Български език 63 (2016), 4, 7-36

ГЮНЕШ ЕРКАН, ДРАГОМИР РАДЕВ

# LEXRANK: ЛЕКСИКАЛНА ЦЕНТРАЛНОСТ НА ГРАФИ КАТО МЯРКА ЗА ЗНАЧИМОСТ ПРИ РЕЗЮМИРАНЕТО НА ТЕКСТ[1]

GÜNEŞ ERKAN, DRAGOMIR R. RADEV

# LEXRANK: GRAPH-BASED LEXICAL CENTRALITY AS SALIENCE IN TEXT SUMMARIZATION

(Abstract)

We introduce a stochastic graph-based method for computing relative importance of textual units for Natural Language Processing. We test the technique on the problem of Text Summarization (TS). Extractive TS relies on the concept of sentence salience to identify the most important sentences in a document or set of documents. Salience is typically defined in terms of the presence of particular important words or in terms of similarity to a centroid pseudo-sentence.

*Keywords:* automatic text summarisation, lexical centrality, graph-based methods

В статията се предлага нов подход, наречен LexRank, за изчисляване на важността на изречението въз основа на понятието за централност на собствения вектор в представени под формата на графи изречения. В този модел като матрица на съседство за представянето на изреченията като графи се използва матрицата на косинусово сходство (cosine similarity) между компонентите на изречението.

Предлаганата от нас система, използваща подхода LexRank, показва добри резултати при няколко отделни тестови задания по време на *Конференцията за разбиране на информация от документи* през 2004 г. (Document Understanding Conference, DUC 2004). В настоящата статия представяме подробен анализ на подхода, като го прилагаме към по-голям масив от данни, в който влизат и данни от предишни задания на DUC. Ще разгледаме няколко метода за изчисляване на централността чрез графи на близост.

Разглежданите резултати показват, че степенните методи (включително LexRank) в повечето случаи се представят по-добре както от центроидните методи, така и от други системи, участвали в DUC. Методът Lex Rank с праг (Lex Rank with threshold) се представя по-добре от другите сте-

пенни методи, включително и от непрекъснатия LexRank (continuous Lex Rank). Резултатите показват, че предлаганият от нас подход е сравнително устойчив към шум в данните, причинен от непрецизно тематично клъстеризиране на документите.

**1. Въведение**

През последните години изследователските усилия в областта на обработката на естествен език бяха поставени на сериозна математическа основа. За разрешаване на много от изследователските задачи, като например парсирането (Collins 1997), снемането на семантична многозначност (Yarowsky 1995) и автоматичното перифразиране (Barzilay 2003), успешно се прилагат надеждни статистически техники. Устойчивите методи, базирани на графи, също предизвикват засилен интерес например при клъстеризация на думи (Brew 2002) и присъединяване на предложни фрази (Toutanova 2004).

В настоящата статия се разширява представата за базираните на графи методи в областта на обработката на естествен език. Ще разгледаме как „случайното блуждаене“ (random walk) по изреченските графи може да допринесе за подобряване на резултатите при резюмиране на текст. Ще представим накратко и възможните приложения на подобни техники за решаването на други задачи за обработка на естествен език, като например класификацията на именувани обекти, присъединяването на предложни фрази и класификацията на текстове (например за идентифициране на спамови съобщения).

Резюмирането на текст е процесът на автоматично създаване на съкратена версия на даден текст (резюме), която дава обобщена информация за съдържанието на изходния текст.

Информацията в резюмето зависи от нуждите на потребителя. Тематично ориентираните резюмета вземат под внимание темата, която интересува потребителя, и извличат от текста информация, свързана с тази тема. От друга страна, обобщаващите резюмета се опитват да обхванат колкото е възможно повече информация от текста, като се стремят да запазят тематичната му ориентираност. В настоящата статия ще предложим подход за обобщаващо резюмиране чрез директно извличане, приложено върху множество документи, принадлежащи към една и съща, но неопределена тема.

При резюмирането чрез директно извличане (extractive summarization) се генерират резюмета, като се избира подмножество от изреченията в изходните документи, докато при резюмирането чрез абстрактно извличане (abstractive summarization) информацията от изходния текст се перифразира. Макар че резюметата, създавани от хора, обикновено не разчитат на директно извличане, по-голямата част от съвременните методи за автоматично резюмиране използват именно този подход. Резюмирането, използващо само директно извличане, често дава по-добри резултати в сравне-

ние с автоматичното резюмиране чрез абстрактно извличане. Това се дължи на факта, че проблемите при автоматичното резюмиране чрез абстрактно извличане – като например представянето на семантична информация, формулирането на заключения и генерирането на естествен език – са сравнително по-трудни в сравнение с подходите за използване на вече готови езикови данни, като например извличането на изречения.

Методите за резюмиране чрез абстрактно извличане все още не са достатъчно добре разработени. Съществуващите подходи често зависят от компонента за предварително извличане на информация, тъй като механично копират или съкращават резултата от извличането, за да получат резюме на текста (Witbrock 1999, Jing 2002, Knight 2000). SUMMONS (Radev 1998) е пример за приложение за резюмиране на множество документи, което извлича и комбинира информация от няколко източника, а след това я предава на компонента за генериране на естествен език, който изработва крайното резюме.

Ранните изследвания в областта на резюмирането чрез директно извличане прилагат прости евристични методи, използващи определени характеристики на изреченията – например позицията им в текста, общата честота на думите в тях или наличието на ключови фрази, определящи важността на изречението (Baxendale 1958, Edmundson 1969, Luhn 1958).

Често използвана мярка за оценка на важността на думите в изречението е мярката *idf* (Inverse Document Frequency), която се определя по формулата (Sparck-Jones 1972):

(1)

$$idf_i = \log\left(\frac{N}{n_i}\right)$$

където $N$ е общият брой на документите в масива от документи, а $n_i$ е броят на документите, в които се среща думата $i$. Например думите, за които е вероятно да се срещнат в почти всеки документ (като неопределителния член „a“ и определителния член „the“ в английски), имат *idf* стойности, близки до нулата, докато по-рядко срещани думи (например медицински термини, собствени имена) обикновено имат по-високи *idf* стойности.

По-усъвършенстваните подходи отчитат и връзките между изреченията или дискурсната структура, като използват синоними на думите или анафорични зависимости (Mani 1997, Barzilay 1999). Правят се опити за интегриране на машинно обучение в методите за резюмиране, като се прилагат все повече характеристики, и данните, до които имаме достъп, са все повече (Kupiec 1995, Lin 1999, Osborne 2002, Daume 2004).

Предлаганият в настоящата статия подход за резюмиране включва оценка на *централността* на всяко изречение в рамките на даден клъстер и извличане на най-важните изречения, които да бъдат включени в резю-

мето. Изследваме различни начини за дефиниране на принципа на лексикалната централност в резюмирането на множество документи като мярка за централност въз основа на лексикалните свойства на изреченията.

В раздел 2. представяме центроидния метод за резюмиране, който е добре познат метод за оценка на централността на изречението. Въвеждаме три нови мерки за централност: степенна централност (Degree), Lex Rank с праг и непрекъснат LexRank, вдъхновени от понятието за *престиж* в социалните мрежи. Предлагаме представяне на клъстера от документи във вид на граф, в който върховете представляват изреченията, а ребрата са определени от отношението на близост между двойките изречения. Това представяне ни дава възможност да използваме някои евристични методи за централността, дефинирани за графи.

Сравняваме предложените от нас нови методи с центроидния метод за резюмиране с помощта на системата за резюмиране MEAD и показваме, че формулираните нови характеристики в повечето случаи дават по-добри резултати от центроидния метод. В експериментите използваме тестови данни от заданията за резюмиране по време на *Конференцията за разбиране на информация от документи* (DUC) през 2003 г. и 2004 г., като въз основа на тях сравняваме предлаганата от нас система с други системи за резюмиране, както и с резюмиране от човек.

## 2. Централност на изречението и центроидно резюмиране

Резюмирането чрез директно извличане включва избор на подмножество от изречения от изходните документи. Този процес включва идентифициране на най-централните изречения в клъстер (от множество документи), които дават необходима и достатъчна информация за главната тема в клъстера. Централността на дадено изречение често се определя въз основа на централността на думите в него. Един от разпространените подходи за оценяване на централността на дадена дума включва разглеждане на центроида на клъстера от документи във векторно пространство.

Центроидът на даден клъстер е псевдодокумент, който се състои от думите, които имат стойности на *tf* x *idf* над предварително определен праг, където *tf* е честотата на дадена дума в клъстера, а *idf* обикновено се изчислява върху много по-голям масив от жанрово близки документи. При центроидното резюмиране (Radev 2000b) изреченията, съдържащи повече думи от центроида на клъстера, се оценяват като *централни* (Алгоритъм 1).

