - *condition*.

Наличието на атомарни стойности означава, че трябва да е изпълнено определено условие, което свежда задачата до използване на друг SQL шаблон - Select with WHERE clause.

Лексемата ‘преподавател’ стояща точно пред атомарната стойност означава, че това е стойност на атрибута *Name* от таблица *Teacher*, а лексемата ‘дисциплина’ стояща точно пред атомарната стойност означава, че това е стойност на атрибута *Name* от таблица *Discipline*. Наличието на две атомарни стойности е предпоставка за повече от едно условие в WHERE клаузата, т.е. в Select statement-а ще присъства WHERE клауза. Отсъствието на отрицателна частица отхвърля възможността за две различни по стойност условия, което означава, че трябва да са изпълнени и двете с условие true.

```
select Student.Name, Discipline.Name, Teacher.Name
from Student
```

```
inner join Specialty on Specialty.ID = Student.SpecialtyID
```

```
inner join uPlan on uPlan.SpecialtyID = Specialty.ID
```

```
inner join uPlanContent on uPlanContent.uPlanID = uPlan.ID
```

```
inner join Discipline on Discipline.ID = uPlanContent.DisciplineID
```

```
inner join Teacher on Teacher.ID = Discipline.TeacherID
```

```
where Teacher.Name = 'Емилия Христова' and Discipline.Name= 'Облачни технологии'
```

Резултата от потребителското търсене е списък с имена на студенти, изучаващи дисциплината „Облачни технологии“ при преподавател „Емилия Христова“.

Пример 2: Кои преподаватели преподават/водят лекции на специалността „Компютърни системи“?

Понятие	Таблица	Атрибут
Специалност	Specialty-A1 Student-A2 uPlan-A3 ExamProtocol-A4	SpecialtyID
Преподавател	Teacher-B1 ExamProtocol-B2	TeacherID

(A1,B1); (A1,B2); (A2,B1); (A2,B2); (A3,B1); (A3,B2); (A4,B1); (A4,B2) са всички възможните комбинации. uPlan и ExamProtocol отпадат, защото потребителското търсене не е за учебен план, изпитен протокол или студенти. Остава комбинацията (A1,B1), в която A1 и B1 представляват две таблици, а атомарната стойност се отнася за специалност. Следователно SQL шаблона ще се състави обединявайки две таблици посредством JOIN и изпълнявайки конкретно условия посредством WHERE.

```
select Specialty.Name, Teacher.Name from Specialty
inner join uPlan on Specialty.ID = uPlan.SpecialtyID
inner join uPlanContent on uPlan.ID = uPlanContent.uPlanID
inner join Discipline on Discipline.ID = uPlanContent.DisciplineID
inner join Teacher on Teacher.ID = Discipline.TeacherID
where Specialty.Name = 'Компютърни системи'
```

Резултата от потребителското търсене е списък с преподаватели, извеждащи специалността „Компютърни системи“.

Пример 3: Кои дисциплини са избираеми?

От Таблицата на съответствията на понятията с имената на таблиците и полетата, с които се асоциират (вж. Табл.1) се вижда, че понятието „дисциплина“ се среща и като таблица – Discipline, и като връзка (Foreign key) с други таблици – uPlanContent.DisciplineID и Teacher.DisciplineID, а понятието „избираема“ се среща само като атрибут в таблица uPlanContent.

Понятие	Таблица	Атрибут
Дисциплина	Discipline-A1 Teacher-A3	*
Избираема	uPlanContent-B1	IsOptional

Следователно – има две таблици, атрибутите на които, са предмет на търсене. Очевидно търсеният SQL шаблона е SQL Pattern with JOIN. Изследва се типа на атрибутите. Обръща се внимание и на премахнатите излишни думи. Търси се отговор на израза, чиято стойност трябва да се знае, за да се определи условието за търсене. А именно отсъствието на частица „НЕ“ означава, че се търсят всички положителни отговори. IsOptional е параметър, който приема булеви стойности.

