



**УНИВЕРЗИТЕТ „СВ. КИРИЛ И МЕТОДИЈ“**  
**СКОПЈЕ, Р. МАКЕДОНИЈА**  
**ФАКУЛТЕТ ЗА ИНФОРМАТИЧКИ НАУКИ**  
**И КОМПЈУТЕРСКО ИНЖЕНЕРСТВО**  
Институт за софтверско инженерство



Ѓорѓи Маџаров

**НОВИ МЕТОДИ ЗА ГРАДЕЊЕ НА ХИЕРАРХИСКИ**  
**ПОВЕЌЕЦЕЛНИ КЛАСИФИКАТОРИ**

- докторска дисертација -

Скопје, 2012 година

Ментор:

Вонр. проф. д-р Дејан Ѓорѓевиќ  
ФИНКИ – Скопје Р. Македонија

Коментор:

Проф. д-р Сашо Џероски  
Институт „Јожеф Стефан“ – Љубљана, Р.  
Словенија

Комисија за оцена и одбрана:

1. Проф. д-р Драган Михајлов, претседател,  
ФИНКИ – Скопје, Р. Македонија
2. Вонр. проф. д-р Дејан Ѓорѓевиќ, ментор,  
ФИНКИ – Скопје, Р. Македонија
3. Проф. д-р Сашо Џероски, коментор,  
Институт „Јожеф Стефан“ – Љубљана, Р.  
Словенија
4. Проф. д-р Сузана Лошковска, член,  
ФИНКИ – Скопје, Р. Македонија
5. Вонр. проф. д-р Ана Мадевска Богданова, член,  
ФИНКИ – Скопје, Р. Македонија

Датум на одбрана:

Датум на промоција:

Дисертацијата е од областа на техничките науки

*Посветено на моето семејство*

**Gjorgji M. Madjarov, M.Sc.**

**Advanced methods for building hierarchical multi-label classifiers**

**ABSTRACT:** Multi-label learning has received significant attention in the research community over the past few years, motivated by an increasing number of new applications. The latter includes semantic annotation of images and video (news clips, movies clips), functional genomics (gene and protein function), music categorization into emotions, text classification (news articles, web pages, patents, e-mails, bookmarks ...), directed marketing and others. This has resulted in the development of a variety of multi-label learning methods. Previous research efforts have resulted in various methods and algorithms for solving multi-label and hierarchical multi-label classification and ranking problems. However, classification architecture with equal performance over various classification challenges has not been developed yet.

Within this thesis an attempt is made to identify the advanced problems and challenges that the researchers face in the area of multi-label and hierarchical multi-label classification and ranking. The research undertaken in the preparation of this thesis resulted with three new algorithms for multi-label classification and ranking (Two stage architecture, Hybrid Decision Tree Architecture utilizing Local SVMs and Random forest of HOMER hierarchy predictive clustering trees). They are characterized with efficiency, scalability and high predictive performance. To assess the predictive performance of the proposed methods, an extensive experimental comparison with 12 multi-label learning methods using 18 evaluation measures over 11 benchmark datasets was made. The competing methods were selected based on their previous usage by the research community, the representation of different groups of methods and the variety of basic underlying machine learning methods.

The obtained experimental results showed excellent performance regarding the training and classification speeds and high predictive performance in the recognition process.

**KEY WORDS:** multi-label classification, multi-label ranking, machine learning, pattern recognition

**М-р. Ѓорѓи М. Маџаров**

## **Нови методи за градење на хиерархиски повеќецелни класификатори**

**РЕЗИМЕ:** Во последните неколку години истражувањата во областа на повеќезначна класификација и рангирање се повеќе интензивираат како резултат на се поголемата нивна примена во различни апликациски области (класификација на слики, класификација на функција на гени, класификација на функција на протеини, класификација на текст од различни извори, класификација на мултимедијални податоци итн.) Во досегашните истражувања се предложени голем број методи за решавање на повеќезначни и хиерархиски повеќезначни класификациски проблеми. Но, сè уште не е имплементирана класификациска архитектура која со подеднакви перформанси би одговорила на сите предизвици.

Во оваа докторска дисертација е направен обид да се идентификуваат најзначајните проблеми и предизвици со кои се соочуваат истражувачите во областа на машинското учење при повеќезначна и хиерархиска повеќезначна класификација и рангирање. Во рамки на истражувањето опишано во докторската дисертација предложени се три нови методи (Дво-нивовска архитектура, хибридна архитектура базирана на дрвата за одлучување за која користи локални машини со носечки вектори и случајни шуми од дрва за предиктивно кластерирање кои користат вештачки дефинирана хиерархија) за решавање на проблемот на повеќезначна класификација и рангирање. Тие се карактеризираат со ефикасност, скалабилност и високи перформанси во однос на предвидувањата.

За да се проценат перформансите на предложените методи, спроведени се опсежни експерименти во кои е направена споредба на предложените методи со најдобрите и најзначајните 12 методи за повеќезначна класификација и рангирање. Споредбата на методите е направена во однос на 18 евалуациски мерки вклучувајќи ги и времињата на обучување и тестирање на 11 бенчмарк повеќезначни класификациски проблеми.

Добиените резултати покажуваат дека предложените методи се карактеризираат со одлични перформанси во однос на брзината на обучување и класификација и висока прецизност при процесот на препознавање.

**КЛУЧНИ  
ЗБОРОВИ:**

повеќезначна класификација, повеќезначно рангирање, машинско учење, препознавање на примероци

*Докторската дисертација е изработена на Факултетот за информатички науки и компјутерско инженерство во Скопје, Р. Македонија и на Институтот „Јозеф Стефан“ во Љубљана, Р. Словенија. На менторот Вонр. проф. д-р Дејан Ѓорѓевиќ му должам огромна благодарност за континуираната искрена поддршка, сесрдно залагање, упорност, посветеност, корисните совети и за времето и вниманието кои ми ги посвети во текот на работата на докторската дисертација.*

*Огромна благодарност би сакал да искажам и до коменторот Проф. д-р Сашо Џероски за сите конструктивни забелешки и сугестии, за долгите плодни дискусии и огромната поддршка.*

*Благодарност должам и до колегите од Институтот за софтверско инженерство и Институтот „Јозеф Стефан“, особено на колегите Драги Коцев и Томче Делев кои ме поддржаа при изработката на докторската дисертација.*

*Искрена благодарност до сопругата Анета за бескрајната љубов, трпеливост и разбирање, несебичната поддршка и постојаната помош.*

*На крај, би сакал да ја изразам својата огромна благодарност и до моите родители Тодорка и Миле и мојот брат Гроздан за бескрајната поддршка, помошта и љубовта пружени како во текот на работата на докторската дисертација, така и во животот.*

<b>ЛИСТА НА СЛИКИ.....</b>	<b>IV</b>
<b>ЛИСТА НА ТАБЕЛИ.....</b>	<b>IX</b>
<b>ГЛАВА 1   ВОВЕД.....</b>	<b>1</b>
1.1   МОТИВАЦИЈА ЗА ИЗРАБОТКА НА ДОКТОРСКАТА ДИСЕРТАЦИЈА .....	1
1.2   ПРИДОБИВКИ ОД ДОКТОРСКАТА ДИСЕРТАЦИЈА .....	3
1.3   СТРУКТУРА НА ДОКТОРСКАТА ДИСЕРТАЦИЈА .....	4
1.4   ЛИСТА НА ОБЈАВЕНИ ТРУДОВИ ПОВРЗАНИ СО ДОКТОРСКАТА ДИСЕРТАЦИЈА .....	6
<b>ГЛАВА 2   МАШИНСКО УЧЕЊЕ, КЛАСИФИКАЦИЈА И РАНГИРАЊЕ .....</b>	<b>8</b>
2.1   МАШИНСКО УЧЕЊЕ .....	8
2.2   ДЕФИНИЦИИ .....	9
2.3   КЛАСИФИКАЦИЈА .....	10
2.3.1   Повеќекласна класификација .....	10
2.3.2   Повеќезначна класификација .....	11
2.3.3   Хиерархиска повеќезначна класификација .....	12
2.4   РАНГИРАЊЕ .....	12
<b>ГЛАВА 3   ЕВАЛУАЦИСКИ МЕРКИ И ПОДАТОЧНИ МНОЖЕСТВА.....</b>	<b>14</b>
3.1   ЕВАЛУАЦИСКИ МЕРКИ .....	14
3.1.1   Евалуациски мерки базирани на примерок .....	15
3.1.2   Евалуациски мерки базирани на ознака.....	17
3.1.3   Евалуациски мерки базирани на рангирање.....	18
3.2   ПОДАТОЧНИ МНОЖЕСТВА.....	20
<b>ГЛАВА 4   ПРЕГЛЕД НА МЕТОДИ ЗА ПОВЕЌЕЗНАЧНА КЛАСИФИКАЦИЈА             И РАНГИРАЊЕ И НИВНА СПОРЕДБА .....</b>	<b>24</b>
4.1   ПРЕГЛЕД НА МЕТОДИ ЗА ПОВЕЌЕЗНАЧНА КЛАСИФИКАЦИЈА И РАНГИРАЊЕ .....	24
4.1.1   Методи кои го трансформираат проблемот .....	25
4.1.2   Методи кои го адаптираат алгоритмот.....	31
4.1.3   Методи кои градат ансамбли.....	32
<b>ГЛАВА 5   ЕКСПЕРИМЕНТАЛНА СПОРЕДБА НА МЕТОДИ ЗА             ПОВЕЌЕЗНАЧНА КЛАСИФИКАЦИЈА И РАНГИРАЊЕ .....</b>	<b>34</b>
5.1   ОНТОЛОГИЈА НА МЕТОДИ ЗА ПОВЕЌЕЗНАЧНА КЛАСИФИКАЦИЈА И РАНГИРАЊЕ .....	34
5.2   МЕТОДИ НА БИНАРНА РЕЛЕВАНТНОСТ .....	35
5.2.1   Бинарна релевантност .....	35
5.2.2   Верижна архитектура .....	35
5.3   МЕТОДИ ПАРОВИ НА ОЗНАКИ .....	37
5.3.1   Калибрирано рангирање на ознака.....	37
5.3.2   Брзо тежинско гласање за повеќезначна класификација и рангирање .....	39
5.4   МЕТОДИ МНОЖЕСТВА НА ОЗНАКИ.....	39
5.4.1   Хиерархија од повеќезначни класификатори.....	39

5.5	МЕТОДИ КОИ ГО АДАПТИРААТ АЛГОРИТАМОТ .....	41
5.5.1	$k$ - Најблиски соседи за повеќезначна класификација и рангирање.....	41
5.5.2	S4.5 за повеќезначна класификација и рангирање .....	42
5.5.3	Дрва за предиктивно кластерирање.....	43
5.6	МЕТОДИ КОИ ГРАДАТ АНСАМБЛИ.....	45
5.6.1	Ансамбли од случајни шуми .....	45
5.6.2	Случајни $k$ - множества на ознаки .....	46
5.6.3	Ансамбли од верижни архитектури (ЕСС).....	47
5.7	ЕКСПЕРИМЕНТИ.....	47
5.7.1	Експериментален дизајн .....	48
5.7.2	Резултати и дискусија.....	51
5.8	ЗАКЛУЧОК.....	61
<b>ГЛАВА 6 ДВО-НИВОВСКА АРХИТЕКТУРА ЗА ПОВЕЌЕЗНАЧНА КЛАСИФИКАЦИЈА И РАНГИРАЊЕ .....</b>		<b>63</b>
6.1	МЕТОДИ БАЗИРАНИ НА ДВО-НИВОВСКА АРХИТЕКТУРА ЗА ПОВЕЌЕЗНАЧНА КЛАСИФИКАЦИЈА И РАНГИРАЊЕ .....	63
6.1.1	Дво-нивовски метод со гласање.....	65
6.1.2	Дво-нивовски верижен метод.....	68
6.1.3	Дво-нивовски скратен верижен метод .....	70
6.2	ПРЕСМЕТКОВНА КОМПЛЕКСНОСТ НА ДВО-НИВОВСКАТА АРХИТЕКТУРА ЗА ПОВЕЌЕЗНАЧНА КЛАСИФИКАЦИЈА И РАНГИРАЊЕ .....	72
6.3	ЕКСПЕРИМЕНТИ.....	74
6.3.1	Експериментален дизајн .....	75
6.3.2	Резултати и дискусија.....	76
6.4	ЗАКЛУЧОК.....	82
<b>ГЛАВА 7 ХИБРИДНА АРХИТЕКТУРА ЗА ПОВЕЌЕЗНАЧНА КЛАСИФИКАЦИЈА И РАНГИРАЊЕ .....</b>		<b>84</b>
7.1	КОМБИНИРАЊЕ НА ДРВА ЗА ОДЛУЧУВАЊЕ И МАШИНИ СО НОСЕЧКИ ВЕКТОРИ.....	84
7.2	ЕФИКАСНА ХИБРИДНА АРХИТЕКТУРА БАЗИРАНА НА ДРВА ЗА ОДЛУЧУВАЊЕ КОИ КОРИСТАТ ЛОКАЛНИ МАШИНИ СО НОСЕЧКИ ВЕКТОРИ .....	85
7.3	ХИБРИДНИ АРХИТЕКТУРИ БАЗИРАНИ НА ПОТКАСТРЕНИ ДРВА ЗА ОДЛУЧУВАЊЕ КОИ КОРИСТАТ ЛОКАЛНИ МАШИНИ СО НОСЕЧКИ ВЕКТОРИ .....	90
7.4	ПРЕСМЕТКОВНА КОМПЛЕКСНОСТ НА ХИБРИДНИТЕ АРХИТЕКТУРИ БАЗИРАНИ НА ДРВАТА ЗА ОДЛУЧУВАЊЕ .....	90
7.4.1	Пресметковна комплексност во фазата на обучување.....	91
7.4.2	Пресметковна комплексност во фазата на тестирање.....	92
7.5	ЕКСПЕРИМЕНТИ.....	93
7.5.1	Експериментален дизајн .....	94
7.5.2	Резултати и дискусија.....	95
7.6	ЗАКЛУЧОК.....	100
<b>ГЛАВА 8 ИНТЕГРАЦИЈА НА НОМЕР И ДРВА ЗА ПРЕДИКТИВНО КЛАСТЕРИРАЊЕ .....</b>		<b>101</b>

8.1	КРЕИРАЊЕ НА ВЕШТАЧКА ХИЕРАРХИЈА .....	102
8.2	ДРВА ЗА ПРЕДИКТИВНО КЛАСТЕРИРАЊЕ ЗА ХИЕРАРХИСКА ПОВЕЌЕЗНАЧНА КЛАСИФИКАЦИЈА И РАНГИРАЊЕ .....	103
8.2.1	Дрва за предиктивно кластерирање за хиерархиска повеќезначна класификација.....	104
8.2.2	Предвидувања на ознаките .....	105
8.3	УПОТРЕБА НА АНСАМБЛОТ СЛУЧАЈНИ ШУМИ ОД PCTs ЗА ХИЕРАРХИСКА ПОВЕЌЕЗНАЧНА КЛАСИФИКАЦИЈА И МЕТОДОТ НОМЕР ЗА РЕШАВАЊЕ НА ПРОБЛЕМОТ НА ПОВЕЌЕЗНАЧНА КЛАСИФИКАЦИЈА И РАНГИРАЊЕ .....	105
8.3.1	Дизајн.....	105
8.3.2	Интеграција на методот НОМЕР и PCTs за хиерархиска повеќезначна класификација.....	107
8.4	ЕКСПЕРИМЕНТИ.....	107
8.4.1	Експериментален дизајн .....	108
8.4.2	Резултати и дискусија.....	109
8.5	ЗАКЛУЧОК.....	114
<b>ГЛАВА 9 КОМПЛЕТНА СПОРЕДБА НА МЕТОДИТЕ ЗА ПОВЕЌЕЗНАЧНА КЛАСИФИКАЦИЈА И РАНГИРАЊЕ .....</b>		<b>116</b>
<b>ГЛАВА 10 ЗАКЛУЧОК .....</b>		<b>118</b>
<b>ЛИТЕРАТУРА .....</b>		<b>121</b>
<b>ДОДАТОК А.....</b>		<b>127</b>
<b>ДОДАТОК Б.....</b>		<b>157</b>

# Листа на слики

Слика 3.1	Евалуациски мерки за повеќезначна класификација и рангирање и нивна категоризација.....	15
Слика 5.1	Онтологија на методите за повеќезначна класификација и рангирање споредени во докторската дисертација.....	35
Слика 5.2	Процедура за обучување на верижната архитектура .....	36
Слика 5.3	Процедура за тестирање на даден примерок $x$ со верижната архитектура .....	37
Слика 5.4	Пример хиерархија кај <i>HOMER</i> ( $\lambda$ - ознака, $\mu$ - мета ознака, $h$ - повеќезначен класификатор).....	40
Слика 5.5	Општ алгоритам за градење на дрва за предиктивно кластерирање.....	43
Слика 5.6	Алгоритамот <i>случајни шуми</i> за учење на ансамбли. $E$ е множеството на примероци за тренирање, $k$ е бројот на дрва, додека $f(D)$ е големината на просторот на обележја кој се користи при учење на моделот .....	45
Слика 5.7	Процедура за обучување на <i>RAkEL</i> . $m$ е бројот на модели, $k$ големината на множеството на ознаки, $\mathcal{L}$ множеството на ознаки и $E$ множеството на примероци за обучување .....	46
Слика 5.8	Процедура за тестирање на <i>RAkEL</i> . $m$ е бројот на модели, $k$ големината на множеството на ознаки, $\mathcal{L}$ множеството на ознаки, $Y_i$ множество на ознаки со големина $k$ и $M_i$ модели од <i>LP</i> ансамблот .....	47
Слика 5.9	Критични дијаграми за мерките базирани на примерок. Резултати добиени од <i>дополнителниот Немењи</i> тест со ниво на значајност од 0.5 на податочните множества за кои сите методи дале резултати .....	53
Слика 5.10	Критични дијаграми за мерките базирани на примерок. Резултати добиени од <i>дополнителниот Немењи</i> тестот со ниво на значајност од 0.5 на сите податочните множества .....	54
Слика 5.11	Критични дијаграми за мерките базирани на ознака. Резултати добиени од <i>дополнителниот Немењи</i> тест со ниво на значајност од 0.5 на податочните множества за кои сите методи дале резултати .....	56
Слика 5.12	Критични дијаграми за мерките базирани на ознака. Резултати добиени од <i>дополнителниот Немењи</i> тест со ниво на значајност од 0.5 на сите податочните множества .....	57
Слика 5.13	Критични дијаграми за мерките базирани на рангирање. Резултати добиени од <i>дополнителниот Немењи</i> тест со ниво на значајност од 0.5 на податочните множества за кои сите методи дале резултати .....	58
Слика 5.14	Критични дијаграми за мерките базирани на рангирање. Резултати добиени од <i>дополнителниот Немењи</i> тест со ниво на значајност од 0.5 на сите податочни множества .....	59