Това е мярка, която показва колко близко е изречението до центроида на клъстера. Центроидното резюмиране дава добри резултати и е в основата на първата уеб базирана система за резюмиране на множество документи (Radev 2001)[2].

Вход: Масив S от n изречения, праг на синусово сходство t
Изход: Масив *C* от стойностите на центроида
Hash *WordHash*;

```
Array C;
/* изчисли tf x idf за всяка дума */
for i from 1 to n do
        foreach word w of S[i] do
        WordHash{w}{"tfidf"}=WordHash{w}{"tfidf"}+idf{w};
        end
end
/* конструирай центроида на клъстера */
/* с думите, които са над прага */
foreach word w of WordHash do
        if WordHash{w}{"tfidf"}>t then
        WordHash{w}{"centroid"}=WordHash{w}{"tfidf"};
        end
        else
                WordHash{w}{"centroid"}=0;
        end
end
/* изчисли tf x idf за всяка дума */
for i from 1 to n do
        C[i] = 0;
        foreach word w of S[i] do
                C[i] = C[i] + WordHash{w}{"centroid"};
        end
end
return C;
```

Алгоритъм 1. *Изчисляване на стойностите на центроида*

### 3. Значимост на изречението въз основа на централността

В този раздел предлагаме още няколко критерия за оценка на значимостта на изречението. Всички предложени подходи се основават на концепцията за *престиж*[3] в социалните мрежи, която се използва често в областта на компютърните мрежи и извличането на информация. Социалната мрежа представлява карта на взаимоотношенията между взаимодействащи си единици (например хора, организации, компютри). Социалните мрежи се представят като графи, в които върховете представляват единиците, а ребрата – взаимоотношенията между тях.

В този смисъл клъстерът от документи може да се разглежда като мрежа от изречения с връзки между тях. Някои изречения имат по-голяма близост помежду си, докато други съдържат по-малко обща информация с останалата част от изреченията. Приемаме хипотезата, че изреченията, по-

казващи подобие с голяма част от другите изречения в клъстера, са по-централни (или *значими*, англ. *salient*) по отношение на темата. В това определение за централност има две неща, които се нуждаят от разяснение. Първото е как да дефинираме близостта между две изречения. Второто е как да изчислим централността на дадено изречение въз основа на близостта му с други изречения.

За да дефинираме близостта, използваме така наречения модел „торба с думи“ (bag-of-words), при който всяко изречение се представя като *N*-мерен вектор, където *N* е броят на всички възможни думи от езика, на който се превежда. За всяка дума от изречението стойността на съответното измерение във векторното представяне е произведението от броя на срещанията на думата в изречението и *idf* на думата.

Близостта между две изречения се изчислява като косинуса между съответните вектори:

(2)

$$idf-\mathrm{mod}ified-\cos ine(x,y)\equiv \frac{\sum_{w\in x,y} tf_{w,x}tf_{w,y}\left(idf_w\right)^2}{\sqrt{\sum_{x_i\in x}(tf_{x_i}x^{idf_{x_i}})^2}\,x\,\sqrt{\sum_{y_i\in y}(tf_{y_i}y^{idf_{y_i}})^2}}$$

където $tf_{w,s}$ е броят на срещанията на думата *w* в изречението *s*.

Клъстерът от документи може да бъде представен като матрица, в която всеки елемент е стойността на косинусово сходство между съответната двойка изречения. Фигура 1 показва подмножество на клъстер, използван на DUC 2004, и съответната матрица на косинусово сходство. Изречението с идентификационен код (ID) *dXsY* означава изречението *Y* в документа *X*.

| S | ID | Text |
|---|---|---|
| 1 | dlsl | Iraqi Vice President Taha Yassin Ramadan announced today, Sunday, that Iraq refuses to back down from its decision to stop cooperating with disarmament inspectors before its demands are met. |
| 2 | d2sl | Iraqi Vice president Taha Yassin Ramadan announced today, Thursday, that Iraq rejects cooperating with the United Nations except on the issue of lifting the blockade imposed upon it since the year 1990. |
| 3 | d2s2 | Ramadan told reporters in Baghdad that "Iraq cannot deal positively with whoever represents the Security Council unless there was a clear stance on the issue of lifting the blockade off of it. |
| 4 | d2s3 | Baghdad had decided late last October to completely cease cooperating with the inspectors of the United Nations Special Commission (UNSCOM), in charge of disarming Iraq's weapons, and whose work became very limited since the fifth of August, and announced it wfll not resume its cooperation with the Commission even if it were subjected to a military operation. |

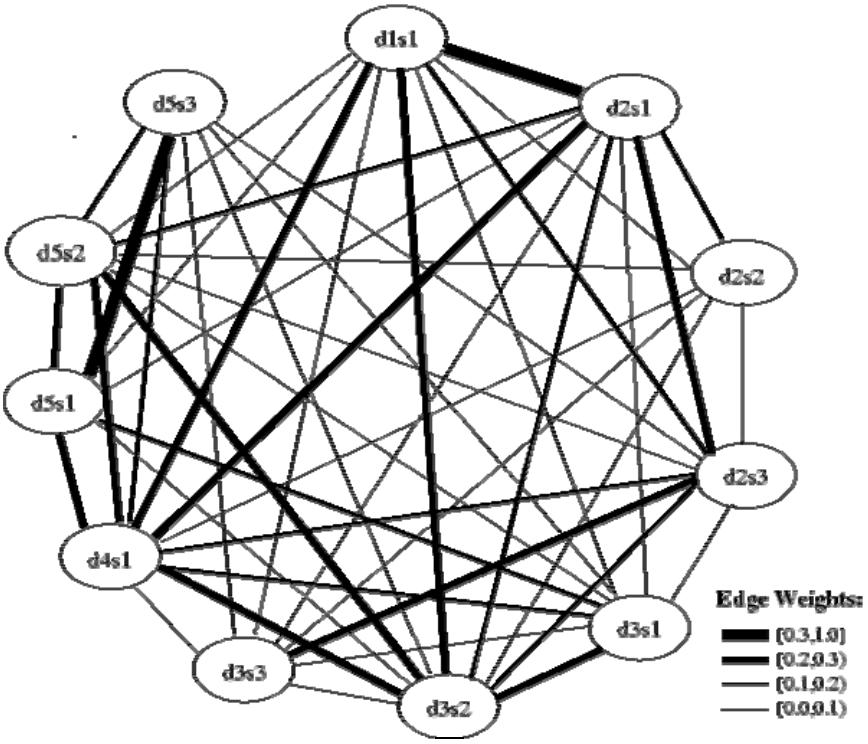
| | | |
|---|---|---|
| 5 | d3sl | The Russian Foreign Minister, Igor Ivanov, warned today, Wednesday against using force against Iraq, which will destroy, according to him, seven years of difficult diplomatic work and will complicate the regional situation in the area. |
| 6 | d3s2 | Ivanov contended that carrying out air strikes against Iraq, who refuses to cooperate with the United Nations inspectors, "will end the tremendous work achieved by the international group during the past seven years and will complicate the situation in the region." |
| 7 | d3s3 | Nevertheless, Ivanov stressed that Baghdad must resume working with the Special Commission in charge of disarming the Iraqi weapons of mass destruction (UNSCOM). |
| 8 | d4sl | The Special Representative of the United Nations Secretary-General in Baghdad, Prakash Shah, announced today, Wednesday, after meeting with the Iraqi Deputy Prime Minister Tariq Aziz, that Iraq refuses to back down from its decision to cut off cooperation with the disarmament inspectors. |
| 9 | d5sl | British Prime Minister Tony Blair said today, Sunday, that the crisis between the international community and Iraq "did not end" and that Britain is still "ready, prepared, and able to strike Iraq." |
| 10 | d5s2 | In a gathering with the press held al the Prime Minister's office, Blair contended that the crisis with Iraq "will not end until Iraq has absolutely and unconditionally respected its commitments" towards the United Nations. |
| 11 | d5s3 | A spokesman for Tony Blair had indicated that the British Prime Minister gave permission to British Air Force Tornado planes stationed in Kuwait to join the aerial bombardment against Iraq. |

