Sumarizarea fluxurilor de microbloguri (Rezumat)

Conducător științific:

Prof. Dr. Denis Enăchescu

Andrei Olariu

Facultatea de Matematică și Informatică

Universitatea din București

București Septembrie 2014

Cuprins

Cuprins **2**

1 Introducere **4**
1.1 Introducere 4
1.2 Motivație 5
1.3 Privire de ansamblu 5

2 Rezultate precedente **7**
2.1 Sumarizarea multi-document 7
2.1.1 Sumarizarea extractivă 7
2.1.2 Sumarizarea abstractivă 8
2.2 Actualizarea sumarizării 8
2.3 Cercetarea bazată pe Twitter 8
2.3.1 Preprocesarea tweeturilor 8
2.3.2 Detectarea evenimentelor în fluxurile de microblogging 9
2.3.3 Sumarizarea fluxurilor de microblogging 9
2.4 Cei mai performanți algoritmi 9
2.4.1 Sumarizarea abstractivă - Multi-Sentence Compression 9
2.4.2 Sumarizarea extractivă - Phrase Reinforcement 10

3 Clusterizarea în îmbunătățirea sumarizării **11**
3.1 Motivație 11
3.2 Descrierea abordării 11
3.2.1 Privire de ansamblu 11
3.2.2 Preprocesarea 11
3.2.3 Detectarea evenimentelor 12
3.2.4 Clusterizarea postărilor 12

3.2.5 Algoritmii de sumarizare 12
3.3 Evaluarea 13
3.3.1 Setul de date 13
3.3.2 Evenimentele detectate în setul de date 13
3.3.3 Metricile pentru evaluarea sumarelor 13
3.3.4 Rezultate și analiză 13

4 Clusterizarea ierarhică 15
4.1 Motivație 15
4.2 Descrierea abordării 15
4.2.1 Privire de ansamblu 15
4.2.2 Analiza ierarhică a evenimentelor 15
4.3 Evaluarea 16
4.3.1 Setul de date 16
4.3.2 Evenimente detectate în setul de date 16
4.3.3 Evaluation Metrics for Summaries 17
4.3.4 Interfața de evaluare 17
4.3.5 Rezultate și analiză 17

5 Sumarizarea incrementală 20
5.1 Motivație 20
5.2 Descrierea abordării 20
5.2.1 Privire de ansamblu 20
5.2.2 Graful de cuvinte 20
5.2.3 Construirea grafului de cuvinte 23
5.2.4 Procesarea incrementală 23
5.2.5 Generarea sumarelor 24
5.3 Evaluare pe un set de date de dimensiune redusă 24
5.4 Evaluare pe un set de date de dimensiune mare 24

6 Concluzii 26

Bibliografie 27

Capitolul 1

Introducere

1.1 Introducere

Deși apărut în urmă cu mai puțin de zece ani, fenomenul de microblogging a înregistrat o creștere convecventă an după an. După cum reiese și din nume, termenul este un derivat al bloggingului, cu un mare accent pe postările foarte scurte. Blogurile au fost primele să conteste dihotomia *producător – consumator* din timpul primei perioade a internetului. Însă microblogurile au fost cele care au permis unui număr de peste un miliard de oameni din întreaga lume să genereze, nu numai să consume informație. Aceasta schimbare este centrală conceptului Web 2.0.

Serviciul de microblogging reprezentativ este Twitter. Punând accent pe transparență și pe date publice, Twitter a atras sute de milioane de utilizatori activi, generând până la 500 de milioane de postări pe zi. Interfața programatică oferită de Twitter a permis cercetătorilor să acceseze fluxul informational, ducând la apariția unor domenii noi de cercetare sau la relansarea unora clasice.

Această teză abordează problema sumarizării fluxurilor de date Twitter. Vom arăta cum pot fi îmbunătățiți algoritmii curenți, atât din punct de vedere al vitezei, cât și al calității, prin combinarea cu tehnici de detectarea evenimentelor și clusterizarea mesajelor. Vom experimenta cu modelarea ierarhică a evenimentelor în scopul generării unor sumare de calitate ridicată. Pe baza acestor rezultate, vom dezvolta un algoritm de sumarizare inovativ – Twitter Online Word Graph Summarizer (TOWGS). Procesând fluxurile în mod incremental, TOWGS este cu un ordin de magnitudine mai rapid decât abordările precedente, în același timp fiind capabil să genereze sumare de calitate comparabilă.

Contribuțiile aceste teze sunt următoarele:

- Combinăm detectarea evenimentelor, clusterizarea de texte și sumarizarea pentru a îmbunătăți calitatea rezultatelor și viteza algoritmilor de sumarizat fluxuri [14]. Aceasta este prima abordare capabilă sa sumarizeze fluxuri nefiltrare, de mari dimensiuni. Algoritmii precedenți erau construiți doar pentru seturi de postări filtrare pe baza unui anumit eveniment sau temă.

- Introducem conceptul de sumarizare ierarhica a evenimentelor, împreună cu un algoritm capabil de a genera sumare ierarhice pentru fluxuri Twitter [15].

- Dezvoltăm un algoritm online pentru sumarizarea fluxurilor Twitter [16]. În momentul publicării, acest algoritm era cu un ordin de magnitudine mai rapid decât abordările precedente, generând în același timp rezultate de calitate ridicată. La ediția din 2014 a Conferinței Filielei Europene a Asociației pentru Lingvistică Computațională, lucrarea prezentând acest algoritm a primit premiul *Best Short Paper*.

1.2 Motivație

În decursul ultimilor patru ani, o serie de abordări au fost propuse pentru a rezolva problema excesului de informație pe Twitter. În ceea ce privește aplicațiile comerciale, se observă două tipuri de soluții: orientate spre utilizatori normali și orientate spre profesioniști în marketing. Utilizatorii normali beneficiază de o gamă largă de opțiuni, cum ar fi cele de filtrare puse la dispoziție de către serviciile de microblogging.

Soluțiile dezvoltate pentru profesioniștii în marketing sunt mai complexe și disponibile pe baza unei taxe. Aplicații precum Radian 6, UberVU sau BrandWatch oferă analitici avansate, analiza sentimentului, detectarea utilizatorilor influenți si a știrilor populare.

Algoritmii dezvoltați pentru combaterea excesului de informație pe rețelele sociale primesc la intrare un set sau un flux de postări. În funcție de scenariul pentru care sunt gândiți, acești algoritmi returnează fie grupuri de cuvinte cheie, în cazul problemei de detectarea evenimentelor, fie propoziții complete, în cazul sumarizării. Vom aprofunda ambele scenarii in capitolul 2.

1.3 Privire de ansamblu

Această teză este organizată în șase capitole. Capitolul 2 prezintă rezultate precedente, în scopul definirii contextului și pregătirii cititorului pentru următoarele capitole. Capitolul 3

propune folosirea detectării de evenimente și a clusterizării postărilor înaintea aplicării algoritmilor standard de sumarizare. Capitolul 4 introduce conceptul de sumarizare ierarhică, construind pe baza sistemului dezvoltat în capitolul 3. Adăugând un modul de analiză ierarhică a evenimentelor, algoritmul este capabil să genereze sumare sub formă de arbori. Capitolul 5 descrie un algoritm online pentru sumarizarea incrementală a fluxurilor Twitter. În final, capitolul 6 prezintă concluziile acestei teze.

Capitolul 2

Rezultate precedente

2.1 Sumarizarea multi-document

Sumarizarea multi-document (MDS) reprezintă problema generării de sumare pe baza unui set de texte, de obicei având o anumită temă. Sumarele rezultate permit utilizatorilor să parcurgă rapid multe documente, cum ar fi articole despre știri, pentru a evita problema excesului de informație.