<b>Слика 5.15</b> Критични дијаграми за мерките на ефикасност. Резултати добиени од <i>дополнителниот Немењи</i> тест со ниво на значајност од 0.5 на податочните множества за кои сите методи дале резултати .....	60
<b>Слика 5.16</b> Критични дијаграми за мерките на ефикасност. Резултати добиени од <i>дополнителниот Немењи</i> тест со ниво на значајност од 0.5 на сите податочни множества.....	60
<b>Слика 6.1</b> Дво-нивовска архитектура за повеќезначна класификација и рангирање .....	64
<b>Слика 6.2</b> Процедура за обучување на <i>TSVM</i> . $E$ множество на примероци, $\mathcal{L}$ множество на ознаки .....	65
<b>Слика 6.3</b> Процедура за тестирање на даден примерок $x$ со дво-нивовскиот метод со гласање.....	67
<b>Слика 6.4</b> Процедура за обучување на <i>TSCCM</i> . $E$ множество на примероци, $\mathcal{L}$ множество на ознаки и $E'$ е множеството на примероци со проширен вектор на влезни обележја.....	69
<b>Слика 6.5</b> Процедура за тестирање на даден примерок $x$ со дво-нивовскиот верижен метод .....	70
<b>Слика 6.6</b> Процедура за обучување на <i>TSPCCM</i> . $E$ множеството на примероци, $\mathcal{L}$ множеството на ознаки и $E'$ е множеството на примероци со проширен вектор на влезни обележја.....	71
<b>Слика 6.7</b> Процедура за тестирање на даден примерок $x$ со дво-нивовскиот скратен верижен метод.....	72
<b>Слика 7.1</b> Алгоритам за градење на <i>ML-SVMMDT</i> .....	87
<b>Слика 7.2</b> Процедура за поткастрување со валидација на дрвото за одлучување по неговата изградба .....	88
<b>Слика 7.3</b> Процедура за обучување на локалните <i>SVM</i> модели .....	88
<b>Слика 7.4</b> <i>ML-SVMMDT</i> го дели оригиналното податочно множество на подмножества и обучува локален <i>SVM</i> модел за некои партиции.....	89
<b>Слика 8.1</b> Балансирани <i>k-средини</i> .....	103
<b>Слика 8.2</b> Пример хиерархија кај <i>PCTs</i> за хиерархиска повеќезначна класификација и векторот на ознаки $\mathcal{Y}_i$ .....	104
<b>Слика 8.3</b> <i>RF-HHPCT</i> (анг. <i>Random Forest - Homer Hierarchy Predictive Clustering Trees</i> ) .....	107
<b>Слика А.1</b> Критични дијаграми за мерките базирани на примерок ( <i>Хаминг загуба, точност и прецизност</i> ). Резултати добиени од <i>дополнителниот Немењи</i> тест со ниво на значајност од 0.5 на податочните множества за кои сите методи дале резултати .....	145
<b>Слика А.2</b> Критични дијаграми за мерките базирани на примерок ( <i>odsub, класификациска точност и <math>F_1</math></i> ). Резултати добиени од <i>дополнителниот Немењи</i> тест со ниво на значајност од 0.5 на податочните множества за кои сите методи дале резултати.....	146

<b>Слика А.3</b> Критични дијаграми за мерките базирани на ознака ( <i>микро прецизност, микро odsuv и микро <math>F_1</math></i> ). Резултати добиени од <i>дополнителниот Немењи тест</i> со ниво на значајност од 0.5 на податочните множества за кои сите методи дале резултати .....	147
<b>Слика А.4</b> Критични дијаграми за мерките базирани на ознака ( <i>макро прецизност, макро odsuv и макро <math>F_1</math></i> ). Резултати добиени од <i>дополнителниот Немењи тест</i> со ниво на значајност од 0.5 на податочните множества за кои сите методи дале резултати .....	148
<b>Слика А.5</b> Критични дијаграми за мерките базирани на рангирање ( <i>една грешка, опфатеност и загуба при рангирање</i> ). Резултати добиени од <i>дополнителниот Немењи тест</i> со ниво на значајност од 0.5 на податочните множества за кои сите методи дале резултати .....	149
<b>Слика А.6</b> Критични дијаграми за мерката базирана на рангирање ( <i>просечна прецизност</i> ) и времињата на обучување и тестирање. Резултати добиени од <i>дополнителниот Немењи тест</i> со ниво на значајност од 0.5 на податочните множества за кои сите методи дале резултати .....	150
<b>Слика А.7</b> Критични дијаграми за мерките базирани на примерок ( <i>Хаминг загуба, точност и прецизност</i> ). Резултати добиени од <i>дополнителниот Немењи тест</i> со ниво на значајност од 0.5 на сите податочни множества .....	151
<b>Слика А.8</b> Критични дијаграми за мерките базирани на примерок ( <i>odsuv, класификациска точност и <math>F_1</math></i> ). Резултати добиени од <i>дополнителниот Немењи тест</i> со ниво на значајност од 0.5 на сите податочни множества ..	152
<b>Слика А.9</b> Критични дијаграми за мерките базирани на ознака ( <i>микро прецизност, микро odsuv и микро <math>F_1</math></i> ). Резултати добиени од <i>дополнителниот Немењи тест</i> со ниво на значајност од 0.5 на сите податочни множества .....	153
<b>Слика А.10</b> Критични дијаграми за мерките базирани на ознака ( <i>макро прецизност, макро odsuv и макро <math>F_1</math></i> ). Резултати добиени од <i>дополнителниот Немењи тест</i> со ниво на значајност од 0.5 на сите податочни множества .....	154
<b>Слика А.11</b> Критични дијаграми за мерките базирани на рангирање ( <i>една грешка, опфатеност и загуба при рангирање</i> ). Резултати добиени од <i>дополнителниот Немењи тест</i> со ниво на значајност од 0.5 на сите податочни множества.....	155
<b>Слика А.12</b> Критични дијаграми за мерката базирана на рангирање ( <i>просечна прецизност</i> ) и времињата на обучување и тестирање. Резултати добиени од <i>дополнителниот Немењи тест</i> со ниво на значајност од 0.5 на сите податочни множества.....	156
<b>Слика Б.1</b> Зависност на перформансите на дво-нивовскиот метод со гласање во однос на прагот $t$ ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за <i>emotions</i> податочното множество.....	157
<b>Слика Б.2</b> Зависност на перформансите на дво-нивовскиот метод со гласање во однос на прагот $t$ ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за <i>scene</i> податочното множество .....	158

<b>Слика Б.3</b> Зависност на перформансите на дво-нивовскиот метод со гласање во однос на прагот $t$ ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за <i>yeast</i> податочното множество.....	159
<b>Слика Б.4</b> Зависност на перформансите на дво-нивовскиот метод со гласање во однос на прагот $t$ ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за <i>medical</i> податочното множество.....	160
<b>Слика Б.5</b> Зависност на перформансите на дво-нивовскиот метод со гласање во однос на прагот $t$ ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за <i>enron</i> податочното множество.....	161
<b>Слика Б.6</b> Зависност на перформансите на дво-нивовскиот метод со гласање во однос на прагот $t$ ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за <i>corel5k</i> податочното множество.....	162
<b>Слика Б.7</b> Зависност на перформансите на дво-нивовскиот метод со гласање во однос на прагот $t$ ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за <i>tmc2007</i> податочното множество.....	163
<b>Слика Б.8</b> Зависност на перформансите на дво-нивовскиот метод со гласање во однос на прагот $t$ ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за <i>mediamill</i> податочното множество.....	164
<b>Слика Б.9</b> Зависност на перформансите на дво-нивовскиот метод со гласање во однос на прагот $t$ ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за <i>bibtex</i> податочното множество.....	165
<b>Слика Б.10</b> Зависност на перформансите на дво-нивовскиот верижен метод во однос на прагот $t$ ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за <i>emotions</i> податочното множество.....	166
<b>Слика Б.11</b> Зависност на перформансите на дво-нивовскиот верижен метод во однос на прагот $t$ ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за <i>scene</i> податочното множество.....	167
<b>Слика Б.12</b> Зависност на перформансите на дво-нивовскиот верижен метод во однос на прагот $t$ ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за <i>yeast</i> податочното множество.....	168
<b>Слика Б.13</b> Зависност на перформансите на дво-нивовскиот верижен метод во однос на прагот $t$ ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за <i>medical</i> податочното множество.....	169
<b>Слика Б.14</b> Зависност на перформансите на дво-нивовскиот верижен метод во однос на прагот $t$ ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за <i>enron</i> податочното множество.....	170
<b>Слика Б.15</b> Зависност на перформансите на дво-нивовскиот верижен метод во однос на прагот $t$ ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за <i>corel5k</i> податочното множество.....	171
<b>Слика Б.16</b> Зависност на перформансите на дво-нивовскиот верижен метод во однос на прагот $t$ ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за <i>tmc2007</i> податочното множество.....	172

<b>Слика Б.17</b> Зависност на перформансите на дво-нивовскиот верижен метод во однос на прагот $t$ ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за <i>mediamill</i> податочното множество.....	173
<b>Слика Б.18</b> Зависност на перформансите на дво-нивовскиот верижен метод во однос на прагот $t$ ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за <i>bibtex</i> податочното множество.....	174
<b>Слика Б.19</b> Зависност на перформансите на дво-нивовскиот скратен верижен метод во однос на прагот $t$ ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за <i>emotions</i> податочното множество.....	175
<b>Слика Б.20</b> Зависност на перформансите на дво-нивовскиот скратен верижен метод во однос на прагот $t$ ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за <i>scene</i> податочното множество.....	176
<b>Слика Б.21</b> Зависност на перформансите на дво-нивовскиот скратен верижен метод во однос на прагот $t$ ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за <i>yeast</i> податочното множество.....	177
<b>Слика Б.22</b> Зависност на перформансите на дво-нивовскиот скратен верижен метод во однос на прагот $t$ ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за <i>medical</i> податочното множество.....	178
<b>Слика Б.23</b> Зависност на перформансите на дво-нивовскиот скратен верижен метод во однос на прагот $t$ ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за <i>enron</i> податочното множество.....	179
<b>Слика Б.24</b> Зависност на перформансите на дво-нивовскиот скратен верижен метод во однос на прагот $t$ ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за <i>corel5k</i> податочното множество.....	180
<b>Слика Б.25</b> Зависност на перформансите на дво-нивовскиот скратен верижен метод во однос на прагот $t$ ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за <i>tmc2007</i> податочното множество.....	181
<b>Слика Б.26</b> Зависност на перформансите на дво-нивовскиот скратен верижен метод во однос на прагот $t$ ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за <i>mediamill</i> податочното множество.....	182
<b>Слика Б.27</b> Зависност на перформансите на дво-нивовскиот скратен верижен метод во однос на прагот $t$ ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за <i>bibtex</i> податочното множество.....	183

# Листа на табели

<b>Табела 3.1</b> Опис на податочните множества во однос на бројот на примероци за тренирање и тестирање, бројот на номинални и континуални атрибути, вкупниот број на ознаки ( $Q$ ), кардиналност на ознаките ( $l_c$ ), густина на ознаките по примерок ( $l_d$ ) и вкупниот број на различни множества на ознаки во рамки на податочното множество ( $l_{sd}$ ) .....	22
<b>Табела 4.1</b> Пример за повеќезначно податочно множество .....	25
<b>Табела 4.2</b> Трансформација на податочното множество прикажано во Табела 4.1 со методите (а) <i>копија</i> и (б) <i>копија со тежински фактор</i> .....	26
<b>Табела 4.3</b> Трансформација на податочното множество прикажано во Табела 4.1 со (а) <i>select-max</i> , (б) <i>select-min</i> , (в) <i>select-random</i> и (г) <i>select-ignore</i> методите .....	27
<b>Табела 4.4</b> Трансформација на податочното множество прикажано во Табела 4.1 со методот <i>множества на ознаки</i> .....	27
<b>Табела 4.5</b> Пример за рангирање кај методот <i>множества на ознаки</i> .....	28
<b>Табела 4.6</b> Трансформација на податочното множество прикажано во Табела 4.1 со методот <i>бинарна релевантност</i> .....	29
<b>Табела 5.1</b> $p$ -вредностите за проценка на перформансите на методите за повеќезначна класификација и рангирање со <i>Фридман</i> тестот во однос на евалуациските мерки базирани на примерок. Ознаката <b>Сите</b> се однесува на $p$ -вредностите пресметани за резултатите на податочните множества за кои сите методи дале резултати. Ознаката <b>Подмножество</b> се однесува на $p$ -вредностите пресметани за резултатите на сите податочни множества .....	54
<b>Табела 5.2</b> $p$ -вредностите за проценка на перформансите на методите за повеќезначна класификација и рангирање со <i>Фридман</i> тестот во однос на евалуациските мерки базирани на ознака. Ознаката <b>Сите</b> се однесува на $p$ -вредностите пресметани за резултатите на податочните множества за кои сите методи дале резултати. Ознаката <b>Подмножество</b> се однесува на $p$ -вредностите пресметани за резултатите на сите податочни множества.....	57
<b>Табела 5.3</b> $p$ -вредностите за проценка на перформансите на методите за повеќезначна класификација и рангирање со <i>Фридман</i> тестот во однос на евалуациските мерки базирани на рангирање. Ознаката <b>Сите</b> се однесува на $p$ -вредностите пресметани за резултатите на податочните множества за кои сите методи дале резултати. Ознаката <b>Подмножество</b> се однесува на $p$ -вредностите пресметани за резултатите на сите податочни множества .....	59
<b>Табела 5.4</b> $p$ -вредностите за проценка на перформансите на методите за повеќезначна класификација и рангирање со <i>Фридман</i> тестот во однос на времињата на обучување и тестирање. Ознаката <b>Сите</b> се однесува на $p$ -	

вредностите пресметани за резултатите на податочните множества за кои сите методи дале резултати. Ознаката <b>Подмножество</b> се однесува на $p$ -вредностите пресметани за резултатите на сите податочни множества.....	60
<b>Табела 6.1</b> Вредности на параметрите $a_{brmf}$ , $t$ и $r_{real}$ за сите податочни множества за трите варијации на дво-нивовската архитектура .....	77
<b>Табела 6.2</b> Перформанси на методите ( $CLR$ , $QWML$ , $TSVM$ , $TSCCM$ и $TSPCCM$ ) во однос на евалуациските мерки базирани на примерок.....	78
<b>Табела 6.3</b> Перформанси на методите ( $CLR$ , $QWML$ , $TSVM$ , $TSCCM$ и $TSPCCM$ ) во однос на евалуациските мерки базирани на ознака .....	79
<b>Табела 6.4</b> Перформанси на методите ( $CLR$ , $QWML$ , $TSVM$ , $TSCCM$ и $TSPCCM$ ) во однос на евалуациските мерки базирани на рангирање .....	80
<b>Табела 6.5</b> Перформанси на методите ( $CLR$ , $QWML$ , $TSVM$ , $TSCCM$ и $TSPCCM$ ) во однос на времињата на обучување и тестирање мерени во секунди.....	81
<b>Табела 6.6</b> Ниво на значајност $p$ според <i>Вилкоксон тестот</i> за секоја евалуациска мерка во однос на перформансите на методите $CLR$ , $TSVM$ , $TSCCM$ и $TSPCCM$ . Знакот $>$ покажува дека првиот метод е статистички подобар од вториот (прв метод vs втор метод), додека знакот $<$ покажува дека вториот метод е статистички подобар од првиот. Вредностите на $p$ помали од 0.05 покажуваат дека меѓу двата споредувани метода постои статистички значајна разлика .....	81
<b>Табела 7.1</b> Перформанси на методите ( $BR$ , $ML-C4.5$ , $ML-SVMDT_{HL}$ , $ML-SVMDT_{F1}$ , $ML-SVMDT_{pre}$ и $ML-SVMDT_{post}$ ) во однос на евалуациските мерки базирани на примерок .....	96
<b>Табела 7.2</b> Перформанси на методите ( $BR$ , $ML-C4.5$ , $ML-SVMDT_{HL}$ , $ML-SVMDT_{F1}$ , $ML-SVMDT_{pre}$ и $ML-SVMDT_{post}$ ) во однос на евалуациските мерки базирани на ознака .....	97
<b>Табела 7.3</b> Перформанси на методите ( $BR$ , $ML-C4.5$ , $ML-SVMDT_{HL}$ , $ML-SVMDT_{F1}$ , $ML-SVMDT_{pre}$ и $ML-SVMDT_{post}$ ) во однос на евалуациските мерки базирани на рангирање .....	98
<b>Табела 7.4</b> Перформанси на методите ( $BR$ , $ML-C4.5$ , $ML-SVMDT_{HL}$ , $ML-SVMDT_{F1}$ , $ML-SVMDT_{pre}$ и $ML-SVMDT_{post}$ ) во однос на евалуациските мерки базирани на рангирање .....	99
<b>Табела 7.5</b> Ниво на значајност $p$ според <i>Вилкоксон тестот</i> за секоја евалуациска мерка во однос на перформансите на методите $CLR$ , $TSVM$ , $TSCCM$ и $TSPCCM$ . Знакот $>$ покажува дека првиот метод е статистички подобар од вториот (прв метод vs втор метод), додека знакот $<$ покажува дека вториот метод е статистички подобар од првиот. Вредностите на $p$ помали од 0.05 покажуваат дека меѓу двата споредувани метода постои статистички значајна разлика .....	99
<b>Табела 7.6</b> Број на листови и локални $BR$ модели во $ML-SVMDT$ архитектурата .....	100
<b>Табела 8.1</b> Пример за оригинално повеќезначно податочно множество.....	106