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 1.00 | 0.45 | 0.02 | 0.17 | 0.03 | 0.22 | 0.03 | 0.28 | 0.06 | 0.06 | 0.00 |
| 2 | 0.45 | 1.00 | 0.16 | 0.27 | 0.03 | 0.19 | 0.03 | 0.21 | 0.03 | 0.15 | 0.00 |
| 3 | 0.02 | 0.16 | 1.00 | 0.03 | 0.00 | 0.01 | 0.03 | 0.04 | 0.00 | 0.01 | 0.00 |
| 4 | 0.17 | 0.27 | 0.03 | 1.00 | 0.01 | 0.16 | 0.28 | 0.17 | 0.00 | 0.09 | 0.01 |
| 5 | 0.03 | 0.03 | 0.00 | 0.01 | 1.00 | 0.29 | 0.05 | 0.15 | 0.20 | 0.04 | 0.18 |
| 6 | 0.22 | 0.19 | 0.01 | 0.16 | 0.29 | 1.00 | 0.05 | 0.29 | 0.04 | 0.20 | 0.03 |
| 7 | 0.03 | 0.03 | 0.03 | 0.28 | 0.05 | 0.05 | 1.00 | 0.06 | 0.00 | 0.00 | 0.01 |
| 8 | 0.28 | 0.21 | 0.04 | 0.17 | 0.15 | 0.29 | 0.06 | 1.00 | 0.25 | 0.20 | 0.17 |
| 9 | 0.06 | 0.03 | 0.00 | 0.00 | 0.20 | 0.04 | 0.00 | 0.25 | 1.00 | 0.26 | 0.38 |
| 10 | 0.06 | 0.15 | 0.01 | 0.09 | 0.04 | 0.20 | 0.00 | 0.20 | 0.26 | 1.00 | 0.12 |
| 11 | 0.00 | 0.00 | 0.00 | 0.01 | 0.18 | 0.03 | 0.01 | 0.17 | 0.38 | 0.12 | 1.00 |

Фигура 1. *Вътреизреченско косинусово сходство в подмножество на клъстер d1003t от DUC 2004. Източник: Agence France Presse (AFP) Arabic Newswire (1998). Ръчно преведени на английски*

Тази матрица може да бъде представена и като тегловен граф, в който всяко ребро показва косинусовото сходство между двойка изречения (Фигура 2). В следващите раздели ще представим няколко начина за изчисля-

ване на централност на изречение с помощта на матрицата на косинусово сходство и съответното представяне във вид на граф.



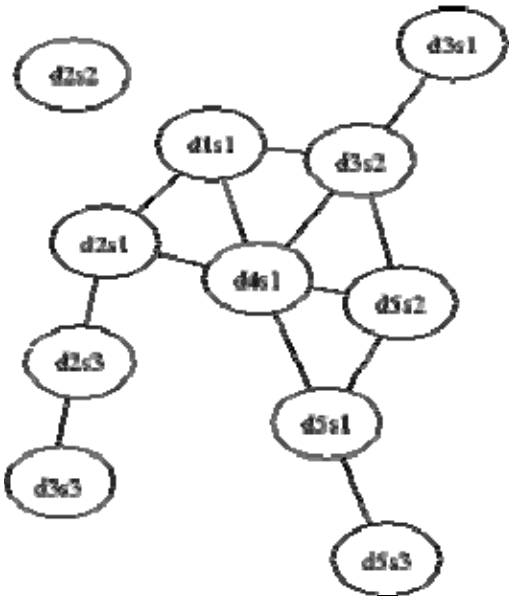

Фигура 2. *Тегловен граф с косинусово сходство за клъстера от Фигура 1*

### 3.1. Централността като степен

В клъстер от свързани документи се очаква много от изреченията да бъдат относително сходни, тъй като всички са на една и съща тема. Това е илюстрирано на Фигура 1, където повечето от стойностите в матрицата на близост са различни от нула. Тъй като се интересуваме от *значителна* близост, можем да елиминираме някои от по-ниските стойности в матрицата, като определим такъв праг, че клъстерът да се разглежда като (ненасочен) граф, в който всяко изречение е връх, а изреченията със значителна близост помежду си са свързани.

Фигура 3 представя графите, които съответстват на матриците на съседство, които са резултат от допускането, че двойка изречения с близост съответно над 0.1, 0.2 и 0.3 от Фигура 1 са сходни помежду си. Трябва да се отбележи, че всеки връх е свързан и сам със себе си, тъй като всяко изречение е тривиално сходно със себе си. За по-ясна визуализация не включваме в представянето тези връзки, но в хода на аргументацията в следващите раздели наличието им се предполага.


d3s3
d2s3
d3s2
d3s1
d1s1
d4s1
d5s1
d2s1
d2s2
d5s2
d5s3


d3s1
d2s2
d1s1
d3s2
d2s1
d4s1
d5s2
d2s3
d5s1
d3s3
d5s3


Фигура 3. *Графи на близост, съответстващи на прагове 0.1, 0.2 и 0.3, за клъстера от Фигура 1*

Един сравнително елементарен начин за оценяване на централността на дадено изречение в графите на Фигура 3 е да се преброят близките до него изречения. Дефинираме *степенната централност* на изречението като степента на съответния връх в графа на близост. Както се вижда от Таблица 1, избраният праг на косинусово сходство оказва голямо влияние върху интерпретацията на централността. Ако определим прекалено ниски прагове, можем погрешно да включим и изречения с ниска близост, а ако праговете са прекалено високи, има риск да пропуснем много от близко свързаните изречения в клъстера.

Таблица 1. *Степенна централност за графите на Фигура 3. Изречението d4s1 е най-централно за прагове 0.1 и 0.2.*

| ID | Степен (0.1) | Степен (0.2) | Степен (0.3) |
|---|---|---|---|
| d1s1 | 5 | 4 | 2 |
| d2s1 | 7 | 4 | 2 |
| d2s2 | 2 | 1 | 1 |
| d2s3 | 6 | 3 | 1 |
| d3s1 | 5 | 2 | 1 |
| d3s2 | 7 | 5 | 1 |
| d3s3 | 2 | 2 | 1 |
| d4s1 | 9 | 6 | 1 |
| d5s1 | 5 | 4 | 2 |
| d5s2 | 6 | 4 | 1 |
| d5s3 | 5 | 2 | 2 |

### 3.2. Централността като собствен вектор и LexRank

Когато изчисляваме степенната централност, приемаме всяко ребро като „глас“ за определяне на стойността на общата централност на съответния връх. Това е демократичен метод, при който всички „гласове“ имат еднакво тегло. В много социални мрежи обаче не всички връзки се възприемат като еднакво важни. Да вземем за пример социална мрежа от хора, които са приятели (т.е. свързани са помежду си с отношение на прия-

телство). Престижът на даден човек зависи не само от това колко приятели има, но и *кои* са приятелите му.

Същата идея може да се приложи и в резюмирането чрез директно извличане. Използването на степенната централност може да има отрицателен ефект върху качеството на резюметата, тъй като няколко недобре преведени изречения може да „гласуват“ едно за друго, с което да повишат своята централност. Като пример нека разгледаме клъстер с наличен шум, в който всички документи са свързани помежду си, освен един, който е на по-различна тематика. Очевидно не бихме искали някое от изреченията от различния документ да бъде включено в резюмето на клъстера.

Да предположим обаче, че различният документ съдържа изречения с висок *престиж*, изчислен само въз основа на „гласове“ от същия документ. Тези изречения ще получат изкуствено високи стойности на централност на базата само на локални „гласове“ от ограничено множество от изречения. Тази ситуация може да се избегне, ако в изчислението се вземе предвид откъде е „гласът“ и се отчита и централността на гласуващите върхове.

Тази идея може да се формулира по следния начин: приемаме, че всеки връх има стойност за централност и я разпределя между своите съседи. Формулировката може да се изрази с уравнението

(3)

$$p(u) = \sum_{v \in adj[u]} \frac{p(v)}{deg(v)}$$

където $p(u)$ е централността на върха $u$, $adj[u]$ е множеството от върхове, свързани с $u$, а $deg[v]$ е степента на върха $v$. Еквивалентното представяне на горното уравнение в матрична форма е:

(4)

$$\mathrm{p} = \mathrm{B}^{\mathrm{T}}\mathrm{p}$$

или

(5)

$$\mathrm{p}^{\mathrm{T}}\mathrm{B} = \mathrm{p}^{\mathrm{T}},$$

където матрицата **B** е получена от матрицата на съседство на графа на близост, като всеки елемент е разделен на сумата на елементите от съответния ред:

(6)

$$\boldsymbol{B}(i,j) = \frac{\boldsymbol{A}(i,j)}{\sum_k \boldsymbol{A}(i,k)}$$

Трябва да се отбележи, че сумата на даден ред е равна на степента на съответния връх. Тъй като всяко изречение е близко поне на себе си, сумата на всеки ред е различна от нула. Уравнение (5) гласи, че $\mathbf{p}^T$ е левият собствен вектор на матрицата **B** със съответна собствена стойност 1. За да се гарантира, че такъв собствен вектор съществува и може да бъде еднозначно определен и изчислен, се нуждаем от някои математически постановки.

Стохастичната матрица **X** е преходна матрица на Марковска верига. Елементът $\mathbf{X}(i,j)$ на стохастичната матрица определя вероятността за преход от състояние $i$ в състояние $j$ в съответната Марковска верига. Според аксиомите от теорията на вероятностите сумата на всеки ред на стохастичната матрица трябва да е равна на 1. $\mathbf{X}^n(i,j)$ представлява вероятността за достигане от състояние $i$ до състояние $j$ с $n$ на брой прехода. Марковска верига със стохастична матрица X клони към стационарно разпределение, ако

(7)

$$\lim_{n \to \infty} \boldsymbol{X}^n = \mathbf{1}^T \boldsymbol{r}$$

където $\mathbf{1} = (1,1,..., 1)$, а векторът **r** се нарича стационарно разпределение на Марковската верига.