Există doua abordări generale în realizarea sumarelor multi-document: extractive și abstractive. Cu riscul simplificării excesive, considerăm sumarizarea extractivă ca fiind procesul de selectare a propozițiilor întregi din documente, în timp ce sumarizarea abstractivă generează fraze ce pot să nu apară printre datele de intrare. Considerăm tehnicile bazate pe grafuri de cuvinte ca fiind abstractive deoarece ele pot genera propoziții noi. Din punct de vedere teoretic, grafurile de cuvinte sunt extractive la nivel de cuvânt.

2.1.1 Sumarizarea extractivă

Sumarizarea extractivă generează un sumar selectând un număr redus de propoziții din cadrul documentelor date ca intrare. Deoarece propozițiile inițiale nu sunt modificate, nu trebuie avute în vedere problemele de rang gramatical. În schimb trebuie avut grijă ca propozițiile să nu conțină informație redundantă și să fie coerente în sumar, în urma extragerii lor din contextul original.

2.1.2 Sumarizarea abstractivă

În categoria algoritmilor de sumarizare abstractivă, ne vom concentra pe cei folosind grafuri de cuvinte. Nodurile unui graf de cuvinte reprezintă cuvintele ce apar în datele de intrare. Muchii orientate conectează cuvinte consecutive. Sumarele sunt generate căutând drumurile din graf care optimizează un scor.

Filippova [4] a propus un algoritm abstractive denumit Multi-Sentence Compression (MSC). O aplicație a MSC este sumarizarea grupurilor de titluri de știri făcând referire la același eveniment. Ganesan et al. [6, 7] au introdus abordări pentru a genera recenzii succinte bazate pe feedbackul utilizatorilor unor produse.

2.2 Actualizarea sumarizării

Cele mai importante referințe pentru problema actualizării sumarizării pot fi găsite în cadrul sesiunilor dedicate organizate în 2008 și 2009 [2, 3]. O problemă derivată este cea a sumarizării incrementale, când un sumar trebuie modificat pentru a include elemente noi de informație.

2.3 Cercetarea bazată pe Twitter

În cadrul serviciilor de microblogging, Twitter a primit cea mai mare atenție din partea comunității științifice datorită ușurinței cu care pot fi colectate seturi de date de mari dimensiuni. Tweet-urile (postările de pe Twitter) sunt limitate la 140 de caractere, influențând foarte mult stilul de scriere. Abrevierile, jargonul specific internetului, cât și cuvintele scrise greșit sunt comune. Deoarece majoritatea tehnicilor de procesare a limbajului natural au fost dezvoltate pentru texte lungi și gramaticale, tweeturile sunt dificil de procesat. Tehnicile de normalizare pot ajuta în îmbunătățirea rezultatelor altor algoritmi [10]. Contextul postărilor poate fi extins folosind alte postări similare, tweeturi ale aceluiași utilizator sau recunoașterea entităților [19].

2.3.1 Preprocesarea tweeturilor

Pentru a lucra în mod eficient cu datele de pe Twitter, tehnici specifice de normalizare trebuie folosite. Cele mai importante probleme în ce privește preprocesarea datelor sunt tokenizarea si detectarea părților de vorbire (POS tagging). Soluțiile pentru tokenizare sunt în general bazate pe reguli [13]. Algoritmii tradiționali pentru POS tagging, antrenați pe articole din

ziare, nu sunt potriviți pentru textele scurte și conversaționale găsite pe serviciile de microblogging. Gimpel et al. [8] au propus un model avansat antrenat pe tweeturi si conținând un set extins de etichete.

2.3.2 Detectarea evenimentelor în fluxurile de microblogging

Considerăm sumarizarea fluxurilor de microblogging ca fiind o problemă înrudită cu detectarea de evenimente și cea de teme. În lucrările anterioare de cercetare, cât și în această teză, sumarizarea este folosită pentru a înțelege evenimente sau teme populare.

O'Connor et al. [13] au fost primii să propună un algoritm pentru detectarea celor mai importante teme dintr-un flux. Aplicația lor de căutare explorativă permite utilizatorilor să găsească temele legate de un cuvânt cheie, împreună cu exemple de postări. În [9], autorii propun un algoritm (denumit ETree) pentru modelarea ierarhică a evenimentelor.

2.3.3 Sumarizarea fluxurilor de microblogging

În privința sumarizării fluxurilor de postări Twitter, observăm că toate abordările sunt fie axate pe fluxuri filtrate după un anumit criteriu, fie combinate cu detectarea de evenimente.

Sumarizarea extractiva este predominantă. A fost folosită inițial pentru a sumariza postări legate de evenimente simple și structurate. Tweeturile referitoare la evenimentele sportive erau sumarizate folosind regulile și vocabularul specific evenimentului respectiv [12].

Sharifi et al. [20, 21, 22] au introdus algoritmul Phrase Reinforcement (PR). Algoritmul PR sumarizeaza un flux sortat pe baza unui cuvânt cheie prin construirea unui graf aciclic orientat.

2.4 Cei mai performanți algoritmi

2.4.1 Sumarizarea abstractivă - Multi-Sentence Compression

Un scenariu tipic sumarizării extractive multi-document este următorul: propozițiile sunt clusterizate în funcție de similaritate și o propoziție este aleasă din fiecare cluster. Această propoziție ar putea conține informație irelevantă, prin urmare comprimarea ei ar îmbunătăți calitatea sumarului. Acesta este contextul în care Filippova [4] a introdus problema comprimării propozițiilor. Un graf de cuvinte poate fi construit din propozițiile unui cluster. Sumarul comprimat este generat prin găsirea unui drum în graf care optimizeaza o funcție.

2.4.2 Sumarizarea extractivă - Phrase Reinforcement

Sharifi et al. [20, 21, 22] au fost printre primii care au abordat problema sumarizării fluxurilor Twitter. Algoritmul propus se numește Phrase Reinforcement (PR). Fiind dată o temă populară sub forma unui cuvânt sau a unei fraze cheie, algoritmul PR construiește un graf al frazelor și o extrage pe cea mai folosită care conține tema dată. Algoritmul este bazat pe două observații. Întâi, utilizatorii au tendința să folosească aceleași cuvinte pentru a vorbi despre un anumit aspect al unei teme. În al doilea rând, fenomenul de retweeting (re-postare) este foarte popular în a evidenția postări de calitate.

Capitolul 3

Clusterizarea în îmbunătățirea sumarizării

3.1 Motivație

Acest capitol propune clusterizarea postărilor ca pas de preprocesare înaintea sumarizării fluxurilor de microblogging. În momentul publicării [14], această tehnica era prima capabilă a sumariza fluxuri nefiltrate. Era de asemenea prima care folosea clusterizarea, devenită apoi o componentă centrala în sistemele de sumarizare [23, 24].

3.2 Descrierea abordării

3.2.1 Privire de ansamblu

Algoritmul propus în acest capitol poate fi prezentat ca o serie de componente ce proceseaza datele succesiv – a se observa figura 3.1. Fluxul de date este împărțit în ferestre. În continuare vom lucra cu ferestre conținând datele pe o zi.

Tehnicile și rezultatele prezentate în acest capitol au fost publicate în [14].

3.2.2 Preprocesarea

În cadrul acestei teze vom folosi tehnici de preprocesare de bază. Tokenizarea este implementată direct. Cuvintele sunt separate pe baza spațiilor și a punctuației. Pentru detectarea părților de vorbire folosim librăria dezvoltată de Frederik De Bleser [5]. Pentru a extrage rădăcinile cuvintelor folosim Porter Stemmer [17].