Следователно условието на което трябва да отговаря потребителската заявка е IsOptional=true

Наличието на определено условие вече довежда до друг SQL шаблон: Select with WHERE clause.

```
SELECT Discipline.Name, uPlanContent.IsOptional FROM Discipline
inner join uPlanContent on uPlanContent.DisciplineID = Discipline.ID
where uPlanContent.IsOptional = '1'
```

Резултат от потребителското търсене е списък с дисциплини, отговарящи на условието да са избираеми.

6. АНАЛИЗ НА ЕКСПЕРИМЕНТИТЕ

Изследванията демонстрират как потребителите могат да изпращат заявки към съвместими бази данни, използвайки естествен език и да получават информация, различна от обичайната, получена от традиционните системи за бази данни. Както SQL операторът, така и NLP обработката водят до едни и същи резултати в система от бази данни, въпреки че NLP улеснява процеса на подаване на заявки. Семантичния анализ предложен в разработката, показва възможността за комбиниране на характеристиките на тези методологии за създаване на ефективна система от бази данни и подобрява допълнително комуникацията между потребителите на базата данни и системите от бази данни по естествен начин спрямо традиционния начин за взаимодействие с бази данни. Комбинирането на естественоезиковата обработка и предложения метод за създаване на SQL заявка към база знания по ефективен начин подобрява процеса на комуникация между човек и компютър и е средство, чрез което обикновените потребители могат да получат полезна информация на родния си език.

7. ЗАКЛЮЧЕНИЕ

Семантичният анализатор и последващите обработки на заявката, правят системите с естественоезиков интерфейс по-полезни за потребителите от традиционните диалогови системи, които предоставят директни отговори, чрез прием на параметри от графичен потребителски интерфейс. Предлаганата архитектура на системата и алгоритъм за анализ, за разлика от останалите, не ограничават потребителя до избор на думи и синтаксис. Този подход преодолява разликата между случайните потребители и компютърните системи.

Общата цел на бъдещото развитие е практическото използване на случайни потребители да използват бази данни и информационни системи със собствен език.

ЛИТЕРАТУРА

- Жекова М. & Тотков Г. (2019). Фреймов модел за представяне на семантични роли и процеси за изграждане на естествено-езиков интерфейс, XII Национална конференция „Образованието и изследванията в информационното общество”, стр.42-52, Пловдив 30-31 май 2019.
- Жекова М. & Тотков Г. (2019). Метод за генериране на SQL заявки към база данни, зададени с текст на български език, Юбилейната научна сесия „Дни на науката – 2019” към Съюза на учените в България, 31 октомври – 1 ноември 2019 г.
- Жекова М. & Тотков Г. (2019). Методология за създаване на система от шаблони за съответствие във въпрос-отговор система с естественоезиков интерфейс, Национален младежки форум пролет, 2019 „Наука, технологии, иновации, бизнес” 30-31 май 2019 Пловдив
- Колковска С. (2005). Модели на концептуално-семантичните отношения между термините в специфичен (химически) текст с оглед на автоматичното им разпознаване. Издателство "LiterNet", Варна.
- Осенова П. (2016). Граматическо моделиране на българския език (с оглед на автоматичната обработка на естествен език). Издателство Парадигма, София 2016.
- Стайкова К. (2015). Лингвистични и семантични ресурси при компютърно генериране и аотиране на български текстове. Дисертационен труд, София, 2015.
- Sowah E. (2018). Natural language processing in cooperative query answering databases (NLPiCQA). International Journal of Computer Science and Mobile Computing, Vol.7 Issue.11, November 2018.
- Sowah, E. (2018). Natural language processing in cooperative query answering databases (NLPiCQA). Computer Science and Technology. China
- Sun R., Jiang J., Yee Fan Tan, Hang Cui, Tat-Seng Chua, Min-Yen Kan. (2005). Using Syntactic and Semantic Relation Analysis in Question Answering. Proceedings of the Fourteenth Text REtrieval Conference, Maryland USA, November 15-18, 2005.
- Zhekova, M. & Totkov, G. (2019). Conceptual Frame Model For The Presentation Of The Concepts And Rules In Natural Language Interface For Database. IOP Conference Series: Materials Science and Engineering. Plovdiv, 2019.