За интуитивната интерпретация на стационарното разпределение може да се използва понятието за „случайно блуждаене“ (random walk). Всеки елемент на вектора **r** дава асимптотичната вероятност за съответното крайно състояние в дългосрочен план, независимо от началното състояние. Марковската верига е регулярна, ако от всяко състояние може да се достигне от произволно друго състояние, т.е. за всички $i,j$ съществува такова $n$, за което $\mathbf{X}^n(i,j) \neq 0$. Марковската верига е апериодична, ако за всяко $i$ имаме: $gcd\{n : \mathbf{X}^n(i,i) > 0\} = 1$. По силата на Теоремата на Перон-Фробениус (Seneta 1981) всяка регулярна и апериодична Марковска верига клони към единствено стационарно разпределение. Ако Марковската верига има нерегулярни или периодични компоненти, при случайно блуждаене може да се образува цикъл в тях и никога да не се стигне до другите части на графа.

Тъй като матрицата на близост **B** в (4) удовлетворява условията за стохастична матрица, можем да я разглеждаме като Марковска верига. Векторът на централност **P** съответства на стационарното разпределение на **B**. Трябва обаче да се уверим, че матрицата на близост винаги е регулярна и апериодична. За решаването на този проблем Пейдж, Брин, Мотвани и Виноград (Page 1998) предлагат да се запазят някои от ниските вероятности, за да може да се достигне до всеки връх в графа.

По този начин при случайното блуждаене може да се „избяга“ от периодични или изолирани компоненти, което прави графа регулярен и апериодичен. Ако зададем равномерна вероятност за преход към всеки връх в

графа, се получава следната модифицирана версия на (3), известна като PageRank:

(8)

$$p(u) = \frac{d}{N} + (1-d) \sum_{v \in adj[u]} \frac{p(v)}{deg(v)}$$

където $N$ е общият брой върхове в графа, а $d$ е „коефициент на затихване“ (damping factor), който обикновено се избира в интервала [0.1, 0.2] (Brin 1998). Уравнение 8 може да бъде представено като матрица

(9)

$$p = [dU + (1 - d)B]^T p$$

където **U** е квадратна матрица, в която всички елементи са равни на *1/N*.

Ядрото на прехода [$d$**U** + (1 – $d$)**B**] на получената Марковска верига е комбинация от двете ядра **U** и **B**. При случайно блуждаене върху тази Марковска верига се избира едно от съседните състояния на текущото състояние с вероятност 1 – $d$ или се скача в произволно друго състояние в графа, включително текущото състояние, с вероятност $d$. Първоначално формулата за PageRank е предложена за изчисляване на престижа на уебстраници и все още е сред основните механизми, използвани в търсачката на Google.

Сходимостта на Марковските вериги ни дава и прост итеративен алгоритъм, наречен степенен метод, за изчисляване на стационарното разпределение (Алгоритъм 2). Алгоритъмът започва с равномерно разпределение. На всяка итерация собственият вектор се актуализира, като се умножава с транспонираната итерация на стохастичната матрица. Тъй като Марковската верига е регулярна и апериодична, е гарантирано, че алгоритъмът ще завърши.

Вход: Стохастична, регулярна и апериодична матрица M
Вход: размер на матрицата $N$, толерантност за грешка ε
Изход: собствен вектор **p**

$\mathbf{P}_0 = \frac{1}{N} * \mathbf{1}$;
t=0;
**repeat**
    t = t+1;
    $\mathbf{p}_t = \mathbf{M}^T \mathbf{p}_{t-1}$;
    $\delta = \| \mathbf{p}_t - \mathbf{p}_{t-1} \|$
**until** $\delta < \varepsilon$;
**return** $\mathbf{p}_t$;

Алгоритъм 2. *Степенен метод за изчисляване на стационарното разпределение на Марковска верига*

За разлика от оригиналния метод PageRank графът на близост при изреченията е ненасочен, тъй като косинусовото сходство е симетрична релация. Това обаче не оказва влияние при изчисляването на стационарното разпределение. Новата мярка за близост на изречения наричаме *лексикален PageRank* или *LexRank*. Алгоритъм 3 показва как се изчисляват стойностите на LexRank за дадено множество изречения.

Трябва да се има предвид, че стойностите на степенната централност също се изчисляват (в масива *Degree*) като страничен продукт на алгоритъма. Таблица 2 показва стойностите на LexRank за графите на Фигура 3, с коефициент на затихване 0.85. За сравнение, центроидните стойности за всяко изречение също са показани в таблицата. Всички числа са нормализирани, така че изречението с най-висок рейтинг получава стойност 1. От данните е видно, че избраният праг оказва влияние при ранкирането на някои изречения по LexRank.

```
Вход: Масив S от n изречения, праг на косинусово сходство t
Изход: Масив L от стойности на LexRank
Array CosineMatrix[n][n];
Array Degree[n];
Array L[n];
for i from 1 to n do
    for j from 1 to n do
        CosineMatrix[i][j] = idf-modified-cosine(S[i],S[j]);
        if CosineMatrix[i][j] > t then
            CosineMatrix[i][j] = 1;
            Degree[i]++;
        end
        else
            CosineMatrix[i][j]=0;
        end
    end
end
for i from 1 to n do
    for j from 1 to n do
        CosineMatrix[i][j] = CosineMatrix[i][j]/Degree[i];
    end
end
L = PowerMethod(CosineMatrix, n, ε);
return L;
```

Алгоритъм 3. *Изчисляване на стойностите на LexRank*

Таблица 2. *Стойностите за LexRank за графите на Фигура 3. Стойностите са нормализирани, така че най-голямата стойност във всяка колона е 1. Изречение d4s1 е най-централно с прагове 0.1 и 0.2*

| ID | LR (0.1) | LR (0.2) | LR (0.3) | Centroid |
|---|---|---|---|---|
| d1s1 | 0.6007 | 0.6944 | 1.000 | 0.7209 |
| d2s1 | 0.8466 | 0.7317 | 1.000 | 0.7249 |
| d2s2 | 0.3491 | 0.6773 | 1.000 | 0.1356 |
| d2s3 | 0.7520 | 0.6550 | 1.000 | 0.5694 |
| d3s1 | 0.5907 | 0.4344 | 1.000 | 0.6331 |
| d3s2 | 0.7993 | 0.8718 | 1.000 | 0.7972 |
| d3s3 | 0.3548 | 0.4993 | 1.000 | 0.3328 |
| d4s1 | 1.0000 | 1.0000 | 1.000 | 0.9414 |
| d5s1 | 0.5921 | 0.7399 | 1.000 | 0.9580 |
| d5s2 | 0.6910 | 0.6967 | 1.000 | 1.0000 |
| d5s3 | 0.5921 | 0.4501 | 1.000 | 0.7902 |

### 3.3. Непрекъснат LexRank

Графите на близост за изчисляване на степенната централност и Lex Rank не използват тегла. Това се дължи на двоичната дискретизация с избран праг, която прилагаме върху косинусовата матрица. Както при всички операции на дискретизация, и тук се стига до загуба на информация. LexRank може да се подобри, като използваме *силата* на връзките на близост. Ако при конструиране на графа на близост използваме директно косинусовите стойности, графът ще е много по-плътен, но и тегловен (Фигура 2). Можем да нормализираме сумите от редовете на съответната преходна матрица, така че да получим стохастична матрица. Полученото уравнение е модифицирана версия на LexRank за тегловни графи:

(10)

$$p(u) = \frac{d}{N} + (1-d) \sum_{v \in adj[u]} \frac{idf - modified - cosine(u,v)}{\sum idf - modified - cosine(z,v)} p(v)$$

По този начин, като изчисляваме LexRank за дадено изречение, умножаваме стойностите на LexRank на свързваните с него изречения по теглото на връзките. Теглата са нормализирани от сумите по редовете и коефициентът на затихване *d* е добавен, за да се осигури сходимост на метода.

### 3.4. Централност или центроид

Методът, основан на централността на графите, има няколко предимства пред центроидния метод. На първо място той отчита информационните връзки между изреченията. Ако информацията в дадено изречение включва информацията от друго изречение в клъстера, за резюмето ще се предпочете онова изречение, което съдържа повече информация. Степента на даден връх в графа на косинусовите сходства е показател за това колко обща информация съдържа изречението спрямо другите изречения.

Изречение d4s1 от Фигура 1 получава най-висока стойност, защото включва почти изцяло информацията от първите две изречения в клъстера и съдържа информация и от останалите изречения. Друго предимство на предложения от нас подход е, че той предотвратява приписването на неестествено високи *idf* стойности на изречения, които не са свързани с темата. Макар честотата на думите да се взема под внимание при изчисляването на стойностите на центроида, изречение, съдържащо много редки думи с високи *idf* стойности, може да получи висока стойност дори ако думите не се срещат другаде в клъстера.