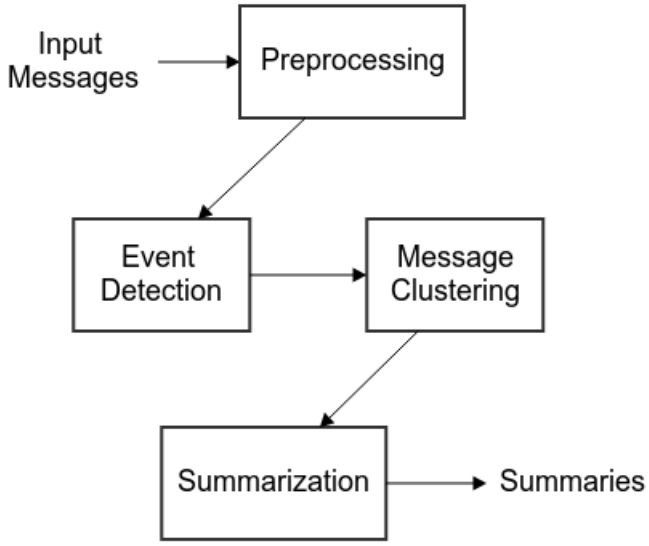


Fig. 3.1 O privire de ansamblu asupra arhitecturii algoritmului de sumarizare.

3.2.3 Detectarea evenimentelor

Detectarea evenimentelor este realizată prin analizarea cuvintelor care prezintă creșteri neobișnuite ale frecvenței de apariție in fereastra curentă, raportat la o fereastră precedentă, de referință. Abordarea este inspirată de algoritmi precedenți. O'Connor et al. [13] sunt primii care folosesc contrastul frecvențelor în identificarea temelor relevante într-un set filtrat de postări.

3.2.4 Clusterizarea postărilor

În acest moment avem grupuri de cuvinte, fiecare grup reprezentând un eveniment. Continuăm prin a atribui fiecare postare unui eveniment sau, dacă nu există niciun eveniment potrivit, a o eticheta drept zgomot.

3.2.5 Algoritmii de sumarizare

Vom folosi doi algoritmi diferiți pentru a sumariza postările. Prima opțiune este reprezentată de Multi-Sentence Compression [4]. A doua alegere este o versiune îmbunătățită a Phrase Reinforcement [21]. Vom testa în ce măsură clusterizarea crește calitatea sumarizării folosind atât MSC, cât și PR-ul modificat. Astfel vom fi siguri că diferența nu va fi legată de vreo particularitate a algoritmului de sumarizare. Cele două opțiuni acoperă ambele clase de abordări în sumarizare, MSC fiind abstractiv, iar PR fiind extractiv.

Algoritmul PR, prezentat în secțiunea 2.4.2, sumarizează un flux de postări filtrate pe

Ziua \ Tipul evenimentului	Real	Virtual	Comun	Fals	Total
29 aprilie	13	8	3	4	28
30 aprilie	19	3	2	8	32
1 mai	16	7	2	4	29
2 mai	11	6	3	2	22
Total	59 (53%)	24 (22%)	10 (9%)	18 (16%)	111

Tabelul 3.1 Distributia evenimentelor descoperite în setul de date.

baza unei fraze cheie. În scenariul nostru avem nevoie de o abordare mai flexibilă. Seturile noastre de postări nu au o anumită frază în comun. Prin urmare, introducem algoritmul Frequent Phrase Summarization (FPS) pentru a selecta cea mai frecventă frază pe baza datelor de intrare.

3.3 Evaluarea

3.3.1 Setul de date

Am colectat 140000 de postări pe zi pentru perioada 22 aprilie – 2 mai 2012. Am folosit postările din intervalul 22 – 28 mai ca set de referință. Algoritmul nostru de sumarizare a fost aplicat asupra datelor din celelalte patru zile. Am evaluat și analizat rezultatele.

3.3.2 Evenimentele detectate în setul de date

Analizând frecvențele cuvintelor din cele patru zile de test am detectat, în medie, 28 de evenimente pe zi. Am împărțit aceste evenimente în patru categorii: reale, virtuale, comune si false. Mai multe informații pot fi observate în tabelul 3.1.

3.3.3 Metricile pentru evaluarea sumarelor

Pentru a evalua sumarele am folosit două metrici (inspirate din [1, 11]): completitudine și gramaticalitate, ambele notate pe o scara de la 1 la 5. Sumarele celor 111 evenimente au fost evaluate de către un voluntar.

3.3.4 Rezultate și analiză

Pentru cele 111 sumare generate, nota medie pentru completitudine a fost 2.98, în timp ce nota medie pentru gramaticalitate a fost 3.63. Cu astfel de note medii consideram calitatea

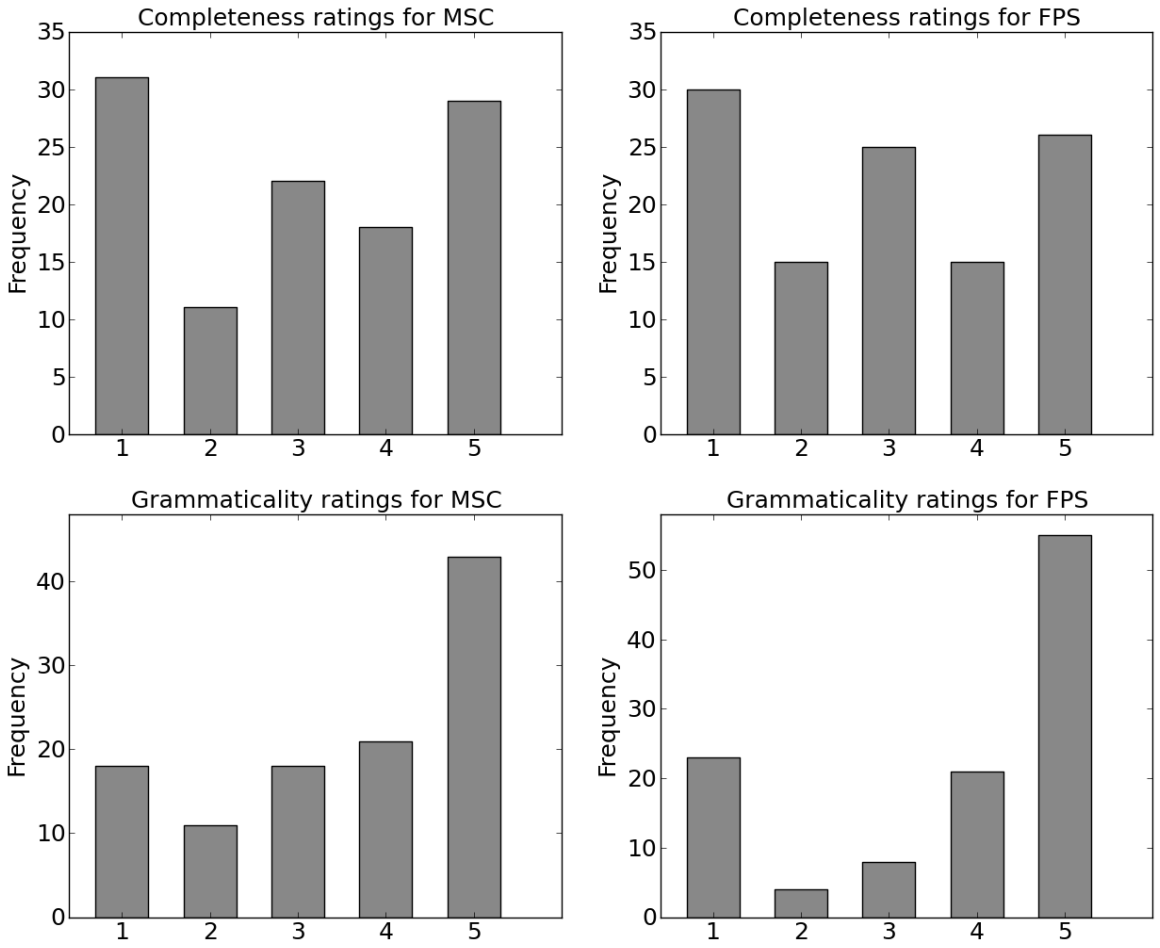


Fig. 3.2 Distributia notelor în funcție de algoritm și de metrică.

sumarelor ca fiind decentă.