<b>Табела 8.2</b> Повеќезначно податочно множество добиено со трансформација на оригиналното повеќезначно податочно множество прикажано во Табела 8.1 со помош на алгоритмот за балансирано хиерархиско кластерирање ...	106
<b>Табела 8.3</b> Вредностите на параметарот $k$ (на <i>balanced k-means</i> алгоритмот) за сите податочни множества за кои се добиени просечно најдобрите перформанси за <i>RF-HHPCT</i> .....	110
<b>Табела 8.4</b> Перформанси на методите ( <i>HOMER</i> , <i>RF-PCT</i> и <i>RF-HHPCT</i> ) во однос на евалуациските мерки базирани на примерок (DNF – <i>Did Not Finish</i> кај <i>HOMER</i> методот означува дека тој не конструирал предиктивен модел во временски период од една недела на расположливите ресурси) .....	111
<b>Табела 8.5</b> Перформанси на методите ( <i>HOMER</i> , <i>RF-PCT</i> и <i>RF-HHPCT</i> ) во однос на евалуациските мерки базирани на примерок (DNF – <i>Did Not Finish</i> кај <i>HOMER</i> методот означува дека тој не конструирал предиктивен модел во временски период од една недела на расположливите ресурси) .....	112
<b>Табела 8.6</b> Перформанси на методите ( <i>HOMER</i> , <i>RF-PCT</i> и <i>RF-HHPCT</i> ) во однос на евалуациските мерки базирани на рангирање (DNF – <i>Did Not Finish</i> кај <i>HOMER</i> методот означува дека тој не конструирал предиктивен модел во временски период од една недела на расположливите ресурси) .....	112
<b>Табела 8.7</b> Перформанси на методите ( <i>HOMER</i> , <i>RF-PCT</i> и <i>RF-HHPCT</i> ) во однос на времињата на обучување и тестирање (DNF – <i>Did Not Finish</i> кај <i>HOMER</i> методот означува дека тој не конструирал предиктивен модел во временски период од една недела на расположливите ресурси) .....	113
<b>Табела 8.8</b> Ниво на значајност $p$ според <i>Вилкоксон тестот</i> за секоја евалуациска мерка во однос на перформансите на методите <i>RF-HHPCT</i> , <i>RF-PCT</i> и <i>HOMER</i> . Знакот $>$ покажува дека првиот метод е статистички подобар од вториот (прв метод <i>vs</i> втор метод), додека знакот $<$ покажува дека вториот метод е статистички подобар од првиот. Вредностите на $p$ помали од 0.05 покажуваат дека меѓу двата споредувани метода постои статистички значајна разлика .....	113
<b>Табела A.1</b> Перформанси на методите за повеќезначна класификација и рангирање во однос на <i>Хаминг загуба</i> евалуациската мерка (DNF – <i>Did Not Finish</i> кај методите означува дека тие не конструирале предиктивен модел во временски период од една недела на расположливите ресурси) .....	127
<b>Табела A.2</b> Перформанси на методите за повеќезначна класификација и рангирање во однос на <i>прецизност</i> евалуациската мерка (DNF – <i>Did Not Finish</i> кај методите означува дека тие не конструирале предиктивен модел во временски период од една недела на расположливите ресурси) .....	128
<b>Табела A.3</b> Перформанси на методите за повеќезначна класификација и рангирање во однос на <i>одсув</i> евалуациската мерка (DNF – <i>Did Not Finish</i> кај методите означува дека тие не конструирале предиктивен модел во временски период од една недела на расположливите ресурси) .....	129
<b>Табела A.4</b> Перформанси на методите за повеќезначна класификација и рангирање во однос на <i>точност</i> евалуациската мерка (DNF – <i>Did Not Finish</i> кај	



<b>Табела A.15</b> Перформанси на методите за повеќезначна класификација и рангирање во однос на <i>загуба при рангирање</i> евалуациската мерка (DNF – <i>Did Not Finish</i> кај методите означува дека тие не конструирале предиктивен модел во временски период од една недела на расположливите ресурси)....	141
<b>Табела A.16</b> Перформанси на методите за повеќезначна класификација и рангирање во однос на <i>просечна прецизност</i> евалуациската мерка (DNF – <i>Did Not Finish</i> кај методите означува дека тие не конструирале предиктивен модел во временски период од една недела на расположливите ресурси)....	142
<b>Табела A.17</b> Перформанси на методите за повеќезначна класификација и рангирање во однос на <i>времињата на тренирање</i> (DNF – <i>Did Not Finish</i> кај методите означува дека тие не конструирале предиктивен модел во временски период од една недела на расположливите ресурси) .....	143
<b>Табела A.18</b> Перформанси на методите за повеќезначна класификација и рангирање во однос на <i>времињата на тестирање</i> (DNF – <i>Did Not Finish</i> кај методите означува дека тие не конструирале предиктивен модел во временски период од една недела на расположливите ресурси) .....	144

# Глава 1

## Вовед

### 1.1 Мотивација за изработка на докторската дисертација

Се поголем е бројот на апликациски области (класификација на слики, класификација на функција на гени, класификација на функција на протеини, класификација на текст од различни извори, класификација на мултимедијални податоци итн.) каде можат да се применат различните системи за повеќезначна класификација и рангирање. Се поголеми се барањата за подобрување на перформансите на методите, особено од аспект на нивната пресметковна комплексност и точност на предвидување. Се поголем број на секојдневни реални проблеми се идентификуваат и препознаваат како проблеми на повеќезначна класификација и рангирање (Фејсбук<sup>1</sup>: каков жанр на филмови, книги и музика сака корисникот; кои корисници ќе му бидат предложени за пријатели, Твитер<sup>2</sup>: кои информации да се изберат и да му бидат предложени за читање и следење на корисникот, со кои компании би можела да соработува една компанија, движење на акции на берза, системи за препорака, итн.).

Поради тоа истражувањата во оваа област се интензивираат и стануваат доминантни во однос на останатите видови на класификација. Бројот на методи за повеќезначна класификација станува се поголем, а изборот на вистинскиот класификациски метод сè потежок. За разлика од повеќекласната класификација каде перформансите на секој систем се евалуираат во однос на *прецизноста*, *точноста* и *одзивот*, кај системите за повеќезначна класификација и рангирање постои широк спектар на различни евалуациски мерки. Овој голем број на евалуациски мерки кои се користат од истражувачите во оваа област, остава простор за манипулација со вистинските квалитети на системите за повеќезначна класификација и рангирање, односно да бидат истакнати предностите на одредени системи за конкретни евалуациски мерки, а вешто да бидат сокриени нивните слабости (што за системите со кои се споредуваат може да претставуваат предности). Во ниту едно истражување (литературата на која се референцираме) не се користат повеќе од пет евалуациски мерки, но она што е посимптоматично е тоа што при споредувањата на конкретен метод со конкурентните методи се користат и евалуациски мерки кои не се третираат во оригиналните истражувања на конкурентните методи. За таа цел во оваа докторска дисертација се обидовме да направиме обемна експериментална евалуација во која ќе бидат споредени најцитираните и најчесто употребувани методи за решавање на различни повеќезначни класификациски проблеми, во однос на 16 евалуациски мерки и времињата на обучување и тестирање на класификациски проблеми од 5 различни

---

<sup>1</sup> англ. Facebook

<sup>2</sup> англ. Twitter

апликативни области. Со оваа експериментална евалуација се обидовме да воспоставиме една платформа на методи (критериуми за избор на методи) за повеќезначна класификација и рангирање. Со тоа ќе бидат идентификувани најрелевантните методи за споредба во оваа област, а исто така и изборот на методи за решавање на конкретен проблем ќе биде значително олеснет.

Истражувањето е дополнително мотивирано од фактот што и покрај релативно големиот број предложени методи за препознавање на примероци и решавање на повеќезначни класификациски проблеми сеуште не е имплементирана класификациска архитектура која со подеднакви перформанси би одговорила на сите предизвици поставени од повеќезначните проблеми. Цел на истражувањето е да се дадат значајни заклучоци за квалитетот, перформансите и применливоста на постоечките класификациски методи базирани на бинарните класификатори, ансамблиите од дрва за одлучување и  $k$ -најблиски соседи при решавањето на различни повеќезначни проблеми. Целта е преку детална анализа и евалуација на секој од овие класификациски системи да се детектираат нивните главни недостатоци и да се предложат модифицирани или целосно нови алгоритми кои делумно или во целост ќе ги надминат нивните недостатоци.

Цел на истражувањето на докторската дисертацијата е развој на методи и алгоритми за ефикасно и ефективно решавање на повеќезначни класификациски проблеми. Акцентот ќе биде ставен на развивање, дизајн и имплементација на методи базирани на бинарни класификатори и дрва за одлучување. Дополнително, во оваа дисертација се очекува да се добијат и важни резултати и заклучоци во однос на успешноста и ефикасноста на предложените и конкурентните методи и системи за повеќезначна класификација и рангирање, како и да се направи целосно обединување на најрелевантните системи за оваа намена и да се констатираат нивните перформанси.

Во крајниот заклучок на докторската дисертација сакаме да ги дадеме одговорите на следните прашања:

- Кои се предностите и недостатоците на различните типови на методи за повеќезначна класификација и рангирање?
- Кои методи се поефикасни, а кои имаат подобри перформанси на предвидување?
- Кои методи се најсоодветни за решавање на конкретни типови на класификациски проблеми?
- Како влијаат карактеристиките на податочните множества (број на примероци, број на обележја, кардиналност и густина на ознаки) на перформансите на различните методи за повеќезначна класификација и рангирање?
- Дали мета-податоците кои можат да се добијат со анализа на повеќезначните класификациски проблеми можат да допринесат за поточна и поефикасна предикција на методите за повеќезначна класификација и рангирање?
- Дали постои единствен метод со најдобри перформанси од аспект на предвидување и пресметковна комплексност?

## 1.2 Придобивки од докторската дисертација

Во оваа докторска дисертација е обработена областа на повеќезначна класификација и рангирање. Направена е целосна анализа од аспект на дефинирање и поставување на проблемот на повеќезначна класификација и рангирање, категоризација на постоечките методи во оваа област, опис на методите за градење на системи за повеќезначна класификација и рангирање и нивна експериментална евалуација, анализа на добиените резултати и статистичка евалуација на повеќезначните предикции. Исто така, преку запознавање со проблемите и предизвиците со кои се соочуваат истражувачите во оваа област, како и постојните техники и методи кои се предложени во развојот на истата, голем е придонесот на докторската дисертација кон популаризацијата на оваа област на истражување.

Во продолжение се сублимирани најважните придобивки од докторската дисертација:

Прво е даден широк преглед на областа на повеќезначна класификација и рангирање. Опишани и референцирани се голем број од постоечките техники и методи за повеќезначна класификација и рангирање кои се предложени во изминатите неколку години за решавање на проблеми од различни апликациски области.

- Како резултат на анализата направена на карактеристиките на различните методи, како и начините според кои тие ги решаваат повеќезначните класификациски проблеми, дефинирана е нова онтологија за методите за повеќезначна класификација и рангирање.

Потоа, докторската дисертација дава придонес и во развојот на методи за решавање на повеќезначни класификациски проблеми. Таа предлага дизајн и имплементација на брзи и ефикасни методи за градење на класификациски архитектури за повеќезначна класификација и рангирање:

- Дво-нивовска *парови на ознаки* (анг. *pair-wise*) архитектура. Оваа архитектура ја намалува пресметковната комплексност на *парови на ознаки* методите (еден од најголемите недостатоци на овие методи) и ги подобрува нивните перформанси во однос на прецизноста на предвидување користејќи ги мета-информациите на верижната структура на нејзините класификатори. Оваа архитектура покажа најдобри резултати во споредба со останатите методи кои користат бинарни модели.
- Хибридна архитектура базирана на дрвата за одлучување која користи локални машини со носечки вектори. Нејзина главна карактеристика и предност е високата ефикасност. При решавањето на различни повеќезначни проблеми таа се одликува со високо ниво на генерализација, скалабилност, ефикасност, брзина и робустност. Оваа архитектура понуди одличен баланс на перформансите на предвидување во однос на пресметковната комплексност.
- Примена на вештачки дефинирана хиерархија за решавање на повеќезначна класификација и рангирање. Експериментално покажавме дека дефинирањето на вештачка хиерархија на целните обележја и трансформацијата на повеќезначниот класификациски проблем во хиерархиски повеќезначен класификациски проблем позитивно влијае на перформансите на предвидување.

Имајќи во вид дека не постои систем кој е способен во целост да одговори на сите поставени барања, односно не постои универзален алгоритам со идеална генерализациска способност, најбрзи времиња на обучување и предикција, како и широко спектрална скалабилност, оваа докторска дисертација дава значаен придонес во областа на повеќезначната класификација и рангирање. Деталните анализи на карактеристиките на предложените методи како и обемните и сеопфатни споредби кои се направени на нивните перформанси и перформансите на најдобрите методи за повеќезначна класификација и рангирање познати денес, даваат голем придонес кон разрешувањето на дилемата кои методи се реално најдобри за решавање на конкретни повеќезначни класификациски проблеми.

### 1.3 Структура на докторската дисертација

Во **втората** глава од докторската дисертација се претставени основните концепти на машинското учење со посебен акцент на надгледуваното учење. Во неа се опишани основните теоретски и практични концепти на машинското учење, а дополнително се опишани и најзначајните термини и поими кои се среќаваат во оваа област. Во вториот дел формално се дефинирани задачите на повеќекласна, повеќезначна и хиерархиска повеќезначна класификација (поделба која е направена врз основа на видот на проблемите кои се решаваат со помош на класификација), а дефинирана е и заедничката задача на класификацијата и рангирањето.

**Третата** глава содржи опис на податочните множества и мерките што се користени за евалуација на перформансите на методите за повеќезначна класификација и рангирање. Во првиот дел се претставени евалуациските мерки кои се поделени (групирани) според перформансите кои тие ги мерат и истите формално се опишани. Во продолжение на главата, претставени се податочните множества кои се користат за евалуација на перформансите на системите и неколку статистички мерки со кои се карактеризираат истите.

Во **четвртата** глава е даден широк преглед на постоечките техники и методи за повеќезначна класификација и рангирање предложени во изминатите неколку години. Претставени се теориските основи на кои тие почиваат и начините за нивна реализација. Дополнително, предложена е и нова поделба на методите за повеќезначна класификација и рангирање во три основни групи (методи кои го трансформираат проблемот, методи кои го адаптираат алгоритмот и методи кои градат ансамбли).

Во **петтата** глава направена е обемна експериментална евалуација и споредба на перформансите на најзначајните и најдобрите методи за повеќезначна класификација и рангирање. Споредбата на методите е направена во однос на 18 евалуациски мерки вклучувајќи ги и времињата на обучување и тестирање на 11 бенчмарк повеќезначни класификациски проблеми. Главата започнува со детален опис на методите опфатени со експерименталната евалуација, а во делот на експериментите детално се опишани експерименталниот дизајн и условите под кои се споредени методите. На крајот се прикажани резултатите добиени од експерименталната евалуација, статистичката анализа и донесените заклучоци.

Во **шестата** глава е претставена *дво-нивовска ефикасна парови на ознаки архитектура* (анг. *Two Stage Architecture - TSA*) за повеќезначна класификација и рангирање. Прво е претставена дво-нивовската архитектура, нејзиниот дизајн и

начинот на кој таа се евалуира, а потоа се претставени три различни варијанти на градење на оваа архитектура: Дво-нивовски метод со гласање, дво-нивовски верижен метод и дво-нивовски скратен верижен метод. На крај, направена е формална анализа на пресметковната комплексност на дво-нивовската архитектура. Основната идеја на оваа архитектура е да ја намали пресметковната комплексност на методите *парови на ознаки*, а притоа да ги задржи или подобри нивните перформанси во однос на прецизноста на предвидување. Главата завршува со анализа и заклучоци донесени врз основа на резултатите добиени од експерименталната евалуација на оваа архитектура.

Во **седмата** глава е претставена хибридна архитектура базирана на дрва за одлучување кои користат локални машини со носечки вектори за повеќезначна класификација и рангирање. Користејќи го методот „раздели па владеј“, основната идеја на оваа архитектура е да ја намали пресметковната комплексност на методите кои го трансформираат проблемот и да ги задржи или подобри перформансите во однос на предвидувањата. Таа прво го трансформира глобалниот повеќезначен класификациски проблем во повеќе помали локални проблеми, а потоа овие ново генерирани проблеми ги решава со употреба на локални класификатори. За решавање на локалните класификациски проблеми се користат локални машини со носечки вектори во архитектура на *бинарна релевантност* (анг. *binary relevance*), а за локализација на проблемите се користат дрва за одлучување кои се карактеристични по својата висока ефикасност. Оваа глава завршува со анализа и заклучоци донесени врз основа на резултатите добиени од експерименталната евалуација на оваа архитектура.

Имајќи ги во предвид предностите и недостатоците на различните методи за повеќезначна класификација и рангирање, во **осмата** глава се обидовме преку експериментална евалуација да покажеме дека доколку постои хиерархиска законитост на означување на примероците во рамки на податочното множество, најдобро е прво таа да ја детектираме, а потоа ново дефинираниот хиерархиски повеќезначен класификациски проблем да го решаваме со метод за хиерархиска класификација. За таа цел, во оваа глава предложивме нов метод за повеќезначна класификација и рангирање кој претставува интеграција на двата метода: методот *HOMER* и ансамблите од дрва за предиктивно кластерирање – *RF-PCT* за решавање на хиерархиски повеќезначни класификациски проблеми. Исто како и претходните глави, така и оваа глава завршува со анализа и заклучоци донесени врз основа на добиените резултати.

Во **деветтата** глава се споредени перформансите на предложените методи и најзначајните методи за повеќезначна класификација и рангирање. Споредбата е направена во однос на истите 18 евалуациски мерки кои се користени и во петтата глава на 11-те повеќезначни класификациски проблеми. На крајот на оваа глава донесени се најзначајните заклучоци за перформансите на споредуваните методи.