## 4. Експерименти

В този раздел се представят използваните данни, показателят за оценка и системата за резюмиране, приложени в експериментите.

### 4.1. Данни и метод за оценка

В проведените експерименти използваме данните от DUC 2003 и DUC 2004. Задание 2 както в DUC 2003, така и в DUC 2004 включва обобщаващо резюмиране на клъстери от новини. В данните от DUC 2003 се съдържат 30 клъстера, а в тези от DUC 2004 – 50 клъстера. В допълнение използваме още два масива от данни от Задание 4 на DUC 2004, което включва междуезиково обобщаващо резюмиране. Първите данни (Задание 4a) включват 24 клъстера от новинарски текстове, които са резултат от машинен превод от арабски на английски. Втората група данни (Задание 4b) са човешки преводи на текстовете от същите клъстери. Всички данни са на английски език.

Използваме новия показател за оценка на автоматичното резюмиране ROUGE[4], приложен за първи път в DUC 2004. ROUGE се основава на оценка на покритието (recall) за резюмета с фиксирана дължина чрез статистически анализ на n-грами. При ROUGE се изчисляват отделни оценки за 1-, 2-, 3- и 4-грами за съвпадение между образцовите резюмета и оце-

няваното резюме. Най-близки резултати до човешката оценка се постигат с ROUGE върху униграми (ROUGE-1) (Lin 2003).

Разполагаме с 10 човешки оценки за Задание 2 на DUC 2003; 8 – за Задание 2 на DUC 2004; и 4 – за Задание 4 на DUC 2004. За произволен клъстер подмножество от точно четири човешки оценки генерира образцово резюме. ROUGE изисква ограничение на дължината на резюметата, за да бъде направена коректна оценка. За да бъдат изпълнени спецификациите на DUC 2004 и за да могат да се сравнят резултатите от нашата система с ръчно направените резюмета, както и с резултатите на другите системи, участвали в DUC, бяха създадени резюмета от по 665 байта за всеки клъстер и бяха изчислени ROUGE стойностите спрямо ръчно направените резюмета.

### 4.2. Системата за резюмиране MEAD

Разработените методи бяха внедрени в системата за резюмиране MEAD[5] (Radev 2001) – публично достъпна система с инструменти за обобщаващо резюмиране чрез директно извличане. Макар че MEAD е центроидна система за резюмиране, тя може да бъде разширена, за да се приложат и други методи.

Системата за резюмиране MEAD се състои от три компонента. На първия етап – т.нар. *извличане на характеристики* – всяко изречение от документа (или клъстера от документи) се представя във векторна форма, като се използват дефинирани от потребителя характеристики. На втория етап векторът се конвертира в скаларна стойност с помощта на *комбинатор*. Комбинаторът извежда линейна комбинация от характеристиките въз основа на предварително определени тегла. На последния етап, известен като *преранкиране*, оценките на изреченията, включени в съответните двойки, се коригират нагоре или надолу в зависимост от вида на връзката между изреченията. Преранкирането намалява оценката на изречения, подобни на вече включените в резюмето изречения, с цел да се постигне по-добро информационно покритие.

Трите стандартни характеристики, включени в дистрибуцията на MEAD, са центроид (Centroid), позиция (Position) и дължина (Length). Позицията е нормализираната стойност на позицията на дадено изречение в документа, така че първото изречение в документа получава максимална стойност 1, а последното изречение получава стойност 0. Дължината (Length) не е същинска характеристика, а праг, така че да се изключат изречения с дължина под прага. В MEAD са включени няколко модула за преранкиране – единият работи на основата на максимално гранично съответствие (Maximal Marginal Relevanc, MMR) (Carbonell 1998), а по подразбиране системата ползва модула, основан на междуизреченското информационно съотнасяне (Cross-Sentence Informational Subsumption, CSIS) (Radev 2000a). Всички експерименти от раздел 5 използват модула CSIS.

MEAD използва комбинация от три компонента: (а) командните редове за всички характеристики; (б) формулата за превръщане на вектора в скалар и (в) команден ред за преранкирането. Фигура 4 представя примерна стратегия, използваща трите характеристики по подразбиране (центроид, позиция, дължина) и новата LexRank, използвана в експериментите ни. Имплементацията на LexRank изисква като аргумент минимален праг за косинусово сходство (в примера 0.2).

Числото до всяка характеристика показва относителното тегло на характеристиката (с изключение на LengthCutoff, където 9 е прагът, използван за подбор на изречения въз основа на броя думи в него). В примера е използвано MMR преранкиране, базирано на дума и с праг на косинусово сходство 0.5. Накрая *enidf* посочва файла, съдържащ списък с предварително изчислените стойности за *idf* за думите в английски.

```
feature LexRank LexRank.pl 0.2
Centroid 1 Position 1 LengthCutoff 9 LexRank 1
mmr-reranker-word.pl 0.5 MEAD-cosine enidf
```

Фигура 4. *Пример от MEAD*

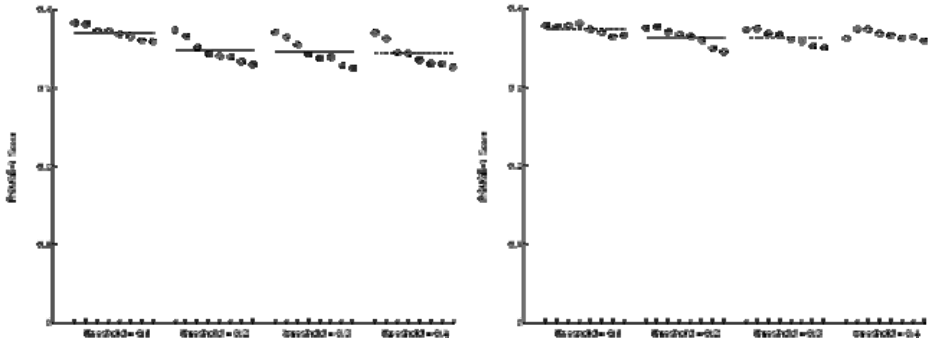
**5. Резултати и дискусия**

Следващите раздели представят резултатите от извършените експерименти върху данните от DUC с различни имплементации за изчисление на централността в графи. Въведохме степенната централност, LexRank с праг и непрекъснатия LexRank като отделни характеристики в MEAD. Всички стойности на характеристиките се нормализират, така че най-високата стойност да е 1, а най-ниската – 0.

Във всички експерименти използваме характеристиките дължина и позиция от MEAD като допълнителни евристични характеристики към стойностите за централност. Прагът на стойността за дължина е 9, т.е. всички изречения с дължина, по-малка от 9 думи, се изключват. Теглото на характеристиката позиция е фиксирано на 1 за всички етапи на експериментите. Освен тези две евристични характеристики използваме самостоятелно и всяка една от характеристиките на централност, без да ги комбинираме с други методи, за да направим по-добро сравнение. За всяка отделна характеристика на централност правим 8 отделни пускания в MEAD, като поставяме тегло на характеристиката съответно 0.5, 1.0, 1.5, 2.0, 2.5, 3.0, 5.0 и 10.0.

**5.1. Ефект от прага върху степенната централност и LexRank**

До момента показахме, че много високите прагове може да доведат до загуба на почти цялата информация в матрицата на близост (Фигура 3). В подкрепа на това твърдение прилагаме степенна централност и LexRank с различни прагове. Фигура 5 показва ефекта на прага при степенната централност и LexRank върху данните от Задание 2 на DUC 2004. Експериментираме с четири различни прага: 0.1, 0.2, 0.3 и 0.4.



(а) (б)

Фигура 5. *Стойности на ROUGE-1 за: (a) степенна централност; (b) LexRank с различни прагове върху данните от Задание 2 на DUC 2004*

Осемте точки за всеки праг отразяват резултатите при използване на една и съща характеристика с осем различни тегла, както вече беше споменато. Средната стойност при осемте различни експеримента е показана като хоризонтална линия. На графиките се вижда, че най-ниският праг – 0.1, генерира най-добри резюмета. Това означава, че загубата на информация при по-високите прагове е достатъчно значима, за да доведе до по-лоши стойности на ROUGE. Спадът в стойността на ROUGE между праг 0.1 и праг 0.2 е по-значителен при степенната централност.

Този ефект на прага показва, че новите ни методи всъщност работят добре при резюмиране с директно извличане. Колкото по-висок е прагът, толкова по-малко информативни или дори по-подвеждащи графи на близост получаваме. В най-крайния случай, т.е. при много висок праг, няма да има ребра в графа, така че степенната централност и LexRank няма да ни бъдат от полза.