Distributia notelor pentru fiecare din cei doi algoritmi este prezentată in figura 3.2. Comparând algoritmii, observăm că aceștia au performanțe similare în privința completitudinii, cu note medii de 2.93 pentru FPS si 3.03 pentru MSC. Distribuția notelor pentru gramaticalitate arată mai bine. Cu o medie de 3.73, FPS produce sumare mai gramaticale decât MSC (3.54).

Capitolul 4

Clusterizarea ierarhică

4.1 Motivație

Gu et al. [9] a introdus un algoritm de detectat evenimente capabil să descompună evenimente complexe și să genereze ierarhii. Pe baza acestuia si a rezultatelor teoretice din capitolul 3 dezvoltăm un sistem pentru generarea de arbori de sumarizare.

4.2 Descrierea abordării

4.2.1 Privire de ansamblu

Abordarea noastră o urmează îndeaproape pe cea din capitolul 3. Arhitectura generală este prezentată în figura 4.1. Comparând cu arhitectura din figura 3.1, se observă apariția unei componente noi. Aceasta primește ca date de intrare postările asignate unui eveniment și generează un arbore de clusterizare ierarhică pe baza similarității dintre postări.

Tehnicile și rezultatele prezentate în acest capitol au fost publicate în [15].

4.2.2 Analiza ierarhică a evenimentelor

Având determinate clusterele de postări, fiecare cluster asociat unui eveniment, dorim să îmbunătățim analiza unui eveniment prin descoperirea aspectelor și a sub-evenimentelor acestuia. Modulul de analiză ierarhică primește ca date de intrare postările despre un eveniment și returnează un arbore de clusterizare în care fiecare nod reprezintă un eveniment, cu atât mai specific cu cât nodul se află mai în profunzimea arborelui.

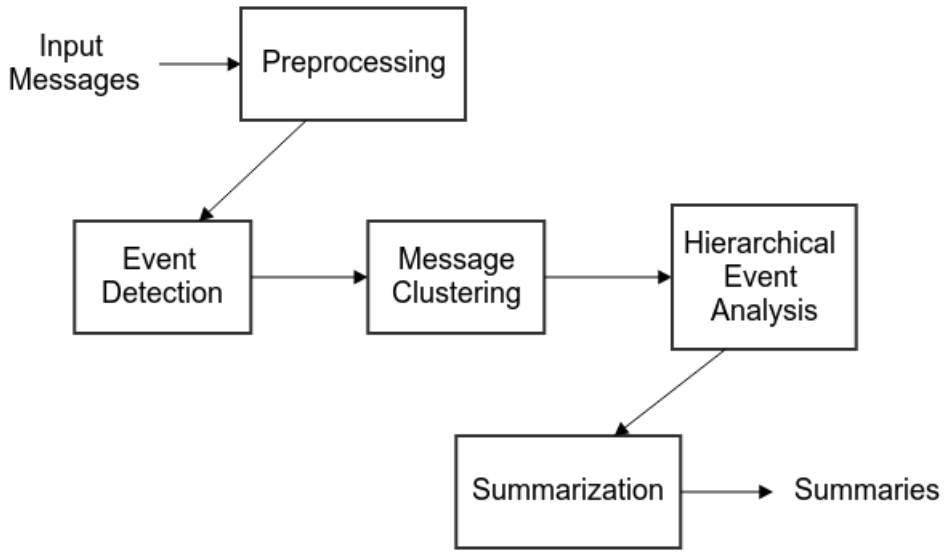


Fig. 4.1 O privire de ansamblu asupra arhitecturii algoritmului de sumarizare.

Day \ Event Type	Real	Virtual	Total
4th July	12	19	31
5th July	6	6	12
6th July	4	15	19
7th July	5	15	20
Total	27(33%)	55(67%)	82

Tabelul 4.1 Distribuția evenimentelor detectate în setul de date.

4.3 Evaluarea

4.3.1 Setul de date

Evaluarea a fost efectuată folosind un set de date de 1.6 milioane de postări colectate între 4 si 8 iulie 2012. De asemenea am colectat alte 1.7 milioane de postări în săptămâna precedentă pentru a le folosi ca set de referință.

4.3.2 Evenimente detectate în setul de date

Algoritmul a detectat, în medie, 20 de evenimente pe zi. Am împărțit evenimentele în două categorii: reale și virtuale. Distribuția acestora este prezentată în tabelul 4.1.

4.3.3 Evaluation Metrics for Summaries

Fiecare sumar a fost evaluat în funcție de cele două metrici introduse în secțiunea 3.3.3: completitudine și gramaticalitate.

Pentru a observa îmbunătățirea produsă de sumarizarea ierarhică în fața sumarizării simple a unui cluster, am tăiat arborele ierarhic de sumarizare la nivelul la care are patru clustere, pe care apoi le-am sumarizat. Sumarele cu mai puțin de patru clustere în total au fost eliminate, rezultând un total de 50 de sumare. Am rugat apoi voluntarii să noteze și nivelul de non-redundanță, pentru a măsura dacă cele patru propoziții din fiecare sumar repetă aceeași informație.

4.3.4 Interfața de evaluare

Patru voluntari au fost rugați să noteze sumarele generate pentru cele 50 de evenimente. Ei au avut acces la o interfață grafică. Fiecare eveniment a fost prezentat într-o pagină web separată.

4.3.5 Rezultate și analiză

În cazul sumarelor clasice(o singură propoziție), notele medii pentru completitudine au fost 3.05 (pentru MSC) si 3.28 (pentru FPS). Analizând sumarele ierarhice (patru popozitii), observăm o creștere a notelor până la 4.28 (MSC) si 4.11 (FPS).

În privința gramaticalității, notele pentru sumare clasice sunt bune: 4.05 pentru MSC si 4.00 pentru FPS. În schimb, crescând mărimea sumarelor la patru propoziții, notele scad la 4.00 (MSC) si 3.61 (FPS).

Notele medii pentru non-redundanță sunt foarte bune. MSC și FPS obțin 4.01, respectiv 3.82. Voluntarii au considerat că foarte puțină informație este repetată în cadrul sumarelor de patru propoziții.