Во последната (**десетта**) глава се дадени заклучоците до кои се дојде врз основа на анализите, имплементацијата и експерименталните резултати на методите за повеќезначна класификација и рангирање.

## 1.4 Листа на објавени трудови поврзани со докторската дисертација

### Трудови во журналы:

- [1] G. Madjarov, D. Kocev, D. Gjorgjevikj, and S. Dzeroski, “An extensive experimental comparison of methods for multi-label learning,” *Pattern Recognition* (2012), DOI: [10.1016/j.patcog.2012.03.004](https://doi.org/10.1016/j.patcog.2012.03.004). (IF: 2.607, 5yr IF: 3.402)
- [2] G. Madjarov, D. Gjorgjevikj, and S. Džeroski, “Two stage architecture for multi-label learning,” *Pattern Recognition*, vol. 45, pp. 1019–1034, 2012. (IF: 2.607, 5yr IF: 3.402)
- [3] G. Madzarov, D. Gjorgjevikj, and I. Chorbev, “A multi-class SVM classifier utilizing binary decision tree,” *Informatica*, vol. 33, no. 2, pp. 233–241, 2009.
- [4] D. Gjorgjevikj, G. Madjarov, D. Kocev, and S. Džeroski, “Hybrid decision tree architecture utilizing local SVMs for multi-label learning,” *Neurocomputing*, (submitted).

### Трудови на конференции со зборник уреден од меѓународна издавачка куќа:

- [5] G. Madjarov and D. Gjorgjevikj, “Hybrid decision tree architecture utilizing local SVMs for multi-label classification,” in *Hybrid Artificial Intelligent Systems* (E. Corchado, V. Snasel, A. Abraham, M. Wozniak, M. Grana, and S.-B. Cho, eds.), vol. 7209 of *Lecture Notes in Computer Science*, pp. 1–12, Springer Berlin / Heidelberg, 2012.
- [6] G. Madjarov, D. Gjorgjevikj, and S. Džeroski, “Dual layer voting method for efficient multi-label classification,” in *Pattern Recognition and Image Analysis* (J. Vitrià, J. a. Sanches, and M. Hernández, eds.), vol. 6669 of *Lecture Notes in Computer Science*, pp. 232–239, 2011.
- [7] G. Madjarov, D. Gjorgjevikj, and T. Delev, “Efficient two stage voting architecture for pairwise multi-label classification,” in *AI 2010: Advances in Artificial Intelligence* (J. Li, ed.), vol. 6464 of *Lecture Notes in Computer Science*, pp. 164–173, 2011.
- [8] G. Madzarov and D. Gjorgjevikj, “Evaluation of distance measures for multi-class classification in binary svm decision tree,” in *Artificial Intelligence and Soft Computing* (L. Rutkowski, R. Scherer, R. Tadeusiewicz, L. Zadeh, and J. Zurada, eds.), vol. 6113 of *Lecture Notes in Computer Science*, pp. 437–444, 2010.

### Останато:

- [9] D. Gjorgjevikj and G. Madjarov, “Two stage classifier chain architecture for efficient pair-wise multi-label learning,” in *Proc. of IEEE International Workshop on Machine Learning for Signal Processing*, (Beijing, China), pp. 1–6, 2011.
- [10] G. Madzarov and D. Gjorgjevikj, “Multi-class classification using support vector machines in decision tree architecture,” in *Proc. of International Conference EUROCON*, (Saint Petersburg, Russia), pp. 306–313, 2009.
- [11] K. Trojancanec, G. Madjarov, S. Loskovska, and D. Gjorgjevikj, “Hierarchical classification architectures applied to magnetic resonance images,” in *Proc. of 33rd*

- International Conference on Information Technology Interfaces*, (Cavtat, Croatia), pp. 501–506, 2011.
- [12] K. Trojacanec, G. Madzarov, D. Gjorgjevikj, and S. Loskovska, “Classification of magnetic resonance images,” in *Proc. of 32nd International Conference on Information Technology Interfaces*, (Cavtat, Croatia), pp. 597–602, 2010.
- [13] G. Madjarov, D. Gjorgjevikj, and T. Delev, “Ensembles of binary SVM decision trees,” in *Proc. of 2nd ICT Innovations Conference*, (Ohrid, Macedonia), pp. 181–188, 2010.
- [14] K. Trojacanec, G. Madjarov, D. Gjorgjevikj, and S. Loskovska, “Hierarchical classification of magnetic resonance images,” in *Proc. of 13th International Multiconference Information Society*, (Ljubljana, Slovenia), pp. 51–54, 2010.
- [15] G. Madzarov, I. Dimitrovski, S. Loskovska, and D. Gjorgjevikj, “Hierarchical video classification,” in *Proc. of 9th National Conference with International Participation, ETAI*, (Ohrid, Macedonia), pp. I5–4, 2009.
- [16] G. Madzarov, D. Gjorgjevikj, and I. Dimitrovski, “A multi-class SVM classifier utilizing binary decision tree,” in *Proc. of 11th International Multiconference Information Society*, (Ljubljana, Slovenia), pp. 13–17, 2008.
- [17] G. Madzarov, D. Gorgevik, and E. Ikonomovska, “Tree of SVM for pattern recognition,” in *Proc. of 8th National Conference with International Participation, ETAI*, (Ohrid, Macedonia), pp. I6–3, 2007.

## Глава 2

# Машинско учење, класификација и рангирање

Во ова поглавје се претставени основните концепти на машинското учење со посебен акцент на учењето со учител (надгледувано учење). Во првиот дел дадени се основните теоретски и практични концепти на проблемите на класификација. Опишани се најзначајните термини и поими кои се среќаваат во областа на машинското учење и препознавањето на шаблони (мостри, облици). Во вториот дел формално се дефинирани задачите на повеќекласна, повеќезначна и хиерархиска повеќезначна класификација (поделба која е направена врз основа на видот на проблемите кои се решаваат со помош на класификација). Дополнително е дефинирана задачата на системите за рангирање како и заедничкиот проблем на класификација и рангирање.

### 2.1 Машинско учење

Појавата на дигиталните компјутери стави на располагање машини кои далеку ги надминуваат човековите способности за комплексни нумерички операции. Се постави и прашањето дали овие машини ќе се покажат подеднакво успешни и во ненумеричките операции, особено во менталните функции како размислувањето, учењето, препознавањето на примероци итн. Проблемите од оваа категорија генерално ги истражува подрачјето на *вештачка интелигенција* или *машинска интелигенција*.

Менталните процеси со кои се постигнува учењето се очигледно сложени и сè уште недоволно познати. Учењето не му е својствено само на човекот, тоа е присутно и кај животните и имало значајна улога во еволуциска смисла. Притоа, генерализацијата овозможила организмот успешно да се снајде во нови ситуации, и преземајќи слични акции или однесување, на некои кои се покажале успешни во претходното искуство со ситуации од ист или сличен тип, успешно да излезе од нив. Природно, нервниот систем на човекот е многу поефикасен во справувањето со проблемите кои се од животно значење за човекот (препознавање на опасни ситуации) отколку со оние кои не се.

Едно од најзначајните прашања кое се поставува за компјутерите уште од моментот кога тие се измислени (произведени), е дали тие можат да учат. Односно, дали е возможно да се напише програма која ќе го натера компјутерот автоматски да учи од своите искуства и да ги подобрува своите перформанси?

Замислете компјутерот да учи: од медицинските извештаи и записи за различни болести и дијагнози; за заштедата на енергија од изградбата на различни градежни објекти; за написите, статиите, колумните, информациите и да ви ги предложи токму оние кои би сакале да ги прочитате (дознаете) наутро, без да пребарувате во „океанот“ со информации. Во тој случај влијанието на компјутерот би било огромно, неговите

постигнувања би биле драматични во сите сфери од секојдневниот живот. Со една реченица, развојот на човештвото едноставно ќе нема граници.

Конечниот одговорот на поставеното прашање сеуште не е познат, но се покажа дека раниот оптимизам во поглед на способноста на компјутерите да решаваат проблеми од класата на машинската интелигенција не беше оправдан. Сеуште не е познато како да се направи компјутер чии можности за учење на широко поле ќе ги достигнат и надминат можностите за учење на човекот. Не е познато како да се направи компјутер кој во општ случај лесно и едноставно ќе ги интерпретира (препознава) одредени концепти (шаблони) и соодветно ќе ги класифицира, групира, извлекува знаење од нив итн.

Меѓутоа, секојдневно се појавуваат нови алгоритми кои покажуваат извонредни резултати кај одредени видови на учење. За проблемите како што се препознавањето на говор, текст, музика, слика, алгоритмите базирани на машинско учење ги надминуваат сите останати пристапи. Како разбирањето за компјутерите продолжува да созрева, станува неизбежно дека машинското учење ќе игра централна улога во компјутерските науки и компјутерската технологија.

Една од задачите на машинското учење е да генерира (произведе) модел од зададено множество на обележани примероци. Ваквиот модел може да се употреби за класификација на необележани, претходно невидени примероци. Ваквото својство на обучениот модел правилно да препознава и нови примероци што не биле презентирани за време на обуката е познат како способност за генерализација. Генерираниот модел може да се евалуира од аспект на: точност, разбирливост, компактност, брзина на обучување и друго. Оваа задача е позната како: дискриминација (анг. *discrimination*) или надгледувано учење или учење со учител (анг. *supervised learning*) за разлика од задачата на ненадгледуваното учење (анг. *unsupervised learning*) или кластерирање (анг. *clustering*) каде ознаките (класите) се изведуваат од неозначените примероци во процесот на учење.

Учењето претставува конструкција на модел за даден број на опсервации. Во процесот на обука се акумулира знаење од индивидуалните опсервации со помош на индукција, односно се воопштува на основа на повеќе примероци за обука. За да се конструира моделот мора да се примени индукција, односно да се генерализира (воопшти) на основа на индивидуалните опсервации и да се изгради модел кој ќе важи и надвор од специфичните услови на индивидуалните опсервации. Откако моделот е конструиран, со истиот може по пат на дедукција да се изведат нови примери кои не мора да биле меѓу опсервациите.

## 2.2 Дефиниции

Во областа на машинското учење се среќаваат неколку поими, кои од технички аспект се дефинираат на следниот начин:

*Примерок* или *мостра*, *облик* (анг. *instance*, *example*, *pattern*) претставува специфична комбинација на вредности на подредено множество од променливи. Секоја променлива исто така означена и како *атрибут*, *особина* или *обележје* (анг. *attribute*, *feature*) прима вредности од некое предефинирано множество на можни вредности што се зависни од проблемот. Ова предефинирано множество на вредности се нарекува

домен на обележјето. Во основа, обележјата по својот вид (од аспект на доменот) може да се поделат во три основни групи: бинарни обележја, номинални обележја и континуални обележја.

Бинарните обележја како што зборува и самото име, можат да примат една од само две можни вредности (на пример: има/нема, да/не, 1/0). Номиналните обележја можат да примат вредност од преброиво множество на вредности (на пример боја: црвена, жолта, зелена, или високо, средно, ниско). Во множеството на номинални вредности може, но не мора да постои релација на подреденост. Номиналните обележја од аспект на подреденоста на доменот од кој примаат вредности може да се поделат на две подгрупи: номинални обележја со дефинирана подреденост и номинални обележја без дефинирана подреденост. Континуалните обележја примаат произволна вредност најчесто репрезентирана како реален број, евентуално ограничена во даден опсег, но со непреброиво многу можни вредности.

Една или повеќе од променливите на примероците се специјални атрибути (во зависност од класификацискиот проблем кој се решава) означени како *целни обележја*. Овие обележја го опишуваат феноменот кој е од интерес, односно феноменот во однос на кој би сакале да предвидиме одредено однесување. Примероците за кои целните обележја се познати се нарекуваат означени примероци (анг. *labelled examples*). Примероците за кој целните обележја не се познати се нарекуваат *неозначени примероци* или примероци за препознавање (анг. *query examples*).

Во процесот на машинското учење се употребуваат множества на примероци (анг. *datasets*) кои можат да бидат употребени за обука или пак за тестирање односно проценка на перформансите на моделот. Множеството за обука претставува множество на означени примероци на основа на кое се гради моделот. Множеството за тестирање претставува множество на неозначени примероци со кои се тестира моделот. Кај некои видови на алгоритми за градење на модели или оптимизирање на некои од параметрите на моделот се употребува и т.н. *множество за валидација* (анг. *validation set*). Во текот на изградбата на моделот, со примероците од множество за валидација се проценуваат неговите перформанси на примероци што не ги видел (не учествуваат во обучувањето) и на основа на добиените резултати се нагудуваат некои од параметрите на моделот со цел да се подобрат перформансите на моделот.

*Модел* е пресликување од множеството на неозначени примероци во доменот на целните обележја. Моделот се користи за предвидување на вредностите на целните обележја.

## 2.3 Класификација

### 2.3.1 Повеќекласна класификација

Кога целното обележје е номинална променлива проблемот на надгледувано учење е познат како *повеќекласна класификација* (анг. *multi-class classification*). Целното обележје со номинална вредност се нарекува уште и *класа*, асоцирајќи на ознаката на класата од преброивото множество на класи на која припаѓа примерокот.

Кај проблемите на повеќекласна класификација, целта на моделот е непознатиот примерок да го распореди во точно една од конечниот број класи.

**Дадено:**

- $\mathcal{X}$  е просторот на влезни (дескриптивни) обележја каде секое влезно обележје може да има бинарна, номинална или континуална вредност.  $\mathbf{x}_i = (x_{i_1}, x_{i_2}, \dots, x_{i_D})$  е векторот на влезните обележја со димензија  $D$  на  $i$ -тиот примерок.
- $C = \{c_1, c_2, \dots, c_Q\}$  е множество на класи а  $c_i$  ( $1 \leq i \leq Q$ ) е ознака на класата
- $E$  е множеството на примероци, во кое секој примерок е составен од вектор на влезни обележја ( $\mathbf{x}_i$ ) и ознака со која примерокот е означен.  $E = \{(\mathbf{x}_i, c_i) | \mathbf{x}_i \in \mathcal{X}, c_i \in C, 1 \leq i \leq N\}$ , каде  $N = |E|$  е бројот на примероци во множеството  $E$ .
- $q$  е критериум за квалитет кој ги наградува моделите за точноста на предикциите и малата пресметковна комплексност.

**Да се најде:**

- Функција  $g: \mathcal{X} \rightarrow \{c_1, \dots, c_k\}$  која го максимизира  $q$ .

Формално, задачата на обучувањето е да конструира „обучен“ систем (модел) користејќи го множеството за обука  $E$  со цел да го класификува (или предвиди) у како излез од вака обучениот систем  $g(\mathbf{x})$ , за непознат влез во системот  $\mathbf{x}$ .

### 2.3.2 Повеќезначна класификација

За разлика од *повеќекласната*, *повеќезначната класификација* (анг. *multi-label classification*) се карактеризира со означување на примероците со повеќе целни обележја (ознаки). Имено, кај *повеќезначната класификација* постојат конечен број на целни обележја и секој примерок од повеќезначното податочно множество може да биде означен со едно или повеќе од нив. Целните обележја се нарекуваат *ознаки*.

Во случајот на *повеќезначна класификација*, целта е да се конструира модел за предвидување на множеството на релевантни ознаки на непознат (претходно невиден) примерок.

Формално задачата на *повеќезначна класификација* се опишува на следниот начин:

**Дадено:**

- $\mathcal{X}$  е просторот на влезни (дескриптивни) обележја каде секое влезно обележје може да има бинарна, номинална или континуална вредност.  $\mathbf{x}_i = (x_{i_1}, x_{i_2}, \dots, x_{i_D})$  е векторот на влезните обележја со димензија  $D$  на  $i$ -тиот примерок.
- $\mathcal{L} = \{\lambda_1, \lambda_2, \dots, \lambda_Q\}$  е множеството на ознаки а  $\lambda_i$  ( $1 \leq i \leq Q$ ) е ознака на класата. Во овој случај може да сметаме дека  $\mathcal{L}$  е составено од  $Q$  дискретни променливи (со вредности 0 и 1). Сите ознаки се меѓусебно независни.
- $E$  е множеството на примероци, во кое секој примерок е составен од вектор на влезни обележја ( $\mathbf{x}_i$ ) и множество на ознаки ( $\mathcal{Y}_i | \mathcal{Y}_i \subseteq \mathcal{L}$ ) со кое примерокот е означен.  $E = \{(\mathbf{x}_i, \mathcal{Y}_i) | \mathbf{x}_i \in \mathcal{X}, \mathcal{Y}_i \subseteq \mathcal{L}, 1 \leq i \leq N\}$ , каде  $N = |E|$  е бројот на примероци во множеството  $E$ .

- $q$  е критериум за квалитет кој ги наградува моделите за точноста на предикциите и малата пресметковна комплексност.

Да се најде:

- Функција  $h: \mathcal{X} \rightarrow 2^{\mathcal{L}}$  која го максимизира  $q$ .

### 2.3.3 Хиерархиска повеќезначна класификација

За разлика од *повеќезначната класификација* во која ознаките се независни, ознаките кај *хиерархиската повеќезначна класификација* (анг. *hierarchical multi-label classification*) се зависни и организирани во хиерархија. Тоа значи дека ако еден примерок е означен со конкретна ознака  $\lambda$ , автоматски истиот примерок е означен со сите супер-ознаки на ознаката  $\lambda$  (ова ограничување е познато под името *хиерархиско ограничување*). Дополнително, примерокот може да биде означен со ознаки кои се наоѓаат на различни патеки започнувајќи од коренот на хиерархијата. Хиерархиската повеќезначна класификација е детално опишана од Silla и Freitas [1].