### 5.2. Сравнение на методите, основани на централност

Таблица 3 показва стойностите на ROUGE в извършените експерименти съответно от Задание 2 от DUC 2003, Задание 2 от DUC 2004, Задание 4а от DUC 2004 и Задание 4б от DUC 2004. Показваме минималните, максималните и средните стойности ROUGE-1 за осемте експеримента, които сме провели за всеки метод с осем различни тегла на характеристиките.

Таблица 3. *Стойности за ROUGE-1 при различни параметри в MEAD върху данни от DUC 2003 и 2004*

| 2003 – Задание 2 | | | | 2004 – Задание 2 | | | |
|---|---|---|---|---|---|---|---|
| | Min | max | средно | | min | max | средно |
| Центроид | 0.3523 | 0.3713 | 0.3624 | Центроид | 0.3580 | 0.3767 | 0.3670 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Степен (t=0.1) | 0.3566 | 0.3640 | 0.3595 | Степен (t=0.1) | 0.3590 | 0.3830 | 0.3707 |
| LexRank (t=0.1) | 0.3610 | 0.3726 | 0.3666 | LexRank (t=0.1) | 0.3646 | 0.3808 | 0.3736 |
| Непре-къснат LexRank | 0.3594 | 0.3700 | 0.3646 | Непре-къснат LexRank | 0.3617 | 0.3826 | 0.3758 |

(а) (б)

**Базови методи:**

Random (със случайно избиране): 0.3261
Lead-based: 0.3575

**Базови методи:**

Random (със случайно избиране): 0.3238
Lead-based: 0.3686

| Задание 4а | | | | 2004 – Задание 4б | | | |
|---|---|---|---|---|---|---|---|
| | min | max | средно | | min | max | средно |
| Центроид | 0.3768 | 0.3901 | 0.3826 | Центроид | 0.3760 | 0.3962 | 0.4034 |
| Степен (t=0.1) | 0.3863 | 0.4027 | 0.3928 | Степен (t=0.1) | 0.3801 | 0.4147 | 0.4026 |
| LexRank (t=0.1) | 0.3931 | 0.4038 | 0.3974 | LexRank (t=0.1) | 0.3837 | 0.4167 | 0.4052 |
| Непре-къснат LexRank | 0.3924 | 0.4002 | 0.3963 | Непре-къснат LexRank | 0.3772 | 0.4082 | 0.3966 |

(в) (г)

**Базови методи:**

Random (със случайно избиране): 0.3593
Lead-based: 0.3788

**Базови методи:**

Random (със случайно избиране): 0.3734
Lead-based: 0.3587

При експериментите използваме степенна централност и Lex Rank само с праг 0.1 – който дава и най-добрите наблюдавани резултати. Включваме и два базови метода за сравнение (baseline) за всеки масив от данни. При първия се извличат случайни изречения от клъстера. Прилага се пет пъти за всеки масив от данни, а резултатите в таблицата представят медианата. Вторият базов метод, означен в таблиците като lead-based, използва само характеристиката позиция, без да прилага централност. Това е равносилно на генериране на резюмета, съдържащи просто началните части на текстовете – широко използван базов метод в областта на резюмирането

на текст, който се оказва конкурентен на голяма част от по-сложните методи (Brandow 1995).

Най-добри резултати за всички масиви от данни показват новите методи. И трите метода (степенен, LexRank с праг и непрекъснат LexRank) се представят значително по-добре от базовите методи, както и от центроидното резюмиране – с изключение на експеримента върху данните от DUC 2003, където разликата между центроидния метод и другите методи не е толкова видима. Както изглежда, 0.1 е подходящ праг за постигане на добри резултати, колкото и при непрекъснатия LexRank. Също така е трудно да се каже дали между степенния метод и LexRank има съществена разлика.

Резултатите сочат, че степента вероятно е достатъчно добър показател за оценка на централността на върха в графа на близост. Предвид относително ниската сложност на степенния метод за централност той е добра алтернатива при по-прости имплементации. Степента винаги може да се получи като страничен резултат при изчисляване на LexRank точно преди да се приложи степенният метод (power method) върху графа на близост.

За да получим представа как нашите резултати се съотнасят с тези от други системи за резюмиране, сравнихме стойностите на ROUGE с докладваните от другите участници върху едни и същи данни от DUC. Таблица 4 и Таблица 5 показват стойностите ROUGE-1 за най-добрите пет системи и за ръчно направени резюмета съответно върху данните от DUC 2003 и 2004. Повечето от резултатите ни от LexRank превъзхождат тези на втората най-добра система в DUC 2003, но не успяват да надминат резултатите на първата.

Таблица 4. *Обобщено представяне на официалните стойности за ROUGE върху данните от Задание 2 на DUC 2003.*

| Задание 2 | | |
|---|---|---|
| **Кандидат** | **ROUGE-1** | **95% Confidence** |
| **Код** | **Стойност** | **Интервал** |
| C | 0.4443 | [0.3924,0.4963] |
| B | 0.4425 | [0.4138,0.4711] |
| D | 0.4344 | [0.3821,0.4868] |
| E | 0.4218 | [0.3871,0.4565] |
| A | 0.4168 | [0.3864,0.4472] |
| I | 0.4055 | [0.3740,0.4371] |

| | | |
|---|---|---|
| G | 0.3978 | [0.3765,0.4192] |
| F | 0.3904 | [0.3596,0.4211] |
| J | 0.3895 | [0.3591,0.4199] |
| H | 0.3869 | [0.3659,0.4078] |
| 12 | 0.3798 | [0.3598,0.3998] |
| 13 | 0.3676 | [0.3507,0.3844] |
| 16 | 0.3660 | [0.3474,0.3846] |
| 6 | 0.3607 | [0.3415,0.3799] |
| 26 | 0.3582 | [0.3337,0.3828] |

Таблица 5. *Обобщено представяне на официалните стойности на ROUGE върху данните от Задание 2 и 4 на DUC 2004. Легенда: ръчни резюмета (A – Z) и петте системи с най-добри резултати. Системи 144 и 145 са докладвани от екипа на Мичиганския университет; 144 използва LexRank в комбинация с центроидния метод, докато 145 използва само центроидния метод.*

| Задание 2 | | | Задание 4 | | |
|---|---|---|---|---|---|
| **Кан-дидат** | **ROUGE-1** | **95% доверителен интервал** | **Кан-дидат** | **ROUGE-1** | **95% доверителен интервал** |
| **Код** | **Стойност** | **Интервал** | **Код** | **Стойност** | **Интервал** |
| H | 0.4183 | [0.4019,0.4346] | Y | 0.4445 | [0.4230,0.4660] |
| F | 0.4125 | [0.3916,0.4333] | Z | 0.4326 | [0.4088,0.4565] |
| E | 0.4104 | [0.3882,0.4326] | X | 0.4293 | [0.4068,0.4517] |
| D | 0.4059 | [0.3870,0.4249] | W | 0.4119 | [0.3870,0.4368] |
| B | 0.4043 | [0.3795,0.4291] | **Задание 4a** | | |
| A | 0.3933 | [0.3722,0.4143] | 144 | 0.3883 | [0.3626,0.4139] |
| C | 0.3904 | [0.3715,0.4093] | 22 | 0.3865 | [0.3635,0.4096] |
| G | 0.3890 | [0.3679,0.4101] | 107 | 0.3862 | [0.3555,0.4168] |
| 65 | 0.3822 | [0.3694,0.3951] | 68 | 0.3816 | [0.3642,0.3989] |

| | | | | | |
|---|---|---|---|---|---|
| 104 | 0.3744 | [0.3635,0.3853] | 40 | 0.3796 | [0.3581,0.4011] |
| 35 | 0.3743 | [0.3612,0.3874] | **Задание 4b** | | |
| 19 | 0.3739 | [0.3608,0.3869] | 23 | 0.4158 | [0.3933,0.4382] |
| 124 | 0.3706 | [0.3578,0.3835] | 84 | 0.4101 | [0.3854,0.4348] |
| | | | 145 | 0.4060 | [0.3678,0.4442] |
| | | | 108 | 0.4006 | [0.3700,0.4312] |
| | | | 69 | 0.3984 | [0.3744,0.4225] |

Първите няколко най-добри резултата от всеки метод винаги са статистически неразличими от тези на най-добрата система при официалните оценки в доверителен интервал от 95%. И при трите масива от данни за DUC 2004 нашите методи надминават най-добрия участник при поне един от подходите. Върху данните за DUC 2003 имаме няколко резултата, които са между най-добрата и втората най-добра система.

### 5.3. Експерименти върху данни с шум

Предлаганите от нас методи, използващи графи, разглеждат клъстера от документи като цяло. Централността на дадено изречение се измерва чрез проследяване на отношенията на изречението в рамките на целия клъстер, а не чрез локалната стойност на изречението в документа. Това е особено важно при обобщаващото резюмиране, при което информация, която не е свързана с основната тема на клъстера, би трябвало да бъде изключена от резюмето.