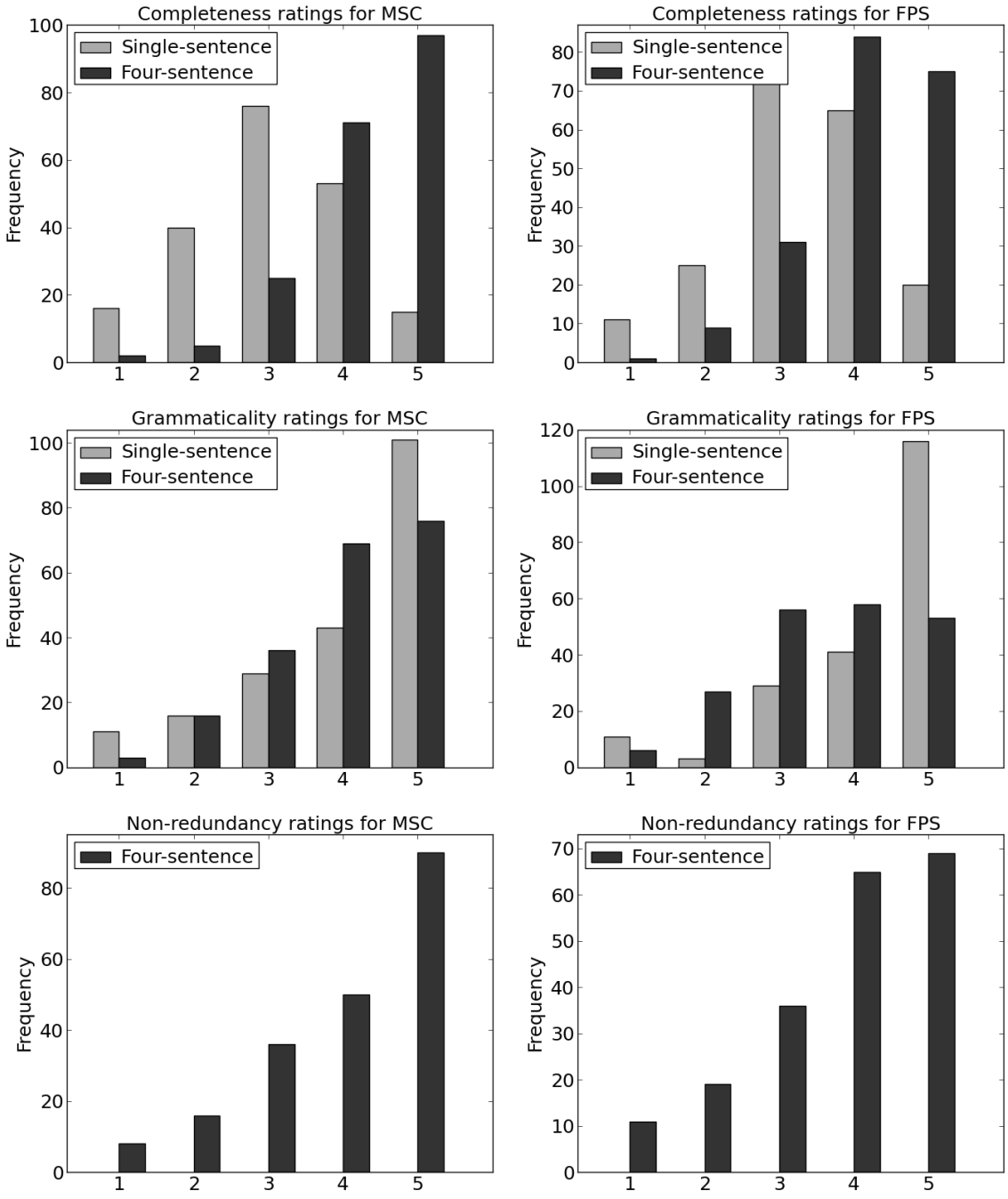


Fig. 4.2 Distribuția sumarelor pentru fiecare algoritm și fiecare metrică.

Metrică	**Mărime sumar**	**Notă medie**	**Creștere**
Completitudine MSC	O propoziție	3.05	40.3%
	Patru propoziții	4.28	
Completitudine FPS	O propoziție	3.28	25.3%
	Patru propoziții	4.11	
Gramaticalitate MSC	O propoziție	4.05	-1.2%
	Patru propoziții	4.00	
Gramaticalitate FPS	O propoziție	4.25	-15.0%
	Patru propoziții	3.61	
Non-redundanță MSC	Patru propoziții	4.01	-
Non-redundanță FPS	Patru propoziții	3.82	-

Tabelul 4.2 Comparație a notelor medii între sumarele de o propoziție și cele de patru propoziții, pentru fiecare metrică și fiecare algoritm.

Capitolul 5

Sumarizarea incrementală

5.1 Motivație

Acest capitol prezintă algoritmul Twitter Online Word Graph Summarizer. TOWGS procesează datele incremental, în forma lor nativă de flux. Fiecare postare, după ce este procesată, nu este salvată în memorie. Reducând cerințele de memorie și îmbunătățind timpul de execuție, algoritmul poate procesa fluxuri de dimensiuni mai mari față de abordările precedente.

5.2 Descrierea abordării

5.2.1 Privire de ansamblu

TOWGS folosește un graf al cuvintelor avansat. Vom prezenta avantajele acestuia în secțiunea 5.2.2. Secțiunea 5.2.3 va descrie modul de construire al grafului. Tehnicile de procesare incrementală vor fi aprofundate în secțiunea 5.2.4. Secțiunea 5.2.5 va prezenta modul de generare al unui sumar.

Tehnicile și rezultatele prezentate în acest capitol au fost publicate în [16].

5.2.2 Graful de cuvinte

În capitolul 2 am menționat grafurile de cuvinte (grafuri având cuvinte ca noduri și bigrame ca muchii). Le-am folosit în capitolele 3 și 4 pentru a efectua sumarizare abstractivă. Am observat cum calitatea rezultatelor depinde de similaritatea postărilor sumarizate. Diversitatea datelor afectează algoritmul de sumarizare deoarece informațiile relevante sunt