Дадено:

- $\mathcal{X}$  е просторот на влезни (дескриптивни) обележја каде секое влезно обележје може да има бинарна, номинална или континуална вредност.  $\mathbf{x}_i = (x_{i_1}, x_{i_2}, \dots, x_{i_D})$  е векторот на влезните обележја со димензија  $D$  на  $i$ -тиот примерок.
- $\mathcal{Y}$  е просторот на излези, дефиниран како хиерархија на ознаки  $(\mathcal{L}, \leq_h)$ , каде  $\mathcal{L}$  е множество на ознаки, а  $\leq_h$  е структурата на хиерархијата.  $\leq_h$  всушност ја претставува зависноста на ознаките и нивните супер-ознаки ( $\forall \lambda_1, \lambda_2 \in \mathcal{L}: \lambda_1 \leq_h \lambda_2$  ако и само ако  $\lambda_1$  е супер-ознака на  $\lambda_2$ ).
- $E$  е множеството на примероци, во кое секој примерок е составен од вектор на влезни обележја ( $\mathbf{x}_i$ ) и множество на ознаки ( $\mathcal{Y}_i$ ) и го задоволува *хиерархиското ограничување*.  $E = \{(\mathbf{x}_i, \mathcal{Y}_i) | \mathbf{x}_i \in \mathcal{X}, \mathcal{Y}_i \subseteq \mathcal{L}, \lambda \in \mathcal{Y}_i \Rightarrow \forall \lambda' \leq_h \lambda: \lambda' \in \mathcal{Y}_i, 1 \leq i \leq N\}$ , каде  $N = |E|$  е бројот на примероци во множеството  $E$ .
- $q$  е критериум за квалитет кој ги наградува моделите за точноста на предикциите и малата пресметковна комплексност.

Да се најде:

- Функција  $f: \mathcal{X} \rightarrow 2^{\mathcal{L}}$  која го максимизира  $q$  и чии предикции го задоволуваат хиерархиското ограничување, т.е.  $\lambda \in f(\mathbf{x}) \Rightarrow \forall \lambda' \leq_h \lambda: \lambda' \in f(\mathbf{x})$ .

## 2.4 Рангирање

За разлика од *класификацијата* која предвидува кои ознаки (ознака) се релевантни за примероците, *рангирањето* го предвидува редоследот (рангот) на ознаките според нивната релевантност (значајност, припадност) за конкретен примерок. Формално задачата на рангирањето е да најде функција која просторот на влезни обележја  $\mathcal{X}$  за конкретен примерок  $\mathbf{x}$  ќе го преслика во редослед (ранг)  $\prec_{\mathbf{x}}$  на ознаките кои припаѓаат на множеството на ознаки  $\mathcal{L} = \{\lambda_1, \lambda_2, \dots, \lambda_Q\}$ , каде  $\lambda_i \prec_{\mathbf{x}} \lambda_j$  означува дека за

примерокот  $\mathbf{x}$ , ознаката  $\lambda_i$  е *повисоко рангирана* (позначајна, порелевантна) од ознаката  $\lambda_j$ :

**Дадено:**

- $\mathcal{X}$  е просторот на влезни (дескриптивни) обележја каде секое влезно обележје може да има бинарна, номинална или континуална вредност.  $\mathbf{x}_i = (x_{i_1}, x_{i_2}, \dots, x_{i_D})$  е векторот на влезни обележја со димензија  $D$  на  $i$ -тиот примерок.
- $\mathcal{L} = \{\lambda_1, \lambda_2, \dots, \lambda_Q\}$  е множеството на ознаки а  $\lambda_i$  ( $1 \leq i \leq Q$ ) е ознака на класата. Во овој случај може да сметаме дека  $\mathcal{L}$  е составено од  $Q$  дискретни променливи (со вредности 0 и 1). Сите ознаки се меѓусебно независни.
- $E$  е множеството на примероци, во кое секој примерок е составен од вектор на влезни обележја ( $\mathbf{x}_i$ ) и множество на ознаки ( $\mathcal{Y}_i | \mathcal{Y}_i \subseteq \mathcal{L}$ ) со кое примерокот е означен.  $E = \{(\mathbf{x}_i, \mathcal{Y}_i) | \mathbf{x}_i \in \mathcal{X}, \mathcal{Y}_i \subseteq \mathcal{L}, 1 \leq i \leq N\}$ , каде  $N$  е бројот на примероци во множеството  $E$ .
- $q$  е критериум за квалитет кој ги наградува моделите за точноста на предикциите и малата пресметковна комплексност.

**Да се најде:**

- Функција  $f: \mathcal{X} \rightarrow R$  која го максимизира  $q$ , каде  $R$  е редоследот (рангот) на ознаките за даден примерок.

Редоследот (рангирањето) на ознаките  $\lambda_i$  ( $1 \leq i \leq Q$ ) може да биде претставен како пермутација на елементите од множеството  $\mathcal{L}$ . Имено, редоследот на ознаките претставува единствена пермутација  $\tau$  на елементите од множеството  $\mathcal{L}$  така што  $\lambda_i \prec_{\mathbf{x}} \lambda_j$  ако  $\tau(\lambda_i) < \tau(\lambda_j)$  каде  $\tau(\lambda_i)$  ја дава позицијата во редоследот (рангот) на ознаката  $\lambda_i$ .

Проблемот на класификација и рангирање опфаќа рангирање на релевантните ознаки на примероците од конкретниот класификациски проблем. Овој заеднички термин најмногу се среќава кај повеќезначната класификација, па поради тоа вообичаено се користи терминологијата *повеќезначна класификација и рангирање*.

## Глава 3

# Евалуациски мерки и податочни множества

Во оваа глава се претставени евалуациските мерки кои се користат за споредба на перформансите на системите за повеќезначна класификација и рангирање. Направена е нивна поделба (групирање) според перформансите кои тие ги мерат и истите детално се опишани. Дополнително, претставени се базите на податоци кои се користат за евалуација на перформансите на системите и неколку статистички мерки со кои се карактеризираат базите на податоци. Овие статистички мерки се користат при анализата на резултатите и извлекувањето на заклучоци од добиените резултати.

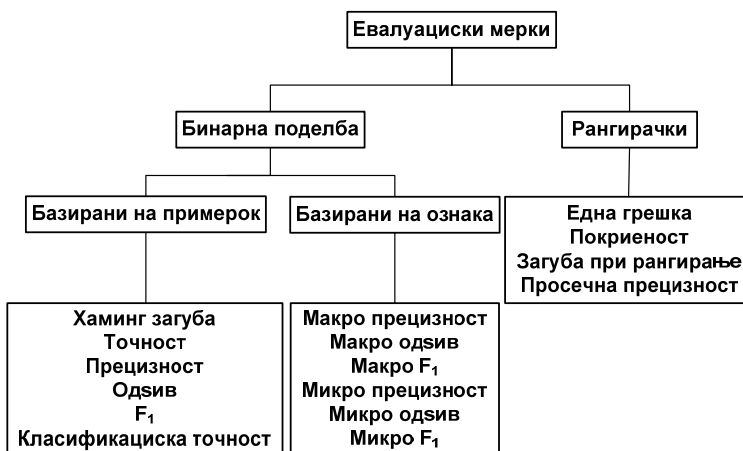
### 3.1 Евалуациски мерки

Евалуацијата на перформансите на системите за повеќезначна класификација и рангирање се разликува од евалуацијата на перформансите на класичните системи за повеќеќласна класификација. Во повеќезначната класификација и рангирање е особено значајно да се употребат повеќе различни мерки за евалуација на перформансите на системите, поради степенот на слобода кој го внесува самиот повеќезначен проблем. Во експериментите направени во докторската дисертација се користени 16 различни евалуациски мерки предложени од Tsoumakas et al. [2]. Дополнително, направена е споредба и на ефикасноста на системите за повеќезначна класификација и рангирање. Имено, измерени се времето на градење на моделите за предвидување (време на тренирање) и времето потребно да се донесе конечна одлука за множеството на ознаки на кое што му припаѓа примерок за тестирање (време на тестирање).

Евалуациските мерки се поделени во две категории: мерки на бинарна поделба и рангирачки мерки. Мерките на бинарна поделба се базираат на споредба на множествата на предвидени ознаки и релевантни ознаки. Оваа група на мерки дополнително се дели на евалуациски мерки базирани на примерок и евалуациски мерки базирани на ознака. Евалуациските мерки базирани на примерок ги мерат разликите помеѓу множествата на предвидени и релевантни ознаки за сите податочни примероци. Од измерените разлики се пресметува просек (по примерок). Од друга страна, евалуациските мерки базирани на ознака прво ги мерат перформансите за секоја ознака посебно, а потоа се пресметува просек од измерените перформанси (по ознака). Во експериментите се користат шест евалуациски мерки базирани на примерок (*Хаминг загуба* (анг. *Hamming loss*), *точност* (анг. *accuracy*), *прецизност* (анг. *precision*), *одсув* (анг. *recall*),  $F_1$  и *класификациска точност* (анг. *classification accuracy*)) и шест евалуациски мерки базирани на ознака [3] (*макро прецизност* (анг. *macro precision*), *макро одсув* (анг. *macro recall*), *макро  $F_1$*  (анг. *macro  $F_1$* ), *микро*

прецизност (анг. *micro precision*), микро одзив (анг. *micro recall*), микро  $F_1$  (анг. *micro F1*)). Треба да се забележи дека овие евалуациски мерки бараат бинарно предвидување, т.е., дали примерокот припаѓа на конкретна ознака или не (бинарни 1/0 предвидувања). Меѓутоа, моделите за предвидување најчесто предвидуваат реална вредност за секоја од ознаките (веројатност со која примерокот припаѓа на одредена ознака), па поради тоа се користи предефиниран праг  $\tau$  за да се одреди дали одредена ознака ќе биде предвидена како релевантна или не. Перформансите на моделите за предвидување директно зависат од изборот на вредноста на прагот  $\tau$ . Мерките базирани на рангирање го споредуваат предвидениот редослед (ранг) на ознаките со вистинскиот редослед (ранг) на ознаките за конкретен примерок. Во експериментите се користат четири евалуациски мерки базирани на рангирање (*една грешка* (анг. *one-error*), *покриеност* (анг. *coverage*), *загуба при рангирање* (анг. *ranking loss*) и *просечна прецизност* (анг. *average precision*)).

На Слика 3.1 се прикажани евалуациските мерки и категориите во кои тие припаѓаат. Во продолжение е даден детален опис на секоја евалуациска мерка.  $\mathcal{Y}_i$  го означува множеството на вистински ознаки за примерокот  $\mathbf{x}_i$ , додека  $h(\mathbf{x}_i)$  го означува множеството на предвидени ознаки за истиот тој примерок.



Слика 3.1 Евалуациски мерки за повеќезначна класификација и рангирање и нивна категоризација

### 3.1.1 Евалуациски мерки базирани на примерок

#### 3.1.1.1 Хаминг загуба

*Хаминг загуба* [4] покажува колку пати просечно парот примерок-ознака е неточно предвиден, т.е., предвидена е дадена ознака за соодветен примерок, иако истиот реално не припаѓа на таа класа, или, не е предвидена дадена ознака за соодветен примерок, иако истиот реално и припаѓа на таа класа. Перформансите на методите се подобри, ако вредноста на оваа евалуациска мерка е помала. Перформансите се најдобри ако  $hamming\_loss(h) = 0$ . *Хаминг загуба* се дефинира на следниот начин:

$$hamming\_loss(h) = \frac{1}{N} \sum_{i=1}^N \frac{1}{Q} |h(\mathbf{x}_i) \Delta \mathcal{Y}_i| \quad (3.1)$$

каде  $\Delta$  ја покажува симетричната разлика помеѓу две множества,  $N$  е бројот на примероци и  $Q$  е вкупниот број на можни ознаки.

### 3.1.1.2 Точност

*Точност* [5] за даден примерок  $\mathbf{x}_i$  се дефинира преку *Џакард* (анг. *Jaccard*) коефициентите на сличност помеѓу множествата на ознаки  $h(\mathbf{x}_i)$  и  $\mathcal{Y}_i$ . *Точноста* е микро-усреднета по сите примероци во податочното множество.

$$accuracy(h) = \frac{1}{N} \sum_{i=1}^N \frac{|h(\mathbf{x}_i) \cap \mathcal{Y}_i|}{|h(\mathbf{x}_i) \cup \mathcal{Y}_i|} \quad (3.2)$$

### 3.1.1.3 Прецизност

*Прецизност* [5] за даден примерок  $\mathbf{x}_i$  се дефинира како однос на бројот на точно предвидените ознаки и бројот на предвидени ознаки. *Прецизноста* е исто така микро-усреднета по сите примероци во податочното множество.

$$precision(h) = \frac{1}{N} \sum_{i=1}^N \frac{|h(\mathbf{x}_i) \cap \mathcal{Y}_i|}{|h(\mathbf{x}_i)|} \quad (3.3)$$

### 3.1.1.4 Одзив

*Одзив* [5] за даден примерок  $\mathbf{x}_i$  се дефинира како однос на бројот на точно предвидените ознаки и вкупниот број на релевантни ознаки. *Одзивот* е микро-усреднет по сите примероци во податочното множество.

$$recall(h) = \frac{1}{N} \sum_{i=1}^N \frac{|h(\mathbf{x}_i) \cap \mathcal{Y}_i|}{|\mathcal{Y}_i|} \quad (3.4)$$

### 3.1.1.5 $F_1$

$F_1$  [5] претставува хармониска средина на прецизноста и одзивот и се дефинира на следниот начин:

$$F_1(h) = \frac{1}{N} \sum_{i=1}^N \frac{2 \times |h(\mathbf{x}_i) \cap \mathcal{Y}_i|}{|h(\mathbf{x}_i)| + |\mathcal{Y}_i|} \quad (3.5)$$

$F_1$  е мерка базирана на примерок и нејзината вредност е добиена како просек од сите примероци во податочното множество.

### 3.1.1.6 Класификациска точност

Класификациска точност [6][7] е просечниот број на точно означени примероци. Таа се дефинира на следниот начин:

$$\text{classification\_accuracy}(h) = \frac{1}{N} \sum_{i=1}^N I(h(\mathbf{x}_i) = \mathcal{Y}_i) \quad (3.6)$$

каде  $I(\text{true}) = 1$  и  $I(\text{false}) = 0$ . Оваа евалуациска мерка е доста строга, затоа што бара целото множество на предвидени ознаки да биде идентично со множеството на релевантни ознаки за конкретен примерок.

## 3.1.2 Евалуациски мерки базирани на ознака

### 3.1.2.1 Макро прецизност

Макро прецизност ја мери просечната прецизност на методите по секоја од ознаките за конкретен повеќезначен класификациски проблем. Оваа мерка се дефинира на следниот начин:

$$\text{macro\_precision} = \frac{1}{Q} \sum_{j=1}^Q \frac{tp_j}{tp_j + fp_j} \quad (3.7)$$

каде  $tp_j$  и  $fp_j$  се бројот на позитивни точно предвидени и бројот на позитивни неточно предвидени вредности на ознаката  $\lambda_j$  соодветно, имајќи во предвид дека ознаката  $\lambda_j$  е бинарна.

### 3.1.2.2 Макро одсив

Макро одсивот го мери просечниот одсив на методите по секоја од ознаките за конкретен повеќезначен класификациски проблем. Оваа мерка се дефинира на следниот начин:

$$\text{macro\_recall} = \frac{1}{Q} \sum_{j=1}^Q \frac{tp_j}{tp_j + fn_j} \quad (3.8)$$

каде  $tp_j$  и  $fp_j$  се дефинирани на ист начин како кај макро прецизност, а  $fn_j$  е бројот на негативни неточно предвидени вредности на ознаката  $\lambda_j$ , повторно имајќи во предвид дека ознаката  $\lambda_j$  е бинарна.

### 3.1.2.3 Макро $F_1$

Макро  $F_1$  претставува хармониска средина на прецизноста и одсивот, така што најпрвин се пресметува просек поединечно за секоја ознака, а потоа се пресметува и

просек по сите ознаки за конкретниот повеќезначен класификациски проблем. Ако  $p_j$  и  $r_j$  се прецизноста и одсивот соодветно, на сите  $\lambda_j \in h(\mathbf{x}_i)$  од  $\lambda_j \in \mathcal{Y}_i$  макро  $F_1$  евалуациската мерка се дефинира како:

$$macro\_F_1 = \frac{1}{Q} \sum_{j=1}^Q \frac{2 \times p_j \times r_j}{p_j + r_j} \quad (3.9)$$

#### 3.1.2.4 Микро прецизност

Микро прецизност се дефинира како прецизност добиена со усреднување по паровите примерок/ознака.

$$micro\_precision = \frac{\sum_{j=1}^Q tp_j}{\sum_{j=1}^Q tp_j + \sum_{j=1}^Q fp_j} \quad (3.10)$$

каде  $tp_j$  и  $fp_j$  се дефинираат на идентичен начин како и кај евалуациската мерка макро прецизност.

#### 3.1.2.5 Микро одсив

Микро одсив се дефинира како одсив добиен со усреднување по паровите примерок/ознака.

$$micro\_recall = \frac{\sum_{j=1}^Q tp_j}{\sum_{j=1}^Q tp_j + \sum_{j=1}^Q fn_j} \quad (3.11)$$

каде  $tp_j$  и  $fn_j$  се дефинираат на идентичен начин како и кај макро одсив евалуациската мерка.

#### 3.1.2.6 Микро $F_1$

Микро  $F_1$  претставува хармониска средина на микро прецизност и микро одсив.

$$micro\_F_1 = \frac{2 \times micro\_precision \times micro\_recall}{micro\_precision + micro\_recall} \quad (3.12)$$

### 3.1.3 Евалуациски мерки базирани на рангирање

#### 3.1.3.1 Една грешка

Евалуациската мерка една грешка [4] го пресметува просечниот број на примероци, за кои најдобро рангираната ознака за секој од нив не припаѓа на множеството на релевантни ознаки на конкретниот примерок. Евалуациската мерка една грешка прима

вредности помеѓу 0 и 1. Колку оваа вредност е помала, толку перформансите се подобри. *Една грешка* се дефинира на следниот начин:

$$one\_error(f) = \frac{1}{N} \sum_{i=1}^N \left[ \left[ \arg \max_{\lambda \in \mathcal{Y}} f(\mathbf{x}_i, \lambda) \right] \notin \mathcal{Y}_i \right] \quad (3.13)$$

каде  $\lambda \in \mathcal{L} = \{\lambda_1, \lambda_2, \dots, \lambda_Q\}$  и  $\llbracket \pi \rrbracket$  е еднакво на 1 ако предикатот  $\pi$  е исполнет и 0 ако не е. Може да се забележи дека, за повеќекласна класификација евалуациската мерка *една грешка* е идентична со класичната класификациска грешка.