За DUC се използват данни, които хора са разделили ръчно на клъстери от свързани документи. За да видим как методите работят върху данни с шум, добавяме във всеки клъстер по 2 произволни документа от друг клъстер. Тъй като първоначално всеки клъстер съдържа 10 документа, това означава, че имаме 2/12 (17%) шум.

Резултатите от експеримента върху данни с шум са представени в Таблица 6.

Таблица 6. *Стойности за ROUGE-1 при различни параметри в MEAD върху данни с 17% шум от DUC 2003 и 2004*

| 2003 – Задание 2 | | | | 2004 – Задание 2 | | | |
|---|---|---|---|---|---|---|---|
| | min | max | средно | | min | max | средно |
| Центроид | 0.3502 | 0.3689 | 0.3617 | Центроид | 0.3563 | 0.3732 | 0.3630 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Степен (t=0.1) | 0.3501 | 0.3650 | 0.3573 | Степен (t=0.1) | 0.3495 | 0.3762 | 0.3622 |
| LexRank (t=0.1) | 0.3493 | 0.3677 | 0.3603 | LexRank (t=0.1) | 0.3512 | 0.3760 | 0.3663 |
| Непрекъснат LexRank | 0.3564 | 0.3653 | 0.3621 | Непрекъснат LexRank | 0.3465 | 0.3808 | 0.3686 |

(а) (б)

**Базови методи:**
Random (със случайно избиране): 0.2952
Lead-based: 0.3246

**Базови методи:**
Random (със случайно избиране): 0.3078
Lead-based: 0.3418

| 2004 – Задание 4а | | | | 2004 – Задание 4б | | | |
|---|---|---|---|---|---|---|---|
| | min | max | средно | | min | max | средно |
| Центроид | 0.3706 | 0.3898 | 0.3761 | Центроид | 0.3754 | 0.3942 | 0.3906 |
| Степен (t=0.1) | 0.3874 | 0.3943 | 0.3906 | Степен (t=0.1) | 0.3801 | 0.4090 | 0.3963 |
| LexRank (t=0.1) | 0.3883 | 0.3992 | 0.3928 | LexRank (t=0.1) | 0.3710 | 0.4022 | 0.3911 |
| Непрекъснат LexRank | 0.3889 | 0.3931 | 0.3908 | Непрекъснат LexRank | 0.3700 | 0.4012 | 0.3905 |

(в) (г)

**Базови методи:**
Random (със случайно избиране): 0.3315
Lead-based: 0.3615

**Базови методи:**
Random (със случайно избиране): 0.3391
Lead-based: 0.3430

Ситуацията изглежда подобна на тази в Таблица 3, като изключим това, че двата базови метода са повлияни от шума в по-голяма степен. Спадът в резултатите е доста малък при методите, използващи централност в графи. Изненадващото е, че центроидното резюмиране също е с добри резултати, макар и сравнително по-лоши от останалите в повечето случаи. Това показва, че шум от 17% не води до значителни промени в центроида на клъстера.

**6. Преглед на изследванията в областта**

В литературата са описани редица опити за използване на ранкиращи методи, основани на графи, за целите на обработката на естествен език. Салтън, Сиктал, Митра и Бъкли (Salton 1997) са едни от първите, които използват степенната централност за резюмиране на текст от единичен документ. В техния подход стойностите на степента се използват за извличане на важните параграфи от текста.

Монс, Атъндил и Дюмортир (Moens 1999) използват косинусово сходство между изреченията, за да клъстеризират текста на различни тематични отрязъци. С помощта на предварително определен косинусов праг параграфите се клъстеризират около стартови параграфи (seed paragraphs), наречени медоиди (medoids). Стартовите параграфи са определени чрез максимизиране на общата близост между тях и другите параграфи в даден клъстер. Стартовите параграфи се приемат за представителни описания на съответните подтеми и се включват в резюмето.

Джа (Zha 2002) дефинира двуделен граф от множеството от термини („термин“ се използва за назоваване на лексикална единица – бел. прев.) към множеството от изречения. Приема се, че от термин *t* до изречение *s* има ребро, ако *t* се появява в *s*. Според Джа термини, които се появяват в много изречения с високи стойности на значимост, трябва да имат високи стойности на значимост, както и изреченията, съдържащи много термини с високи стойности на значимост, също трябва да имат високи стойности на значимост. Този принцип на взаимно укрепване се свежда до решение за сингуларните вектори на преходната матрица на двуделния граф.

Работата, представена в настоящата статия, започна с приложението на LexRank с праг върху нетегловни графи. Тази имплементация беше използвана за пръв път в заданията в рамките на DUC 2004 през февруари 2004 г., като резултатите бяха представени през май 2004 г. (Erkan 2004b). След заданията по DUC беше направен по-подробен анализ и по-адекватна имплементация на метода, както и сравнение спрямо резюмирането, основано на степенна централност, и центроидното резюмиране (Erkan 2004a).

Непрекъснатият LexRank за тегловни графи за първи път беше представен в първоначалната версия на тази статия от юли 2004 г. Друг алгоритъм, използващ централност чрез собствени вектори върху тегловни графи, беше независимо предложен от Михалча и Тарау за резюмиране на един документ (Mihalcea 2004a). По-късно Михалча, Тарау и Фига (Mihalcea 2004b) прилагат PageRank за друга задача в областта на обработката на естествения език – разрешаване на семантична многозначност.

За разлика от нашата система, представените по-горе изследвания не използват евристични характеристики на изреченията с изключение на стойността за централност, както и не търсят решение на задачата за резюмиране на множество документи. Един от основните проблеми при резюмирането на множество документи е вероятността за дублиране на инфор-

мацията, която се съдържа в различните документи – нещо, което е по-слабо вероятно при резюмиране на единичен документ.

Опитваме се да избегнем повторение на информация в резюметата с помощта на преранкирането, заложено в системата MEAD. Този проблем е разгледан и от Салтън, Сингал, Митра и Бъкли (Salton 1997). Вместо да използват преранкиране, те първо разделят текста на отрязъци по различни подтеми, след което от всеки отрязък вземат поне по един представителен параграф с най-висока стойност за степента.

За определяне на близостта между две изречения използваме показателя косинусово сходство, който се основава на съвпадение на думите и претеглянето по idf. Има обаче и по-надеждни техники за оценка на близост, които често се използват при тематично клъстеризиране на документи или изречения (Hatzivassiloglou 2011, McKeown 2001).

Методите за изчисляване на близост могат да се подобрят чрез включване в системата на повече характеристики (например съвпадение на синоними, съвпадение на глаголно-аргументни структури, съвпадение на лексикални основи) или функционалности (например анафорични зависимост, перифразиране и др.). Тези подобрения не противоречат на модела, представен в тази статия, и могат лесно да бъдат интегрирани.

## 7. Заключение

В статията беше представен нов подход за дефиниране на значимостта на изречения въз основа на оценката за централността в графи. Построяването на граф на близостите между изреченията дава възможност да се види кои изречения са важни, за разлика от центроидния метод, който показва тенденция към свръхгенерализация на информацията от клъстера с документи. Въведохме и три различни метода за изчисляване на централността в графи на близост.

Резултатите от прилагането на тези методи за целите на резюмирането с директно извличане са доста обещаващи. Дори най-простият подход, който използвахме – със степенна централност, е толкова добър евристичен подход, че успява да надмине резултатите от резюмирането, използващо начални изречения, както и центроидното резюмиране. При LexRank се опитахме да използваме повече от информацията в графа и получихме още по-добри резултати. Доказахме също така, че нашите методи не са особено чувствителни към шума в данните, който често е резултат от непрецизно клъстеризиране на документите.

Представянето на отношенията между обектите в естествения език във вид на графи ни дава нови възможности за обработка на информацията с приложение в няколко насоки, включително за клъстеризиране на документи, разрешаване на семантична многозначност, присъединяване на предложни фрази. Отношението на близост, което използвахме за построяване

на графите, може да се замени с произволна друга мярка за взаимна информация между единици от естествения език.

В момента работим върху метод за използване на „случайното блуждаене“ върху двуделни графи (бинарни характеристики вляво, обекти за класифициране вдясно) за разработване на полуконтролиран метод за класификация. Например обектите могат да бъдат имейл съобщения, а бинарната характеристика може да бъде „предметът на съобщението съдържа ли думата *пари*“. Всички обекти са свързани с характеристиките, които се отнасят към тях. Път в графа може да достига от един некласифициран обект към множеството от класифицирани обекти, като минава през поредица от други обекти и характеристики.

При традиционните методи за контролирано или полуконтролирано обучение характеристиките, отнасящи се към некласифицираните примери, не могат да се използват ефективно. При този подход тези характеристики представляват междинни върхове по пътя от некласифицирани към класифицирани върхове. Методът за изчисляване на централността чрез собствени вектори може да припише вероятност за всеки обект (класифициран или некласифициран).