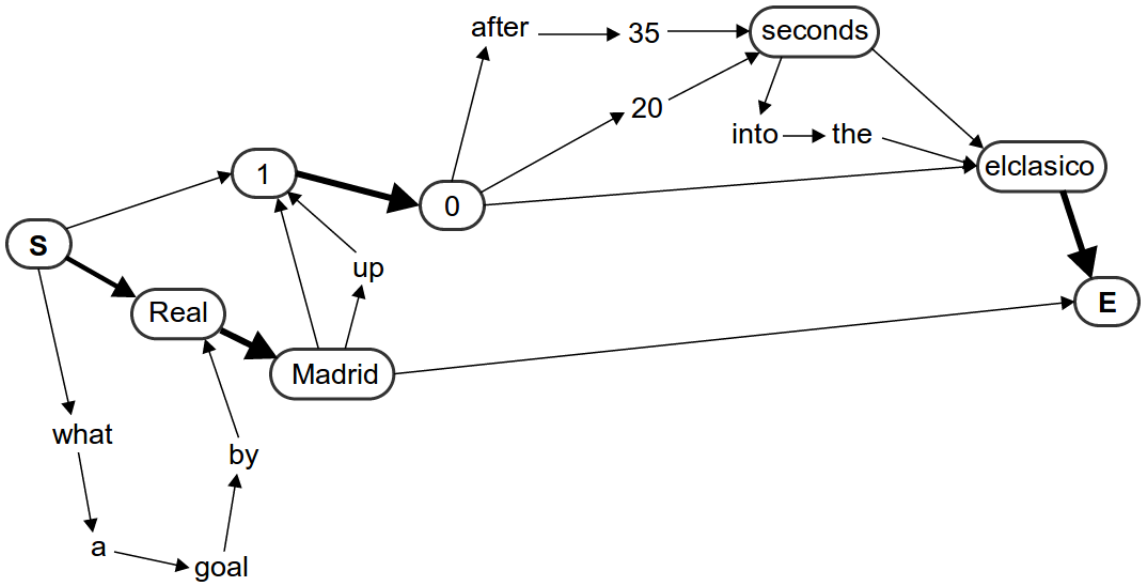


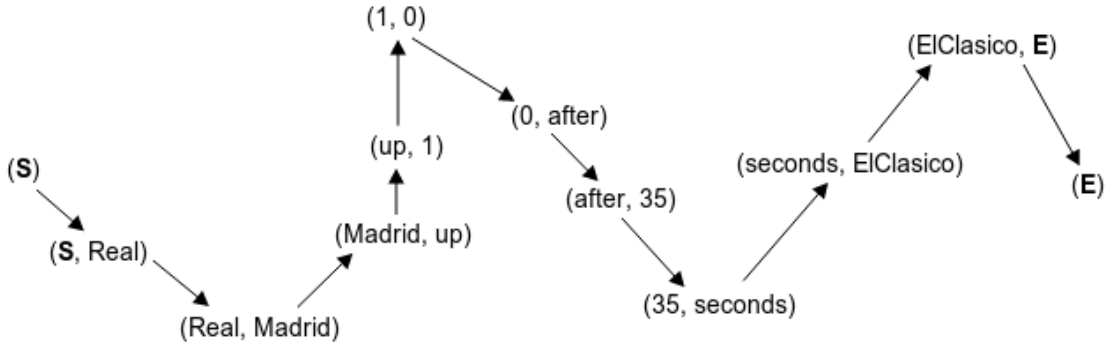
Fig. 5.1 Exemplu de graf de cuvinte simplu (bazat pe bigrame).

diluate de zgomot. Abordarea noastră în combaterea acestei probleme constă în folosirea trigramelor pentru a construi graful. Astfel, vom avea bigrame ca noduri și trigrame ca muchii. Fiind dată o propozitie (w_1, w_2, w_3, w_4), adăugăm două cuvinte speciale (pentru a marca începutul și sfârșitul propoziției) și generăm următoarele muchii: $(S, w_1) \rightarrow (w_1, w_2)$, $(w_1, w_2) \rightarrow (w_2, w_3)$, $(w_2, w_3) \rightarrow (w_3, w_4)$ and $(w_3, w_4) \rightarrow (w_4, E)$. Ponderi sunt adăugate nodurilor și muchiile pentru a păstra frecvențele bigramelor și ale trigramelor.

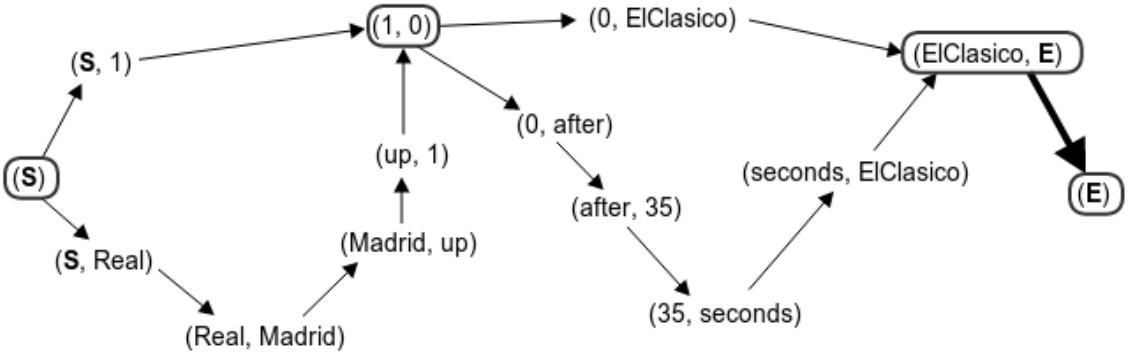
Vom ilustra diferențele între graful de cuvinte simplu și cel bazat pe trigrame pornind de la următorul exemplu format din patru postări:

- *Real Madrid up 1 0 after 35 seconds elclasico*
- *1 0 elclasico*
- *What a goal by Real Madrid*
- *Real Madrid 1 0 20 seconds into the elclasico*

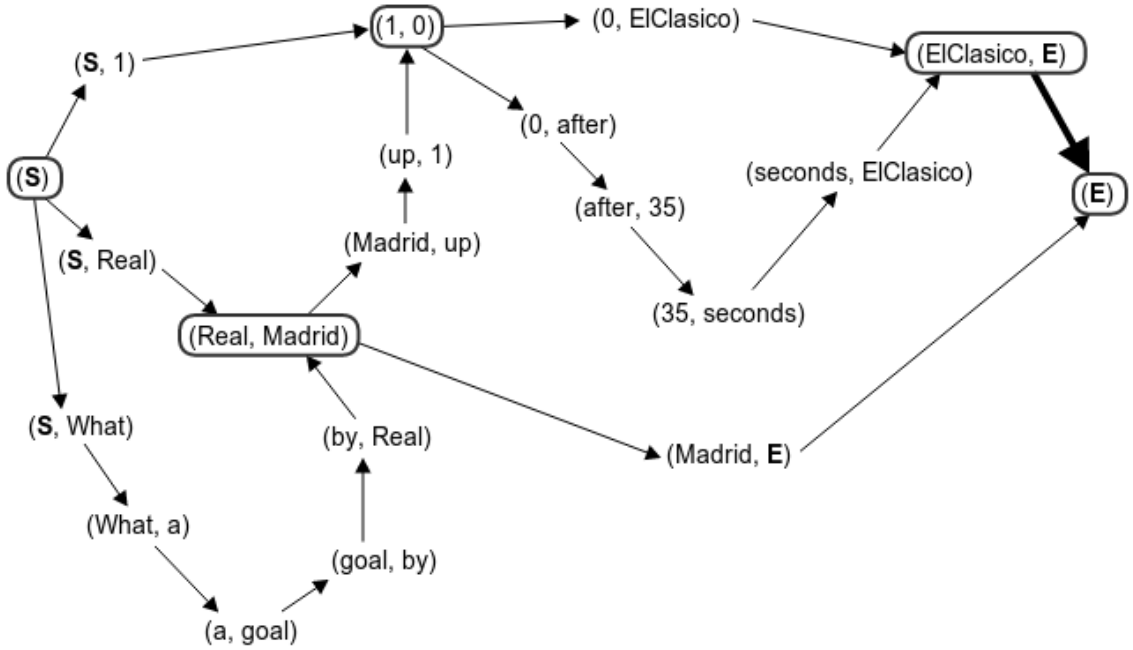
Graful de cuvinte simplu construit pe baza acestor postări este prezentat în figura 5.1. Folosind același input, graful bazat pe trigrame este cel din figura 5.2(d). Frazele frecvente (evidențiate în ambele grafuri) sunt aceleași. În schimb, se observă cum graful bazat pe trigrame are o complexitate crescută.



(a) Graful de cuvinte după adăugarea primei propozitii.