### 3.1.3.2 Опфатеност

*Опфатеноста* [4] покажува колку длабоко во просек е потребно да се измине листата со рангирани ознаки за да се опфатат сите релевантни ознаки за конкретен примерок. Исто како и кај претходната евалуациска мерка (*една грешка*), помалите вредности означуваат подобри перформанси.

$$coverage(f) = \frac{1}{N} \sum_{i=1}^N \max_{\lambda \in \mathcal{Y}_i} rank_f(\mathbf{x}_i, \lambda) - 1 \quad (3.14)$$

каде  $rank_f(\mathbf{x}_i, \lambda)$  го пресликува излезот на функцијата  $f(\mathbf{x}_i, \lambda)$  за секоја ознака  $\lambda \in \mathcal{L} = \{\lambda_1, \lambda_2, \dots, \lambda_Q\}$ , така што  $f(\mathbf{x}_i, \lambda_m) > f(\mathbf{x}_i, \lambda_n)$  значи  $rank_f(\mathbf{x}_i, \lambda_m) < rank_f(\mathbf{x}_i, \lambda_n)$ . Најмалата можна вредност на оваа евалуациска мерка за конкретно податочно множество е  $l_c$  (кардиналност на множеството на ознаки за сите примероци од податочното множество), т.е. просечниот број на релевантни ознаки по примерок.

### 3.1.3.3 Загуба при рангирање

Евалуациската мерка *загуба при рангирање* [4] го определува просечниот број на парови на ознаки кои се погрешно рангирани за конкретен примерок. Таа се дефинира на следниот начин:

$$ranking\_loss(f) = \frac{1}{N} \sum_{i=1}^N \frac{|D_i|}{|\mathcal{Y}_i| |\bar{\mathcal{Y}}_i|} \quad (3.15)$$

каде  $D_i = \{(\lambda_m, \lambda_n) | f(\mathbf{x}_i, \lambda_m) \leq f(\mathbf{x}_i, \lambda_n), (\lambda_m, \lambda_n) \in \mathcal{Y}_i \times \bar{\mathcal{Y}}_i\}$ , а  $\bar{\mathcal{Y}}_i$  го означува комплементарното множество на  $\mathcal{Y}_i$  во  $\mathcal{L}$ . Најдобри перформанси се постигнуваат за  $ranking\_loss(f) = 0$ . Колку вредностите на оваа евалуациска мерка се помали, толку перформансите се подобри.

### 3.1.3.4 Просечна прецизност

*Просечната прецизност* [4] го определува просечниот број на релевантни ознаки подобро рангирани од конкретна ознака  $\lambda \in \mathcal{Y}_i$ . Перформансите се најдобри кога

$avg\_precision(f) = 1$ . Поголемите вредности значат подобри перформанси. Оваа евалуациска мерка се дефинира на следниот начин:

$$avg\_precision(f) = \frac{1}{N} \sum_{i=1}^N \frac{1}{|y_i|} \sum_{\lambda \in y_i} \frac{|\mathcal{L}_i|}{rank_f(\mathbf{x}_i, \lambda)} \quad (3.16)$$

каде  $\mathcal{L}_i = \{\lambda' | rank_f(\mathbf{x}_i, \lambda') \leq rank_f(\mathbf{x}_i, \lambda), \lambda' \in y_i\}$  и  $rank_f(\mathbf{x}_i, \lambda)$  се дефинирани на идентичен начин како кај *опфатеност*.

### 3.2 Податочни множества

Во докторската дисертација се користат единаесет различни податочни множества за евалуација на перформансите на системите за повеќезначна класификација и рангирање. Истите се општо прифатени и се користат во најголем дел од истражувањата и студиите во областа на повеќезначната класификација и рангирање. Во процесот на селекција на различните податочни множества се обидовме да вклучиме податочни множества со различна големина (според бројот на примероци, бројот на ознаки, бројот на атрибути, итн.), комплексност и апликациски домен. Класификациските проблеми вклучуваат:

1. Класификација на слики: scene [8] и corel5k [9];
2. Класификација на функција на гени: yeast [10];
3. Категоризација на текст: enron [11], medical [12], bibtex [13], delicious [14], bookmarks [13] и tmc2007 [15];
4. Класификација на музика: emotions [16];
5. Класификација на видео: mediamill [17].

Кај некои апликации, бројот на релевантни ознаки по примерок може да биде помал во споредба со вкупниот број на ознаки, а кај други поголем. Оваа статистичка мерка која зависи исклучиво од видот на проблемот, претставува параметар кој може различно да влијае на перформансите на различните системи за повеќезначна класификација и рангирање. Tsoumakas et al. [2] ги воведуваат концептите на кардиналност и густина на ознаките. Имено, кардиналност на ознаките се дефинира како просечен број на ознаки по примерок во дадено податочно множество:

$$l_c = \frac{1}{N} \sum_{i=1}^N |y_i| \quad (3.17)$$

Густина на ознаките се дефинира како просечен број на ознаки по примерок во однос на вкупниот број на ознаки во податочното множество:

$$l_d = \frac{1}{N} \sum_{i=1}^N \frac{|y_i|}{Q} \quad (3.18)$$

Кардиналноста на ознаките е независна од вкупниот број на ознаки во класификацискиот проблем и се користи за да се определи бројот на ознаки со кои еден примерок е поврзан. Од друга страна, густината на ознаките го зема во предвид бројот на ознаките во класификацискиот проблем. Две податочни множества со еднаква кардиналност на ознаките и голема разлика во бројот на ознаките (различна густина на ознаки) може да предизвикаат различни перформанси кај различни системи за повеќезначна класификација и рангирање. Бројот на различни множества (групи) на ознаки кои можат да се сретнат (дефинираат) во рамки на едно податочно множество, може исто така да влијае на перформансите на различните системи за повеќезначна класификација и рангирање.

Во Табела 3.1 е дадена листата на податочните множества кои се користат за споредба на перформансите на системите за повеќезначна класификација и рангирање во оваа докторска дисертација, како и основна статистика за секое податочно множество. Може да се забележи дека при експерименталната евалуација се користи широк спектар на податочни множества: од 391 до 60000 примероци за тренирање, од 202 до 27856 примероци за тестирање, од 72 до 2150 атрибути, од 6 до 983 ознаки и кардиналност од 1.07 до 19.02.

*Emotions* [16] е повеќезначен класификациски проблем составен од примероци добиени со екстракција на музички сегменти. Сите примероци од ова податочно множество се означени со шест емоционални состојби: тажен-осамен, лут-агресивен, воодушевен-изненаден, релаксиран-мирен, тивок и среќен-задоволен.

*Scene* [8] е релативно мало, но често употребувано податочно множество кое вклучува шест можни ознаки при класификација на слики: плажа, зајдисонце, поле, зеленило, планина, и урбана средина.

*Yeast* [10] е често употребувано податочно множество од областа на биологијата каде примероците (гени) се поврзани со 14 биолошки функции.

*Medical* [12] е податочно множество што опишува текстови од областа на медицината, генерирано за натпреварот за обработка на природни јазици<sup>1</sup> организиран во 2007 година од Computational Medicine Center. Секој документ вклучува краток текст за симптомите на пациентот и е означен со ICD-9-CM<sup>2</sup> кодовите.

*Enron* [11] податочното множество е добиено од Енрон корпусот на електронски пораки<sup>3</sup>. Примероците се означени со категории организирани во хиерархија предложена од UC Berkeley Enron Email Analysis проектот<sup>4</sup>. Секоја порака е анотирана од две лица без потврда за конзистентноста и деталноста на анотациите. Сите листови од хиерархијата на овој хиерархиски повеќезначен класификациски проблем се

<sup>1</sup> <http://computationalmedicine.org/challenge/previous>

<sup>2</sup> <http://www.cdc.gov/nchs/icd/icd9cm.htm>

<sup>3</sup> <http://www.cs.cmu.edu/~enron/>

<sup>4</sup> [http://bailando.sims.berkeley.edu/enron\\_email.html](http://bailando.sims.berkeley.edu/enron_email.html)

третираат како ознаки на повеќезначниот класификациски проблем кој е дел од експерименталната евалуација на методите за повеќезначна класификација и рангирање во оваа докторска дисертација.

*Corel5k* [9] податочното множество содржи 5000 Corel слики. Во речникот за опишување (означување) на овие слики постојат 374 збора (ознаки). Секоја слика во просек има 4-5 ознаки. Сликите прво се сегментирани користејќи ја техниката *Normalized Cuts*, а потоа од секоја слика се издвоени регионите поголеми од зададен праг. Вообичаено, за секоја слика се издвоени 5-10 региони, кои потоа се кластерираат во 499 групи користејќи го *k-средини* (анг. *k-means*) алгоритмот. Кластерите се користат за опишување на сликите.

*Tmc2007* [15] податочното множество оригинално е претставено на работилницата за текстуално рударење организирана од SIAM<sup>1</sup>. Тоа содржи извештаи од воздухопловна безбедност за проблеми настанати во авионски летови. Ознаките ги претставуваат настанатите проблеми опишани во извештаите. Во оваа докторска дисертација е користена редуцираната верзија со 500 обележја предложена од Tsoumakas и Vlahavas [18].

**Табела 3.1** Опис на податочните множества во однос на бројот на примероци за тренирање и тестирање, бројот на номинални и континуални атрибути, вкупниот број на ознаки ( $Q$ ), кардиналност на ознаките ( $l_c$ ), густина на ознаките по примерок ( $l_d$ ) и вкупниот број на различни множества на ознаки во рамки на податочното множество ( $l_{sd}$ )

име	примероци		атрибути		ознаки			
	за обучување	за тестирање	номинални	континуални	$Q$	$l_c$	$l_d$	$l_{sd}$
emotions	391	202	0	72	6	1.869	0.311	27
scene	1211	1159	0	294	6	1.074	0.179	15
yeast	1500	917	0	103	14	4.237	0.303	198
medical	645	333	1449	0	45	1.245	0.028	94
enron	1123	579	1001	0	53	3.378	0.064	753
corel5k	4500	500	499	0	374	3.522	0.009	3175
tmc2007	21519	7077	500	0	22	2.158	0.098	1341
mediamill	30993	12914	0	120	101	4.376	0.043	6555
bibtex	4880	2515	1836	0	159	2.402	0.015	2856
delicious	12920	3185	500	0	983	19.02	0.019	15806
bookmarks	60000	27856	2150	0	208	2.028	0.01	18716

*Mediamill* [17] податочното множество е оригинално објавено на натпреварот NIST TRECVID<sup>2</sup> во 2005 година. Ова податочно множество е составено од анотирани видео податоци. Просторот на ознаки е претставен со 101 „анотиран концепт“, како на пример: експлозија, авион, лице, камион, урбана средина...

Податоците во *delicious* податочното множество се добиени од страната <http://delicious.com/> за колективно одбележување и споделување на информации и податоци. Податоците се добиени и претпроцесирани од Tsoumakas et al. [14].

<sup>1</sup> <http://www.cs.utk.edu/tmw07/>

<sup>2</sup> <http://www.science.uva.nl/research/mediamill/challenge/>

Интересен за ова податочно множество е фактот дека просторот на ознаки е поголем од просторот на обележја.

*Bibtex* и *bookmarks* податочните множества се користат од Katakis et al. [13] за учење на систем за автоматска сугестија на ознаки (*tags*). *Bibtex* податочното множество содржи мета-податоци како што се насловот на статијата, авторите, насловот на книгата, издавач итн., додека *bookmarks* податочното множество содржи мета-податоци за обележувачите (*bookmark items*) како што се *URL* на страната, *URL hash*, опис на страната и др.

## Глава 4

# Преглед на методи за повеќезначна класификација и рангирање и нивна споредба

Во оваа глава е даден широк преглед на постоечките техники и методи за повеќезначна класификација и рангирање предложени во изминатите неколку години. Претставени се теориските основи на кои тие почиваат и начините за нивна реализација. Дополнително, предложена е и нова поделба на методите за повеќезначна класификација и рангирање во три основни групи.

### 4.1 Преглед на методи за повеќезначна класификација и рангирање

Првиот преглед на постоечките методи за повеќезначна класификација и рангирање е направен од Tsoumakas и Katakis [19]. Врз основа на начинот на кој ги третираат и решаваат проблемите на повеќезначна класификација и рангирање, авторите на овој преглед ги делат методите во две основни групи:

- Методи кои го трансформираат проблемот
- Методи кои го адаптираат алгоритмот

Првите вршат трансформација на повеќезначниот класификациски проблем во еден или повеќе обични (анг. *single-label*) класификациски проблеми и со примена на веќе познати и општо прифатени методи за повеќецелна класификација истите ги решаваат. Потоа, со помош на некој реверзибилен процес предикциите добиени од методите за повеќецелна класификација ги трансформираат во соодветна репрезентација за повеќезначна класификација. Карактеристично за нив е тоа што можат да користат различни видови на основни класификатори за решавање на повеќецелните проблеми. Во зависност од потребите можат да бидат употребени класификатори со поголема пресметковна ефикасност или класификатори со поголема прецизност. Досега во литературата се употребени *Naive Bayes* [13][14], невронски мрежи [20], машини со носечки вектори [12] и др.

Вториот вид на методи вршат адаптација односно проширување на карактеристични класификациски методи, со цел истите да бидат директно применливи за решавање на повеќезначни класификациски проблеми. Овие методи се базирани на оригиналниот класификациски метод и не го менуваат пристапот на решавање на класификацискиот проблем. Најчесто се карактеризираат со поголема пресметковна

ефикасност, што понекогаш може да резултира со послаби перформанси во однос на квалитетот на предикциите.

Меѓутоа, во оваа поделба не е издвоена една многу значајна група на методи за повеќезначна класификација и рангирање. Тоа е групата на методите кои градат ансамбли<sup>1</sup>. Тие се разликуваат од останатите две групи според своите карактеристики, пристапот и начинот на кој ги третираат и решаваат повеќезначните проблеми. Дополнително, ансамблот може да биде составен од методи кои припаѓаат во различни групи според поделбата направена претходно. Поради тоа, ние сметаме дека оваа група на методи треба да се издвои како посебна категорија и истата да биде дел од основната категоризација на методите за повеќезначна класификација и рангирање.

#### 4.1.1 Методи кои го трансформираат проблемот

Како што споменавме претходно, овие методи вршат трансформација на повеќезначниот класификациски проблем во еден или повеќе обични (анг. *single-label*) класификациски проблеми, кои се решаваат со примена на општо прифатени методи за повеќекласна класификација.

За да се опишат подобро трансформациите кои ги применуваат овие методи во продолжение користиме едно едноставно податочно множество (Табела 4.1). Со цел да се поедностави приказот на трансформираното податочно множество, атрибутите/обележјата нема да бидат прикажани, затоа што истите не подлежат на трансформациите кои се применуваат.

Во општ случај, трансформациите на податочните множества се опишани со нотациите користени во 2.3.1 и 2.3.2.

Табела 4.1 Пример за повеќезначно податочно множество

Примерок	Атрибути/обележја	Множество на ознаки
$\mathbf{x}_1$	$x_{11}, x_{12}, \dots, x_{1D}$	$\{\lambda_1, \lambda_4\}$
$\mathbf{x}_2$	$x_{21}, x_{22}, \dots, x_{2D}$	$\{\lambda_3, \lambda_4\}$
$\mathbf{x}_3$	$x_{31}, x_{32}, \dots, x_{3D}$	$\{\lambda_1\}$
$\mathbf{x}_4$	$x_{41}, x_{42}, \dots, x_{4D}$	$\{\lambda_2, \lambda_3, \lambda_4\}$
$\mathbf{x}_5$	$x_{51}, x_{52}, \dots, x_{5D}$	$\{\lambda_1, \lambda_4\}$

Постојат неколку различни видови на трансформации кои можат оригиналното повеќезначно податочно множество да го претворат во повеќекласно податочно множество [8][21]. За да може потоа да се направи рангирање на ознаките, потребно е повеќекласниот класификатор да ја предвиди веројатноста на дистрибуција на сите ознаки. Ознаката со најголема веројатност ќе биде рангирана прва, ознаката со втора најголема веројатност ќе биде рангирана втора итн.

Првата група на трансформации, позната и како *копија* (анг. *copy, all label assignment*) трансформации прави замена на секој примерок ( $\mathbf{x}_i, \mathcal{Y}_i$ ) од повеќезначното

<sup>1</sup> Треба да се забележи дека и бинарните методи можат да се претстават како методи што градат ансамбли поради тоа што вклучуваат повеќе бинарни модели. Меѓутоа, ниту еден од овие модели не може сам да реши повеќезначен класификациски проблем па затоа терминот *ансамбл* ќе биде користен само за ансамбли од методи за повеќезначна класификација и рангирање.

податочното множество со  $|\mathcal{Y}_i|$  примероци  $(\mathbf{x}_i, \lambda_j)$  ( $\forall \lambda_j \in \mathcal{Y}_i$ ). Втората варијанта на оваа трансформација (*копија со тежински фактор*), дополнително придружува и тежински фактор  $\frac{1}{|\mathcal{Y}_i|}$  на секој ново генериран примерок. Режикиот фактор може да биде дефиниран и преку ентропијата на ознаките [21]. Со примена на оваа група на трансформации големината на податочното множество се зголемува. Секој оригинален примерок може да се појави повеќе пати означен со различна ознака. Оваа група на трансформации се обидува да ги зачува информациите за ознаките преку генерирање на нови примероци за тренирање. Меѓутоа, поради тоа што еден ист примерок може да биде означен со повеќе ознаки, може да се зголеми шумот во системот и да се намали точноста при предвидувањето. Табела 4.2 ги прикажува пример податочните множества добиени со овие трансформации од множеството прикажано во Табела 4.1.