Тази вероятност може на свой ред да бъде интерпретирана като степен на достоверност на класификацията на обекта (например имаме 87% вероятност този конкретен имейл да е спам). При активна среда за обучение можем дори да изберем какъв да е следващият класифициращ признак, който бихме искали да предположим, въз основа на стойностите на централността със собствени вектори за всички обекти.

**Благодарности**

Бихме искали да благодарим на Марк Нюмън за някои ценни библиографски източници, които използвахме за тази статия. Благодарности и на Лилиан Лий за полезните коментари по по-ранна версия на статията. Накрая бихме искали да благодарим и на членовете на екипа на CLAIR (Работна група по компютърна лингвистика и извличане на информация) към Мичиганския университет и особено на Сиуъй Шън за съдействието при реализирането на този проект.

Работата, представена в статията, е отчасти подкрепена от Националната научна фондация, проект 0329043 Probabilistic and link-based Methods for Exploiting Very Large Textual Repositories в рамките на програмата IDM. Авторите носят пълната отговорност за всички мнения, резултати, изводи и препоръки в статията, които не е задължително да отразяват позицията на Националната научна фондация.

## БЕЛЕЖКИ

[1] Статията се препечатва от Lexrank: Graph-based lexical centrality as salience in text summarization. Journal of Artificial Intelligence Research (JAIR), 2004, с разрешението на авторите. Преводът е на Ивелина Стоянова и Цветана Димитрова.

[2] http://www.newsinessence.com

[3] Престижът и централността означават едно и също понятие с тази разлика, че първото често се дефинира като насочен граф, докато второто – като ненасочен граф.

[4] http://www.isi.edu/~cyl/ROUGE

[5] http://www.summarization.com

## ЛИТЕРАТУРА

BARZILAY 1999: *Barzilay, R., M. Elhadad.* Using lexical chains for text summarization. – In: Advances in Automatic Text Summarization, MIT Press, 111–121.

BARZILAY 2003: *Barzilay, R., L. Lee.* Learning to paraphrase: An unsupervised approach using multiple-sequence alignment. – In: Proceedings of HLT-NAACL.

BAXENDALE P. B. 1958: Man-made index for technical literature – an experiment. – IBM J. Res. Dev., 2(4): 354–361.

BRANDOW 1995: *Brandow, R., K. Mitze, L. F. Rau.* Automatic condensation of electronic publications by sentence selection. – Information Processing and Management, 31(5): 675–685.

BREW 2002: *Brew, C., S. Schulte im Walde*. Spectral clustering for german verbs. – In: Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics.

BRIN 1998: *Brin, S., L. Page.* The anatomy of a large-scale hypertextual Web search engine. – Computer Networks and ISDN Systems, 30(1–7): 107–117.

CARBONELL 1998: *Carbonell, J. G., J. Goldstein.* The use of MMR, diversity-based reranking for reordering documents and producing summaries. – In: Research and Development in Information Retrieval, 335–336.

COLLINS M. 1997. Three generative, lexicalised models for statistical parsing. – In: Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics.

DAUME 2004: *Daumé III, H., D. Marcu.* A phrase-based hmm approach to document/abstract alignment. – In: Proceedings of EMNLP 2004, 119–126.

EDMUNDSON H. P. 1969. New Methods in Automatic Extracting. – Journal of the Association for Computing Machinery, 16(2): 264–285, April 1969.

ERKAN 2004A: *Erkan, G., D. R. Radev.* Lexpagerank: Prestige in multidocument text summarization. – In: Proceedings of EMNLP 2004, 365–371.

ERKAN 2004B: *Erkan, G., D. R. Radev.* The University of Michigan at DUC 2004. – In: Proceedings of the Document Understanding Conferences, Boston, MA, May 2004.

HATZIVASSILOGLOU 2011: *Hatzivassiloglou, V., J. Klavans, M. Holcombe, R. Barzilay, M. Kan, K. McKeown.* Simfinder: A flexible clustering tool for summarization.

JING H. 2002. Using hidden markov modeling to decompose Human-Written summaries. – Computational Linguistics, 28(4): 527–543.

KNIGHT 2000: *Knight, K., D. Marcu.* Statistics-based summarization – step one: Sentence compression. – In: Proceeding of the 17th National Conference of the American Association for Artificial Intelligence (AAAI–2000), 703–710.

KUPIEC 1995: *Kupiec, J., J. O. Pedersen, F. Chen*. A trainable document summarizer. – In: Research and Development in Information Retrieval, 68–73.

LIN C.-Y. 1999. Training a Selection Function for Extraction. – In: Proceedings of the Eighteenth Annual International ACM Conference on Information and Knowledge Management (CIKM), 55–62.

LIN 2003: *Lin, C.-Y., E. H. Hovy.* Automatic evaluation of summaries using n-gram co-occurrence. – In: Proceedings of 2003 Language Technology Conference (HLT-NAACL 2003), Edmonton, Canada.

LUHN H. P. 1958. The Automatic Creation of Literature Abstracts. – IBM Journal of Research Development, 2(2): 159–165.

MANI 1997: *Mani, I., E. Bloedorn.* Multi-document summarization by graph search and matching. – In: Proceedings of the Fourteenth National Conference on Artificial Intelligence (AAAI-97), 622–628.

MCKEOWN 2001: *McKeown, K. R., R. Barzilay, D. Evans, V. Hatzivassiloglou, S. Teufel, Y. M. Kan, B. Schiffman.* Columbia Multi-Document Summarization: Approach and Evaluation. – In: Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, New Orleans, LA.

MIHALCEA 2004A: *Mihalcea, R., P. Tarau.* Textrank: Bringing order into texts. – In: Proceedings of EMNLP 2004, 404–411.

MIHALCEA 2004B: *Mihalcea, R., P. Tarau, E. Figa.* Pagerank on semantic networks, with application to word sense disambiguation. – In: Proceedings of the 20st International Conference on Computational Linguistics (COLING 2004), Geneva, Switzerland.

MOENS 1999: *Moens, M.-F. C. Uyttendaele, J. Dumortier.* Abstracting of legal cases: the potential of clustering based on the selection of representative objects. – Journal of the American Society for Information Science, 50(2): 151–161.

OSBORNE M. 2002. Using Maximum Entropy for Sentence Extraction. – In: ACL Workshop on Text Summarization, July 12–13, 2002.

PAGE 1998: *Page, L., S. Brin, R. Motwani, T. Winograd.* The pagerank citation ranking: Bringing order to the web. – Technical report, Stanford University, Stanford, CA, 1998.

RADEV 1998: *Radev, D. R., K. R. McKeown.* Generating natural language summaries from multiple on-line sources. – Computational Linguistics, 24(3): 469–500, September 1998.

RADEV D. 2000A. A common theory of information fusion from multiple text sources, step one: Cross-document structure. – In: Proceedings, 1st ACL SIGDIAL Workshop on Discourse and Dialogue, Hong Kong, October 2000.

RADEV 2000B: *R. Radev, D., H. Jing, M. Budzikowska.* Centroid-based summarization of multiple documents: sentence extraction, utility-based evaluation, and user studies. – In: ANLP/NAACL Workshop on Summarization, Seattle, WA, April 2000.

RADEV 2001: *Radev, D., S. Blair-Goldensohn, Z. Zhang.* Experiments in single and multi-document summarization using MEAD. – In: First Document Understanding Conference, New Orleans, LA, September 2001.

SALTON 1997: *Salton, G., A. Singhal, M. Mitra, C. Buckley.* Automatic Text Structuring and Summarization. – Information Processing & Management, 33(2): 193–207.

SENETA E. 1981. Non-negative matrices and markov chains. Springer-Verlag, New York.

SPARCK-JONES K. 1972. A statistical interpretation of term specificity and its application in retrieval. – Journal of Documentation, 28(1): 11–20.

TOUTANOVA 2004: *Toutanova, K., C. Manning, A. Ng.* Learning random walk models for inducing word dependency distributions. – In: Proceedings of ICML, 2004.

WITBROCK 1999: *Witbrock, M., V. O. Mittal.* Ultra-Summarization: A Statistical Approach to Generating Highly Condensed Non-Extractive Summaries. – In: SIGIR99, Berkeley, 315–316.

YAROWSKY D. 1995. Unsupervised word sense disambiguation rivaling supervised methods. – In: Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics.

ZHA H. 2002. Generic Summarization and Key Phrase Extraction Using Mutual Reinforcement Principle and Sentence Clustering. – In: Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Tampere, Finland, 2002.

✉ *Güneş Erkan*
Software Engineer, Google Inc.
76 9th Ave, New York, NY 10011, USA
*gerkan@umich.edu*

✉ *Dragomir R. Radev*
Professor of Computer Science, Yale University
Room 319, 17 Hillhouse Avenue, New Haven, CT 06511, USA
*dragomir.radev@yale.edu*