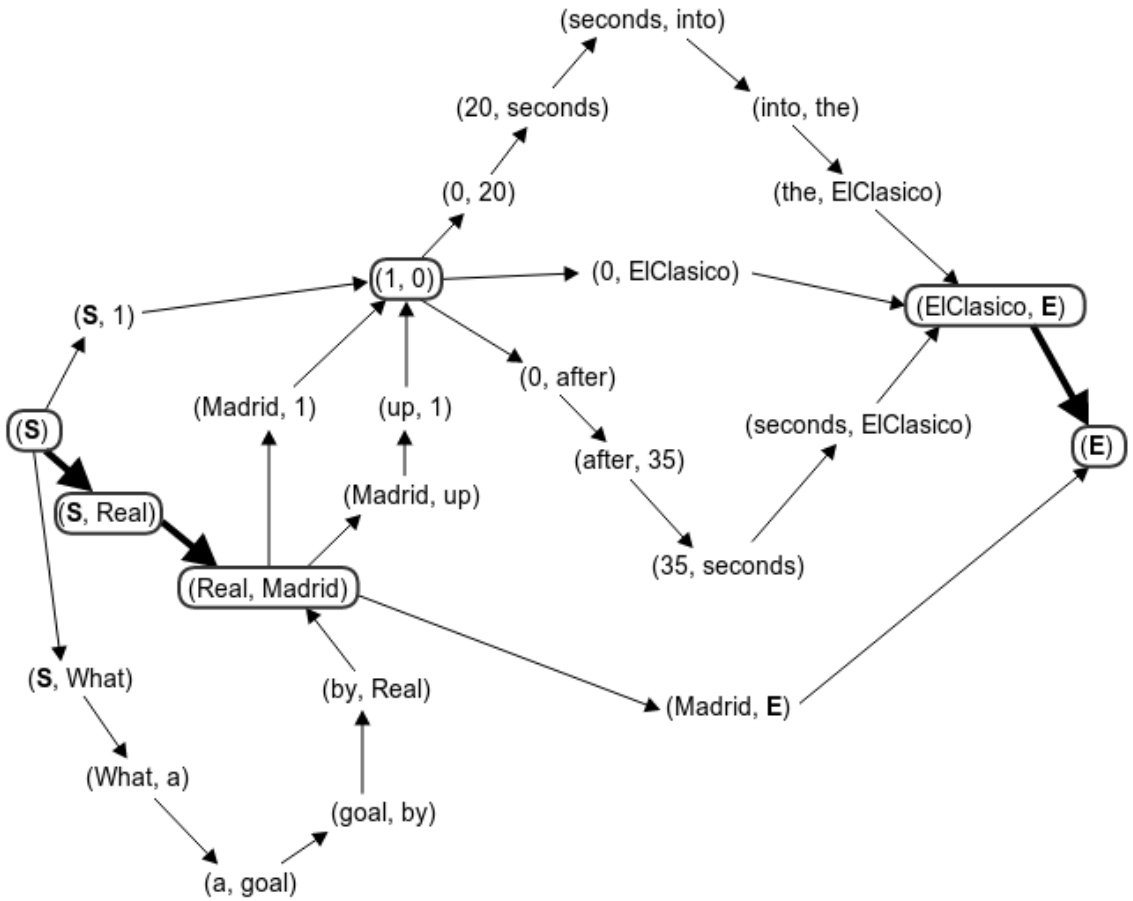


(b) Graful de cuvinte după adăugarea primelor două propozitii.



(c) Graful de cuvinte după adăugarea primelor trei propozitii.

Fig. 5.2 Examplul unui graf de cuvinte bazat pe trigrame.



(d) Graful de cuvinte după adăugarea celor patru propozitii.

Fig. 5.2 (continuare) Examplul unui graf de cuvinte bazat pe trigrame.

5.2.3 Construirea grafului de cuvinte

Figura 5.2 prezintă un graf de cuvinte generat pe baza celor patru propoziții prezentate anterior. Graful obținut după procesarea primei propoziții este cel din figura 5.2(a). Continuăm adăugând a doua propoziție – figura 5.2(b), apoi a treia – 5.2(c) si a patra – 5.2(d). Frazele frecvente observate în graf sunt *S Real Madrid*, *1 0* and *elclasico E*. De asemenea, putem observa că reconstrucția unor propoziții gramaticale nu este dificilă. Urmărind aproape orice drum în graf duce la un sumar corect.

5.2.4 Procesarea incrementală

Algoritmul TOWGS simulează procesul de *uitare* folosind ferestre exponențiale de uitare [18, section 4.7]. Ferestrele clasice acordă o pondere de 1 elementelor recente și 0 celorlalte

elemente. O fereastră exponențială acordă 1 elementului curent, apoi pentru fiecare element nou scade ponderile elementelor precedente.

5.2.5 Generarea sumarelor

Fiind dat un graf de cuvinte, generarea unui sumar implică căutarea drumului cu scor maxim din graf. Acest drum conectează cuvintele speciale ce marchează începutul și sfârșitul fiecărei propoziții. Deoarece găsirea unei soluții exacte este ineficientă, folosim o abordare greedy.

5.3 Evaluare pe un set de date de dimensiune redusă

Această evaluare are scopul de a analiza și înțelege comportamentul algoritmului și modul în care acesta este influențat de către parametrul *c* al ferestrei exponențiale. În urma experimentului a rezultat că parametrul *c* trebuie setat în funcție de intervalul de timp din care utilizatorii se așteaptă să fie generat sumarul. Anumite evenimente se pretează unor sumare pe ferestre scurte (atunci când utilizatorul este interesat de aspecte specifice), în timp ce alte evenimente pot fi analizate folosind ferestre largi, pentru a obține rezultate mai generale.

5.4 Evaluare pe un set de date de dimensiune mare

Această a doua evaluare este efectuată pe un set de date format din peste trei milioane de postări. Analizăm capacitatea algoritmului TOWGS de a sumariza date variate și comparăm sumarele cu cele ale unui algoritm de bază. În rolul acestuia din urmă alegem Multi-Sentence Compression.

Am rugat cinci voluntari să evalueze sumarele rezultate. Notele acordate celor 64 de sumare generate de fiecare din cei doi algoritmi sunt prezentate în figura 5.3. Notele medii pentru completitudine sunt foarte similare, cu un ușor avantaj în favoarea TOWGS (4.29 față de 4.16 pentru MSC). În privința gramaticalității se observă note mai bune pentru TOWGS (4.30) raportat la MSC (4.16).

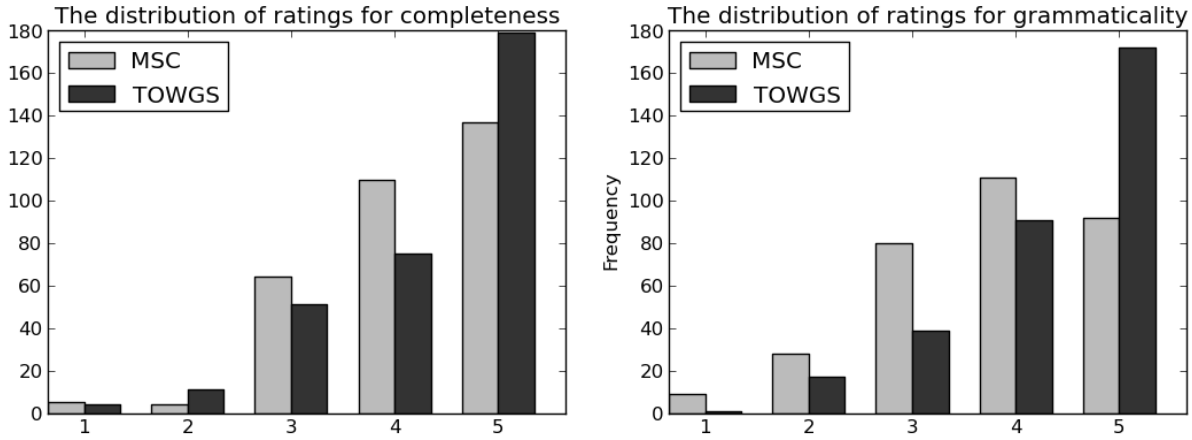


Fig. 5.3 Distribuția notelor acordate de voluntari, în funcție de algoritm și metrică.

Capitolul 6

Concluzii

Din momentul primelor abordări ale problemei sumarizării fluxurilor de microblogging și până astăzi, abordările au evoluat în același timp cu nivelul de înțelegere pe care cercetătorii îl au despre particularitățile și dinamica serviciilor de microblogging. Această evoluție poate fi observată și în teza curentă, cu rezultatele capitolelor 3, 4 și 5 fiind publicate în 2012 [14], 2013 [15], respectiv 2014 [16].