**Табела 4.2** Трансформација на податочното множество прикажано во Табела 4.1 со методите (а) *копија* и (б) *копија со тежински фактор*

Примерок	Ознака
$\mathbf{x}_{1a}$	$\lambda_1$
$\mathbf{x}_{1b}$	$\lambda_4$
$\mathbf{x}_{2a}$	$\lambda_3$
$\mathbf{x}_{2b}$	$\lambda_4$
$\mathbf{x}_3$	$\lambda_1$
$\mathbf{x}_{4a}$	$\lambda_2$
$\mathbf{x}_{4b}$	$\lambda_3$
$\mathbf{x}_{4c}$	$\lambda_4$
$\mathbf{x}_{5a}$	$\lambda_1$
$\mathbf{x}_{5b}$	$\lambda_4$

Примерок	Ознака	Тежински фактор
$\mathbf{x}_{1a}$	$\lambda_1$	0.50
$\mathbf{x}_{1b}$	$\lambda_4$	0.50
$\mathbf{x}_{2a}$	$\lambda_3$	0.50
$\mathbf{x}_{2b}$	$\lambda_4$	0.50
$\mathbf{x}_3$	$\lambda_1$	1.00
$\mathbf{x}_{4a}$	$\lambda_2$	0.33
$\mathbf{x}_{4b}$	$\lambda_3$	0.33
$\mathbf{x}_{4c}$	$\lambda_4$	0.33
$\mathbf{x}_{5a}$	$\lambda_1$	0.50
$\mathbf{x}_{5b}$	$\lambda_4$	0.50

(а)

(б)

Втората група на трансформации, *трансформации со селекција* (анг. *select*) го заменува множеството на ознаки  $\mathcal{Y}_i$  на секој примерок со само една ознака - член на множеството  $\mathcal{Y}_i$ . Во зависност од начинот на избор на ознаката, постојат неколку различни видови трансформации со селекција. Најчесто изборот на ознаката со која ќе биде означен примерокот се прави врз основа на фреквенцијата на појавувања на ознаките во оригиналното податочното множество. Множеството на ознаки  $\mathcal{Y}_i$  на секој примерок може да се замени со ознаката  $\lambda_j \in \mathcal{Y}_i$  која има најголема фреквенција на појавувања (анг. *select-max*) или најмала фреквенција на појавувања (анг. *select-min*). Изборот на ознаката може да биде и случаен (анг. *select-random*). Ако бројот на примероци кои имаат повеќе од една ознака е процентуално мал во однос на вкупниот број на примероци, честопати овие примероци се бришат (отфрлаат) од податочното множество. Оваа трансформација е позната и како *селекција со игнорирање* (анг. *ignore*). Табела 4.3 ги прикажува податочните множества добиени со примена на *трансформациите со селекција* над податочното множество претставено во Табела 4.1.

**Табела 4.3** Трансформација на податочното множество прикажано во Табела 4.1 со (а) *select-max*, (б) *select-min*, (в) *select-random* и (г) *select-ignore* методите

Прим.	Ознака
$x_1$	$\lambda_4$
$x_2$	$\lambda_4$
$x_3$	$\lambda_1$
$x_4$	$\lambda_4$
$x_5$	$\lambda_4$

Прим.	Ознака
$x_1$	$\lambda_1$
$x_2$	$\lambda_3$
$x_3$	$\lambda_1$
$x_4$	$\lambda_2$
$x_5$	$\lambda_1$

Прим.	Ознака
$x_1$	$\lambda_1$
$x_2$	$\lambda_4$
$x_3$	$\lambda_1$
$x_4$	$\lambda_3$
$x_5$	$\lambda_4$

Прим.	Ознака
$x_3$	$\lambda_1$

(а)                      (б)                      (в)                      (г)

Пристапот *множества на ознаки* (анг. *label powerset*) [2][22][23][18] претставува едноставен но ефективен метод за трансформација на повеќезначните класификациски проблеми. Трансформацијата ја прави на следниот начин: Секое единствено множество на ознаки кое што постои во повеќезначниот класификациски проблем се претставува како нова ознака во повеќекласниот проблем. Имено, секој примерок од оригиналното податочно множество  $(x_i, Y_i)$  се заменува со нов примерок  $(x_i, L)$  каде што  $L$  е новата единствена ознака на подмножество на ознаки  $Y_i \subseteq L$ . Доколку постојат два примероци  $x_i$  и  $x_j$  со исто множество на ознаки  $Y_i = Y_j$ , во трансформираното податочно множество истите ќе бидат означени со една единствена ознака  $L$ . Табела 4.3 ги прикажува резултатите од трансформацијата *множества на ознаки* применета на податочното множество претставено во Табела 4.1.

**Табела 4.4** Трансформација на податочното множество прикажано во Табела 4.1 со методот *множества на ознаки*

Примерок	Ознака
$x_1$	$\lambda_{1,4}$
$x_2$	$\lambda_{3,4}$
$x_3$	$\lambda_1$
$x_4$	$\lambda_{2,3,4}$
$x_5$	$\lambda_{1,4}$

Повеќекласниот класификатор на методот *множества на ознаки* го означува секој нов примерок со класата која добила најголема веројатност. Класата со која е означен примерокот е всушност некое множество на ознаки од оригиналниот повеќезначен класификациски проблем. Ако повеќекласниот класификатор може да ја предвиди веројатноста на дистрибуцијата на сите класи, тогаш методот *множества на ознаки* може да направи рангирање на ознаките во повеќезначниот класификациски проблем [22]. Во Табела 4.5 прикажува пример на веројатностна дистрибуција на ознаките на непознат примерок  $x$ , добиена со методот *множества на ознаки* при решавање на повеќезначниот класификациски проблем од Табела 4.1. За да се рангира дадена ознака потребно е да се сумираат веројатностите на сите класи каде истата е содржана. На тој начин може да се прави рангирање на сите ознаки во повеќезначниот класификациски проблем користејќи го методот *множества на ознаки*.

**Табела 4.5** Пример за рангирање кај методот *множества на ознаки*

$c$	$p(c \mathbf{x})$	$\lambda_1$	$\lambda_2$	$\lambda_3$	$\lambda_4$
$\lambda_{1,4}$	0.7	1	0	0	1
$\lambda_{3,4}$	0.2	0	0	1	1
$\lambda_1$	0.1	1	0	0	0
$\lambda_{2,3,4}$	0.0	0	1	1	1
	$\sum_c p(c \mathbf{x}) \lambda_j$	<b>0.8</b>	<b>0.0</b>	<b>0.2</b>	<b>0.9</b>

Пресметковната комплексност на методот *множества на ознаки* зависи најмногу од вкупниот број на класи (бројот на различни множества на ознаки) генерирани во повеќекласниот проблем. Овој број е ограничен со  $\min(N, 2^Q)$ , каде  $N$  е бројот на примероци во множеството за тренирање, а  $Q$  е вкупниот број на ознаки во повеќезначниот класификациски проблем. Иако вистинскиот број на класи честопати е многу помал од граничниот, сепак пресметковната комплексност претставува главен проблем кај овој метод, особено за големи вредности на  $N$  и  $Q$ . Големiot број на класи и малиот број на примероци по класа претставува уште еден голем проблем кој го отежнува процесот на учење.

Read et al. [22] предлагаат метод за намалување на бројот на различни множества на ознаки кој ги селектира само оние множества на ознаки кои се појавиле повеќе пати од некоја предефинирана вредност. HOMER [14] конструира хиерархија од повеќезначни класификатори (модели). Секој класификатор е одговорен за класификациски проблем со мал број на ознаки и балансирана дистрибуција на примероци. Истиот се карактеризира со висока пресметковна ефикасност и е оригинално развиен за решавање на големи повеќезначни класификациски проблеми.

*Бинарна релевантност* (анг. *Binary Relevance (BR)*) или познат како *еден-против-сите* е еден од најчесто употребуваните методи кои припаѓаат на групата на методи за трансформација на проблемот. Овој метод обучува  $Q$  бинарни модели, по еден за секоја ознака од множеството на ознаки  $\mathcal{L}$ . Тој го трансформира оригиналното податочно множество во  $Q$  бинарни податочни множества  $E_{\lambda_j}$ ,  $j = 1 \dots Q$  кои ги содржат сите примероци од оригиналното податочно множество. Сите примероци на податочното множество  $E_{\lambda_j}$  се позитивно означени ако во оригиналното податочно множество биле означени со  $\lambda_j$ , а негативно ако не биле. Излезот на *BR* методот при класификација на даден примерок е множеството на ознаки кои се предвидени како релевантни од бинарните класификатори. Табела 4.6 ги прикажува податочните множества добиени со трансформацијата на податочното множество претставено во Табела 4.1 со методот *бинарна релевантност*.

**Табела 4.6** Трансформација на податочното множество прикажано во Табела 4.1 со методот *бинарна релевантност*

Прим.	Ознака
$x_1$	$\lambda_1$
$x_2$	$-\lambda_1$
$x_3$	$\lambda_1$
$x_4$	$-\lambda_1$
$x_5$	$\lambda_1$

(а)

Прим.	Ознака
$x_1$	$-\lambda_2$
$x_2$	$-\lambda_2$
$x_3$	$-\lambda_2$
$x_4$	$\lambda_2$
$x_5$	$-\lambda_2$

(б)

Прим.	Ознака
$x_1$	$-\lambda_3$
$x_2$	$\lambda_3$
$x_3$	$-\lambda_3$
$x_4$	$\lambda_3$
$x_5$	$-\lambda_3$

(в)

Прим.	Ознака
$x_1$	$\lambda_4$
$x_2$	$\lambda_4$
$x_3$	$-\lambda_4$
$x_4$	$\lambda_4$
$x_5$	$\lambda_4$

(г)

Godbole et al. [5] користат модификација на податочните множества за повеќезначна класификација. Всушност тие сакаат да ја најдат и претстават зависноста која постои меѓу ознаките на кои припаѓаат примероците, преку обележјата добиени од излезите на *BR* методата. Модифицираните податочни множества со новите обележја кои претставуваат некој вид на мета-податоци подлежат на уште една класификација со класификатори базирани на машините со носечки вектори. Овој метод во литературата е наречен *мета бинарна релевантност* (анг. *meta-BR*). Дополнително, тој ги модифицира податочните множества, такашто ги отстранува негативните примероци кои се слични меѓу себе и се наоѓаат многу блиску до делбената хипер-рамнина или се слични на позитивните примероци. Одредувањето на оваа сличност се прави со помош на конфузиона матрица чии членови можат да се добијат со било кој брз и прецизен класификатор со помош на валидационо множество. Tsoumakas et al. [24] прават модификација на овој метод преку отстранување на редувантните мета модели, со цел да ја подобрат неговата вкупна ефикасност.

Read et al. [12] предлагаат верижна архитектура од класификатори (анг. *classifier chains*) базирана на *BR* методот. За секој класификатор од верижната архитектура се генерира посебно податочно множество, составено од оригиналното множество на примероци проширено со дополнителни обележја за секој примерок. За конкретен класификатор, множеството на обележја на секој примерок се проширува со предикциите за примерокот добиени од сите претходни класификатори во верижната архитектура.

Yan et al. [25] предлагаат алгоритам за градење на ансамбли од *BR* за повеќезначна класификација и рангирање. Тие обучуваат бинарни модели на подмножества од примероци и обележја. На тој начин тие ја намалуваат редувантноста при обучувањето. Дополнително со користење на исти модели за различни ознаки ја намалуваат пресметковната комплексност на методот. Овој метод е поврзан со методот предложен во [24], но за разлика од него користат ансамбли наместо *мета бинарна релевантност*.

Ji et al. [26] предлагаат рамка за екстракција на заеднички структури (потпростори) кај *BR* методот. Преку овие заеднички структури, авторите се обидуваат да ја моделираат корелацијата која постои меѓу ознаките во повеќезначниот проблем. Како резултат на високата пресметковна комплексност, авторите користат само 1000 примероци во процесот на обучување на класификацискиот модел.

Ако не се земе во предвид корелацијата која постои меѓу ознаките, голема е веројатноста да се сретнеме со проблемот на неурамнотеженост на ознаките. Ráez et al. [27] се фокусираат на овој проблем преку зголемување на тежините на позитивните примероци на *BR* моделот. Тие го решаваат проблемот на категоризација на текст во реално време на голема колекција на податоци. Во истражувањето кое е направено, авторите заклучуваат дека класификациската брзина може да биде подобрена на сметка на маргинално намалување на перформансите на предвидување ако се игнорираат ретките ознаки (барем на податоците со кои тие експериментираат).

*Парови на ознаки* (анг. *pair-wise, ranking by pair-wise comparison*) [28] методот е модификација на *еден-против-еден* [29][30] методот (повеќекласна класификација) за повеќезначна класификација. Основната идеја е да се конструираат  $\binom{Q(Q-1)}{2}$  бинарни класификациски модели за секој пар од ознаки  $(\lambda_i, \lambda_j)$ ,  $1 \leq i < j \leq Q$ . Секој бинарен класификациски модел се обучува со примероците означени со  $\lambda_i$  како позитивни и примероците означени со  $\lambda_j$  како негативни. За да се добијат предикциите за секоја ознака, овој метод вообичаено го употребува *алгоритамот за мнозинско гласање*. За даден примерок, секој модел предвидува (гласа за) една од двете ознаки за кои е обучен. По евалуацијата на сите  $\frac{Q(Q-1)}{2}$  модели, ознаките се подредуваат според бројот на гласови кои ги добиле. Со употреба на алгоритам за рангирање се предвидуваат релевантните ознаки за примероците.

Brinker et al. [31] предлагаат концептуално нова техника за повеќезначна класификација и рангирање, базирана на методот *парови на ознаки* наречена *калибрирано рангирање на ознака* (анг. *Claibrated Label Ranking - CLR*). Клучна идеја на *CLR* е воведувањето на вештачка калибрирачка ознака што го дели множеството на ознаки на релевантни и нерелевантни. Слично како и методот *парови на ознаки*, *CLR* го користи алгоритамот за мнозинско гласање за предвидување на ознаките. Park et al. [32] предлагаат друг, поефикасен алгоритам за гласање наречен *брзо тежинско гласање* (анг. *Quick Weighted Voting Algorithm*), а Mencia et al. [20] го адаптираат истиот за повеќезначна класификација и рангирање.

*InsDif* алгоритамот [33] за секоја ознака од повеќезначниот проблем пресметува прототип вектори кои потоа се користат за модификација на оригиналното податочно множество. Прототип векторот за дадена ознака се пресметува со усреднување на примероците од множеството за тренирање кои се означени со ознаката. Потоа, секој примерок од оригиналното податочно множество се трансформира во  $Q$  примероци (за секоја ознака по еден), еднакви на разликата меѓу оригиналниот примерок и соодветниот прототип вектор. Вака трансформираното податочно множество се користи за обучување на дво-нивовската класификациска архитектура предложена во [34].

Barutcuoglu et al. [35] предлагаат класификациска архитектура која е базирана на машините со носечки вектори (анг. *Support Vector Machines - SVM*). За секоја ознака посебно се обучува по еден *SVM* класификатор од типот *еден-против-сите*. Излезите од сите овие класификатори се комбинираат меѓусебно користејќи Баесова мрежа, која во процесот на предикција ги зема во предвид и хиерархиските ограничувања.

#### 4.1.2 Методи кои го адаптираат алгоритамот

Adaboost.MH и Adaboost.MR [4] претставуваат две проширувања на AdaBoost [36] алгоритамот за повеќезначна класификација и рангирање. Додека Adaboost.MH е дизајниран да ја минимизира *Хаминг загубата*, Adaboost.MR ја бара хипотезата која ги рангира релевантните ознаки на врвот. Имено, излезите од сите класификатори во однос на постоечките ознаки се рангираат, при што се претпоставува дека ознаките со кои примерокот би можел да биде означен се наоѓаат на врвот од рангираната листа.

Комбинација од AdaBoost.MH и алтернативни дрва за одлучување се претставени од Comite et al. [37]. Главна мотивација е генерирањето на модели за повеќезначна класификација и рангирање кои се разбирливи за човекот.

*BP-MLL* [38] е адаптација на популарниот *пропагирање наназад* (анг. *back-propagation*) алгоритам за повеќезначна класификација и рангирање. Модификацијата се однесува на воведување на нова *erog*-функција за справување со повеќе ознаки. Повеќекласниот повеќезначен перцептрон (анг. *multi-class multi-label perceptron - MMP*) [39] е фамилија на онлајн (анг. *online*) (учење од податочни текови – анг. *data streams*) перцептрон-базирани алгоритми за рангирање на ознаки кај повеќезначни податоци. *MMP* користи по еден перцептрон за секоја ознака. Нивните тежински фактори ги нагудува со цел да се постигне перфектно рангирање на ознаките.

Предложени се повеќе методи базирани на популарниот метод  $k$  – најблиски соседи (анг.  $k$  – *Nearest Neighbors - kNN*) [40] [41] [42] [43] [44]. Првиот чекор кај сите овие методи е ист и се однесува на наоѓање на  $k$ -те најблиски соседи. Она по што тие се разликуваат е начинот на агрегација и формирање на множеството на релевантни ознаки.

Пристап што ги комбинира *мрзливото* (анг. *lazy learning*) и асоцијативното учење е предложен во [45]. Кај овој пристап, индуктивниот процес е одложен се додека примерокот не се класифицира.

Luo и Zincir-Heuywood [40] предлагаат два системи за повеќезначна класификација на документи, кои се базираат на *kNN* алгоритамот. Главниот придонес од нивната работа се состои во фазата за претпроцесирање и ефективната репрезентација на документите предвидени за класификација. При процесот на класификација на еден примерок, системот иницијално ги пронаоѓа  $k$ -те најблиски примероци. Потоа, за секоја различна ознака на овие примероци, системот дефинира соодветен бројач, кој го зголемува при секое нејзино појавување. Конечно, на тестирачкиот примерок му се придружуваат  $N$  ознаки во зависност од вредностите на бројачите. Прагот  $N$  се избира во зависност од вкупниот број на ознаки кои постојат во рамките на класификацискиот проблем.

Cesa-Bianchi, Gentile и Zaniboni [46], предлагаат инкрементален алгоритам за хиерархиска повеќезначна класификација. Класификациската архитектура која во основа претставува хиерархиско дрво се состои од бинарни класификатори со линеарен праг кои се сместени во јазлите на хиерархиското дрво. При процесот на инкрементално обучување, со секој обучувачки примерок всушност се нагудуваат параметрите на класификаторите. Дополнително, примероците во фазата на евалуација, во зависност од вредноста на класификаторите можат да бидат придружени на една или повеќе патеки во рамки на хиерархиското дрво. Истите автори [47] користат

комбинација од Баесов класификатор и машини со носечки вектори за решавање на истите класификациски проблеми.