Prezentăm în continuare câteva idei pentru activitatea viitoare de cercetare. Prima idee este inspirată de similaritățile și diferențele dintre cei doi algoritmi incrementali publicați până în acest moment: TOWGS (capitolul 5) și Sumblr [23]. Considerăm că este posibilă elaborarea unei abordări inspirate din acești doi algoritmi, care să fie incrementală, extractivă și să nu necesite clustering.

O a doua observație este aceea că algoritmii iterativi curenți sunt adaptați pentru a sumariza evenimente importante. Fiind dat ca intrare un flux de mari dimensiuni, sumarizarea unui eveniment minor nu a fost încercată sau analizată. Din cauza arhitecturii acestora, atât TOWGS cât și Sumblr sunt neadecvați pentru acest scenariu.

Teza curentă a introdus tehnici și algoritmi noi pentru abordarea problemei sumarizării fluxurilor de microbloguri. Am construit succesiv, pornind de la abordări simple și continuând spre sisteme complexe, capabile de sumarizare incrementală sau procesare ierarhică. Avem convingerea că aceste idei se vor dovedi esențiale inovațiilor viitoare și vor duce la apariția unor aplicații avansate pentru analiza și monitorizarea social media.

Bibliografie

[1] Dang, H. T. (2005). Overview of duc 2005. In *In Proceedings of the Document Understanding Conf. Wksp. 2005 (DUC 2005) at the Human Language Technology Conf./Conf. on Empirical Methods in Natural Language Processing (HLT/EMNLP*. Cited on page 13.

[2] Dang, H. T. (2009). Tac 2009 update summarization task. Technical report, National Institute of Standards and Technology. Cited on page 8.

[3] Dang, H. T. and Owczarzak, K. (2008). Overview of the tac 2008 update summarization task. In *Proceedings of text analysis conference*, pages 1–16. Cited on page 8.

[4] Filippova, K. (2010). Multi-sentence compression: finding shortest paths in word graphs. In *Proceedings of the 23rd International Conference on Computational Linguistics*, COLING '10, pages 322–330, Stroudsburg, PA, USA. Association for Computational Linguistics. Cited on pages 8, 9, and 12.

[5] Frederik De Bleser, Tom De Smedt, L. N. (2002). Nodebox version 1.9.5 for mac os x. Retrieved March 2010, from: http://nodebox.net. Cited on page 11.

[6] Ganesan, K., Zhai, C., and Han, J. (2010). Opinosis: a graph-based approach to abstractive summarization of highly redundant opinions. In *Proceedings of the 23rd International Conference on Computational Linguistics*, COLING '10, pages 340–348, Stroudsburg, PA, USA. Association for Computational Linguistics. Cited on page 8.

[7] Ganesan, K., Zhai, C., and Viegas, E. (2012). Micropinion generation: An unsupervised approach to generating ultra-concise summaries of opinions. In *Proceedings of the 21st International Conference on World Wide Web*, WWW '12, pages 869–878, New York, NY, USA. ACM. Cited on page 8.

[8] Gimpel, K., Schneider, N., O'Connor, B., Das, D., Mills, D., Eisenstein, J., Heilman, M., Yogatama, D., Flanigan, J., and Smith, N. A. (2011). Part-of-speech tagging for twitter: Annotation, features, and experiments. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: Short Papers - Volume 2*, HLT '11, pages 42–47, Stroudsburg, PA, USA. Association for Computational Linguistics. Cited on page 9.

[9] Gu, H., Xie, X., Lv, Q., Ruan, Y., and Shang, L. (2011). Etree: Effective and efficient event modeling for real-time online social media networks. In *Proceedings of the 2011 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology - Volume 01*, WI-IAT '11, pages 300–307, Washington, DC, USA. IEEE Computer Society. Cited on pages 9 and 15.

[10] Kaufmann, M. and Kalita, J. (2010). Syntactic normalization of Twitter messages. In *Proceedings of the 8th International Conference on Natural Language Processing (ICON 2010)*, Chennai, India. Macmillan India. Cited on page 8.

[11] Lin, C.-Y. and Hovy, E. (2003). Automatic evaluation of summaries using n-gram co-occurrence statistics. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1*, NAACL '03, pages 71–78, Stroudsburg, PA, USA. Association for Computational Linguistics. Cited on page 13.

[12] Nichols, J., Mahmud, J., and Drews, C. (2012). Summarizing sporting events using twitter. In *Proceedings of the 2012 ACM international conference on Intelligent User Interfaces*, IUI '12, pages 189–198, New York, NY, USA. ACM. Cited on page 9.

[13] O'Connor, B., Krieger, M., and Ahn, D. (2010). TweetMotif: Exploratory Search and Topic Summarization for Twitter. In Cohen, W. W., Gosling, S., Cohen, W. W., and Gosling, S., editors, *ICWSM*. The AAAI Press. Cited on pages 8, 9, and 12.

[14] Olariu, A. (2012). Clustering to improve microblog stream summarization. In *Symbolic and Numeric Algorithms for Scientific Computing (SYNASC), 2012 14th International Symposium on*, pages 220–226, Los Alamitos, CA, USA. IEEE Computer Society. Cited on pages 5, 11, and 26.

[15] Olariu, A. (2013). Hierarchical clustering in improving microblog stream summarization. In *Proceedings of the 14th international conference on Computational Linguistics and Intelligent Text Processing - Volume 2*, CICLing'13, pages 424–435, Berlin, Heidelberg. Springer-Verlag. Cited on pages 5, 15, and 26.

[16] Olariu, A. (2014). Efficient online summarization of microblogging streams. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics, volume 2: Short Papers*, pages 236–240, Gothenburg, Sweden. Association for Computational Linguistics. Cited on pages 5, 20, and 26.

[17] Porter, M. F. (1997). Readings in information retrieval. chapter An Algorithm for Suffix Stripping, pages 313–316. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA. Cited on page 11.

[18] Rajaraman, A. and Ullman, J. D. (2011). *Mining of Massive Datasets*. Cambridge University Press, New York, NY, USA. Cited on page 23.

[19] Ritter, A., Clark, S., Mausam, and Etzioni, O. (2011). Named Entity Recognition in Tweets: An Experimental Study. In *EMNLP*. Cited on page 8.

[20] Sharifi, B., Hutton, M.-A., and Kalita, J. (2010a). Automatic summarization of twitter topics. In *National Workshop on Design and Analysis of Algorithm, Tezpur, India*. Cited on pages 9 and 10.

[21] Sharifi, B., Hutton, M.-A., and Kalita, J. (2010b). Summarizing microblogs automatically. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, HLT '10, pages 685–688, Stroudsburg, PA, USA. Association for Computational Linguistics. Cited on pages 9, 10, and 12.

[22] Sharifi, B., Hutton, M.-A., and Kalita, J. K. (2010c). Experiments in microblog summarization. In *Proceedings of the 2010 IEEE Second International Conference on Social Computing*, SOCIALCOM '10, pages 49–56, Washington, DC, USA. IEEE Computer Society. Cited on pages 9 and 10.

[23] Shou, L., Wang, Z., Chen, K., and Chen, G. (2013). Sumblr: Continuous summarization of evolving tweet streams. In *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '13, pages 533–542, New York, NY, USA. ACM. Cited on pages 11 and 26.

[24] Yang, X., Ghoting, A., Ruan, Y., and Parthasarathy, S. (2012). A framework for summarizing and analyzing twitter feeds. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '12, pages 370–378, New York, NY, USA. ACM. Cited on page 11.