Elisseeff и Weston [10] предлагаат рангирачки алгоритам за повеќезначна класификација и рангирање. Нивниот алгоритам ја следи филозофијата на машините со носечки вектори. Имено, тие се обидуваат да ја минимизираат *функцијата на чинење* (анг. *cost function*) со цел да добијат што е можно поголема маргина. Како функција на чинење тие користат *функција на загуба* (анг. *loss function*) според која ги рангираат ознаките.

*ММАС (Multi-class Multi-label Associative Classification)* [48] претставува алгоритам кој ја следи парадигмата на асоцијативната класификација, која го третира проблемот на генерирање на множества од класификациски правила. *ММАС* генерира иницијално множество на класификациски правила со кои се дели податочното множество на примероци кои ги задоволуваат и на примероци кои не ги задоволуваат класификациските правила. Потоа, алгоритамот рекурзивно продолжува со генерирање на нови класификациски правила од примероците кои не го задоволуваат првично генерираното множество на класификациски правила. Притоа некои од класификациските правила можат да бидат многу слични помеѓу себе, но со различна предикција. Овие правила се спојуваат и генерираат едно повеќезначно класификациско правило.

#### 4.1.3 Методи кои градат ансамбли

Ансамбл од класификатори претставува множество од класификатори чии индивидуални одлуки се комбинираат на некој начин (се обединуваат во единствена одлука), со цел подобро да се класификуваат непознатите примероци. Главното откритие во областа на комбинирањето на класификатори е што комбинацијата (ансамблот) е најчесто многу подобра од индивидуалните класификатори од кои е составена.

Потребен и доволен услов за ансамблот од класификатори да биде подобар од кој било од индивидуалните членови е индивидуалните класификатори кои го сочинуваат да бидат *прецизни* и *различни* [49][50]. Прецизен класификатор е класификатор чија рата на грешка е помала од грешката при случаен избор на класата на непознатите примероци. Два класификатори се различни ако прават некорелирани грешки за непознатите примероци. Ако индивидуалните класификатори прават некорелирани грешки со рати помали од 0.5, тогаш и нивната наједноставна комбинација (мнозинско гласање) не е полоша од најлошиот од нив. Од овде клучот за успешно комбинирање на класификаторите е да се конструираат индивидуални класификатори со рата на грешки помала од 0.5 чии грешки ќе бидат барем делумно некорелирани. Се поставува прашањето како два „исти“ класификатори да ги направиме да бидат *различни* и притоа следејќи го тоа правило да направиме ансамбл од класификатори. Постојат неколку начини а најмногу користени се следниве:

1. Промена на обучувачкото множество (или промена на тежинскиот фактор *значајност* на примероците од обучувачкото множество) [51] [52]
2. Промена на вредностите на обележјата на некои од примероците или промена на множеството на обележја [53]
3. Воведување на промени во обучувачкиот алгоритам [54].

Во последните неколку години предложени се неколку методи за градење ансамбли за повеќезначна класификација и рангирање. Најзначајните методи се: *RAkEL* [23], *ECC* [12], *EPS* [55] и *RF-PCTs* [56][57]. *RAkEL*, *ECC* и *RF-PCTs* детално ќе бидат опишани во наредната глава како претставници на методите кои градат ансамбли ансамбли во експерименталната евалуација на методите за повеќезначна класификација и рангирање.

## Глава 5

# Експериментална споредба на методи за повеќезначна класификација и рангирање

Во оваа глава е направена обемна експериментална евалуација и споредба на перформансите на најзначајните и најдобрите методи за повеќезначна класификација и рангирање. Споредбата на методите е направена во однос на 18 евалуациски мерки вклучувајќи ги и времињата на обучување и тестирање на 11 бенчмарк повеќезначни класификациски проблеми. Главата започнува со детален опис на методите опфатени со експерименталната евалуација, а во делот на експериментите, детално се опишани експерименталниот дизајн и условите под кои се споредени методите. На крајот се прикажани резултатите добиени од експерименталната евалуација, статистичката анализа и донесените заклучоци.

### 5.1 Онтологија на методи за повеќезначна класификација и рангирање

Врз основа на категоризацијата која беше предложена за методите за повеќезначна класификација и рангирање, дефиниравме онтологија на методи опфатени со експерименталната евалуација. Онтологијата е дефинирана врз основа на карактеристиките, пристапот и начинот на градење на повеќезначните модели, како и начинот на донесување на предикциите. Во првото ниво од онтологијата се наоѓаат трите категории на методи: методи кои го адаптираат алгоритмот, методи кои го трансформираат проблемот и методи кои градат ансамбли. Дополнително, во второто ниво е направена поделба на методите кои го трансформираат проблемот, според трансформацијата која тие ја прават на ознаките на повеќезначниот класификациски проблем. Конкретно дефинирани се три категории. Првата категорија ја претставуваат методите на *бинарна релевантност BR* и *CC*. Втората категорија ја сочинуваат методите *парови на ознаки CLR* и *QWML*, додека претставник на третата категорија (методите *множества на ознаки*) е *HOMER* методот. Од методите кои градат ансамбли ќе бидат споредени методите *RAkEL*, *ECC*, *RFML-C4.5* и *RF-PCT*, а *ML-C4.5*, *PCT* и *ML-kNN* се претставници на методите кои го адаптираат алгоритмот. Визуелна репрезентација на предложената онтологија е дадена на Слика 5.1.



Слика 5.1 Онтологија на методите за повеќезначна класификација и рангирање споредени во докторската дисертација

## 5.2 Методи на бинарна релевантност

### 5.2.1 Бинарна релевантност

Како што кажавме претходно методот *бинарна релевантност* (анг. *Binary Relevance - BR*) обучува  $Q$  бинарни модели, по еден за секоја ознака  $\lambda_k$  од множеството на ознаки  $\mathcal{L}$ . За дадено множество на примероци  $E = \{(\mathbf{x}_1, \mathcal{Y}_1), \dots, (\mathbf{x}_N, \mathcal{Y}_N)\}$  ( $\mathbf{x}_i \in \mathbf{X}$ ,  $\mathcal{Y}_i \subseteq \mathcal{L}$ ), секој бинарен модел  $M_k$  ( $1 \leq k \leq Q$ ) се обучува со примероците  $(\mathbf{x}_r, \mathcal{Y}_r')$  ( $r \in 1 \dots N$ ) каде што  $\mathcal{Y}_r'$  се дефинира на следниот начин:

$$\mathcal{Y}_r' = \begin{cases} +1, & \text{if } \lambda_k \in \mathcal{Y}_r \\ -1, & \text{if } \lambda_k \notin \mathcal{Y}_r \end{cases} \quad (5.1)$$

Ова значи дека секој модел  $M_k$  се обучува со сите примероци од оригиналното податочно множество, претходно трансформирани со така наречената *едно-значна трансформација* (анг. *single label transformation*). За моделот  $M_k$ , примероците се позитивно означени ако во оригиналното податочно множество биле означени со  $\lambda_k$ , а негативно ако не биле означени со  $\lambda_k$ . Моделите  $M_k$  ( $1 \leq k \leq Q$ ) се нарекуваат модели на *бинарна релевантност*.

Излезот на *BR* методот при класификација на даден примерок е множеството на ознаки кои се предвидени како релевантни од бинарните класификатори, додека при рангирање, излезот е подредено множество од сите ознаки. Подредувањето на ознаките се прави врз основа на предикциите добиени од бинарните модели.

Овој метод е еден од најчесто употребуваните методи кои припаѓаат на групата на методи за трансформација на проблемот. Се карактеризира со висока точност на предикциите и со релативно мала пресметковна комплексност (зависна од видот на моделите).

### 5.2.2 Верижна архитектура

*Верижната архитектура* (анг. *Classifier Chains - CC*) предложена од Read et al. [12] вклучува  $Q$  бинарни модели, слично како и кај архитектурата на *бинарна*

*релевантност*. Моделот  $M_k$  го решава бинарниот проблем што се однесува на ознаката  $\lambda_k \in \mathcal{L}$ . Единствената разлика со архитектурата на *бинарна релевантност* е тоа што бинарните модели во оваа архитектура се поврзани меѓусебно во така наречена верига структура (секој модел има свој претходник (освен моделот што се наоѓа на почетокот) и следбеник (освен моделот што се наоѓа на крајот на веригата)). Ова поврзување на моделите е многу значајно, затоа што просторот на обележја на податочното множество за конкретен модел се проширува со предикциите на секој претходно обучен модел (т.е., на секој примерок се додава ново обележје во секое ниво од веригата). Имено, просторот на обележја на конкретен примерок се проширува со предикциите за ознаките од моделите каде примерокот веќе се користел за обучување. Така, по обучувањето на првиот бинарен модел, моделот предвидува дали примерокот е означен (1) или не е означен (0) со ознаката ( $\lambda_k$ ) за која моделот бил обучен. Оваа предикција се додава како ново обележје на примерокот. Така променетиот примерок се користи понатаму за обучување на вториот бинарен модел од верижната архитектура. По обучувањето на вториот модел, новата предикција за примерокот се додава како негово ново обележје и истиот (сега веќе со две нови обележја) се користи при обучувањето на третиот бинарен модел. Оваа постапка се повторува се додека не се обучат бинарни модели за сите ознаки  $\lambda_k$  од множеството  $\mathcal{L}$ .

На Слика 5.2 е прикажан алгоритмот за обучување на верижната архитектура. За приказ на алгоритмот се користи следната нотација: за даден примерок  $\{(\mathbf{x}, \mathcal{Y}) \mid \mathcal{Y} \subseteq \mathcal{L}\}$ ,  $\vec{\mathcal{Y}}$  е неговиот вектор на ознаки, каде што  $\vec{\mathcal{Y}}^\lambda$  е компонентата која покажува дали примерокот  $\mathbf{x}$  е означен со ознаката  $\lambda$  или не ( $\vec{\mathcal{Y}}^\lambda = 1$  ако примерокот е означен со ознаката  $\lambda$ , а  $\vec{\mathcal{Y}}^\lambda = 0$  ако примерокот не е означен со ознаката  $\lambda$ ). Предвидувањето на моделот  $M_k$  ( $prediction(M_k, \mathbf{x})$ ) може да има две вредности 0 (примерокот  $\mathbf{x}$  не е означен со ознаката  $\lambda_k$ ) и 1 (примерокот  $\mathbf{x}$  е означен со ознаката  $\lambda_k$ ).

```
procedure Training_CC( $E = \{(\mathbf{x}_1, \vec{\mathcal{Y}}_1), \dots, (\mathbf{x}_N, \vec{\mathcal{Y}}_N)\}$ )
1: for  $k \in 1 \dots Q$  do
2:    $E_k = SingleLabelTransformation(E, \lambda_k)$ 
3:    $M_k = TrainingModel(E_k)$ 
4:   for  $(\mathbf{x}, \mathcal{Y}) \in E$  do
5:      $\mathbf{x} \leftarrow \mathbf{x} \cup \{prediction(M_k, \mathbf{x})\}$ 
```

Слика 5.2 Процедура за обучување на верижната архитектура

Процесот на тестирање на даден примерок  $\mathbf{x}$  започнува од моделот кој прв е обучен. Примерокот поминува низ сите модели по истиот редослед по кој тие се обучени. Притоа за секој нареден модел, како нови обележја на примерокот се додаваат предикциите од сите претходни модели, слично како и во процесот на обучување. Алгоритмот за тестирање на даден примерок  $\mathbf{x}$  е прикажан на Слика 5.3.

Слично како и кај *BR* методот, при класификација на даден примерок кај *CC* методот излез е множеството на ознаки кои се предвидени како релевантни од бинарните класификатори, додека при рангирање, излезот е подредено множество од сите ознаки. Подредувањето на ознаките се прави врз основа на предикциите добиени од бинарните модели.

---

```

procedure Testing_CC(x)
1:  $\vec{Y} \leftarrow \{\}$ 
1: for  $k \in 1 \dots Q$  do
2:    $\vec{Y} \leftarrow \vec{Y} \cup \{\text{prediction}(M_k, \mathbf{x})\}$ 
3:    $\mathbf{x} \leftarrow \mathbf{x} \cup \{\text{prediction}(M_k, \mathbf{x})\}$ 
    
```

---

Слика 5.3 Процедура за тестирање на даден примерок  $\mathbf{x}$  со верижната архитектура

Како резултат на мета-информациите кои се користат во процесите на обучување и тестирање, се очекува, овој метод да покаже подобри перформанси во однос на точноста на предвидувањата, додека неговата пресметковна комплексност да биде многу слична споредено со *BR* архитектурата.

## 5.3 Методи парови на ознаки

### 5.3.1 Калибрирано рангирање на ознака

Конвенционалниот метод *парови на ознаки* [28] конструира  $\binom{Q(Q-1)}{2}$  бинарни класификациски модели  $M_{ij}$  за секој пар од ознаки  $(\lambda_i, \lambda_j)$ ,  $1 \leq i < j \leq Q$ . За дадено множеството на примероци  $E = \{(\mathbf{x}_1, \mathcal{Y}_1), \dots, (\mathbf{x}_N, \mathcal{Y}_N)\}$  ( $\mathbf{x}_i \in \mathbf{X}$ ,  $\mathcal{Y}_i \subseteq \mathcal{L}$ ), секој бинарен модел  $M_{ij}$  ( $1 \leq i < j \leq Q$ ) се обучува со примероците  $(\mathbf{x}_r, \mathcal{Y}_r')$  ( $1 \leq r \leq N$ ) каде што  $\mathcal{Y}_r'$  се дефинира на следниот начин:

$$\mathcal{Y}_r' = \begin{cases} +1, & \text{if } \lambda_i \in \mathcal{Y}_r \text{ and } \lambda_j \notin \mathcal{Y}_r \\ -1, & \text{if } \lambda_j \in \mathcal{Y}_r \text{ and } \lambda_i \notin \mathcal{Y}_r \end{cases} \quad (5.2)$$

Ова значи дека секој бинарен класификациски модел  $M_{ij}$  се обучува со примероците означени со  $\lambda_i$  како позитивни и примероците означени со  $\lambda_j$  како негативни. Примероците кои истовремено се означени со ознаките  $\lambda_i$  и  $\lambda_j$  не учествуваат во обучувањето на моделот  $M_{ij}$ . Оваа трансформација на оригиналното податочно множество се нарекува *двозначна трансформација* (анг. *two label transformation*). Моделите  $M_{ij}$  ( $1 \leq i < j \leq Q$ ) се нарекуваат модели на *парови на ознаки*.

За добивање на предикциите за секоја ознака, методот *парови на ознаки* го употребува *алгоритамот за мнозинско гласање*. За даден примерок, секој модел  $M_{ij}$  ја предвидува (гласа за) една од двете ознаки за кои е обучен. По евалуацијата на сите  $\binom{Q(Q-1)}{2}$  модели, ознаките се подредуваат според броевите на гласови кои ги добиле, со што процесот на подредување на ознаките е завршен. За да се направи поделба на ознаките во множество на релевантни и множество на нерелевантни ознаки се користат прагови на одлука чии вредности се определуваат со соодветни алгоритми.

*Калибрирано рангирање на ознака* (анг. *Calibrated Label Ranking - CLR*) [31] е концептуално нова техника за повеќезначна класификација и рангирање што се базира

на методот *парови на ознаки*. Клучната разлика со методот *парови на ознаки* е тоа што покрај рангирањето, *CLR* ги класифицира (дели) ознаките на релевантни и нерелевантни. Класификацијата (поделбата) на ознаките се прави со помош на *вештачка калибрациона ознака*  $\lambda_0$ , која, слично како и останатите (оригинални) ознаки  $(\lambda_1, \dots, \lambda_Q)$  се рангира во процесот на рангирање, а потоа се користи за поделба на ознаките на релевантни и нерелевантни. Од рангираните ознаки  $\lambda_{i_1} > \dots > \lambda_{i_j} > \lambda_0 > \lambda_{i_{j+1}} > \dots > \lambda_{i_Q}$ , сите ознаки кои се рангирани пред *вештачката калибрациона ознака*  $\lambda_0$  се прогласуваат за релевантни, а сите ознаки кои се рангирани по  $\lambda_0$  се прогласуваат за нерелевантни. На овој начин методот *калибрирано рангирање на ознака* овозможува рангирање на ознаките

$$\lambda_{i_1} > \dots > \lambda_{i_j} > \lambda_{i_{j+1}} > \dots > \lambda_{i_Q} \quad (5.3)$$

и нивна поделба на релевантни и нерелевантни.

$$P = \{\lambda_{i_1} > \dots > \lambda_{i_j}\} \text{ и } N = \{\lambda_{i_{j+1}} > \dots > \lambda_{i_Q}\} \quad (5.4)$$

По дефиниција, за секоја ознака кај методот *парови на ознаки* се обучуваат по  $Q - 1$  модели на *парови на ознаки*. Тоа значи, дека со воведувањето на *вештачката калибрациона ознака*  $\lambda_0$  кај *CLR*, потребно е дополнително да се обучат  $Q$  нови модели на *парови на ознаки*. Според Brinker et al. [31] и Furnkranz et al. [58], овие  $Q$  модели на *парови на ознаки* не претставуваат ништо друго, туку моделите  $M_k$  од методот *бинерна релевантност*. На концептуално ниво, овие модели во *CLR* методот се претставени како модели на *парови на ознаки*  $M_{k0}$  на тој начин што за секој модел  $M_k$ , примерокот кој не е означен со ознаката  $\lambda_k$  се означува со вештачката калибрациона ознака  $\lambda_0$ . Имено, за моделот на *парови на ознаки*  $M_{k0}$  што е идентичен со моделот на *бинерна релевантност*  $M_k$ , сите примероци од оригиналното податочно множество кои се означени со ознаката  $\lambda_k$  се разгледуваат како позитивни примероци, а сите останати прво се означуваат со вештачката калибрациона ознака  $\lambda_0$ , а потоа се разгледуваат како негативни примероци. Треба да се забележи дека за секој различен модел  $M_{k0}$ , множеството на примероци означено со вештачката калибрациона ознака  $\lambda_0$  е различно.

Исто како и методот *парови на ознаки*, во процесот на предвидување *CLR* го користи *алгоритмот за мнoзинско гласање*. За да се донесе финалната одлука за означување на даден примерок потребно е да се консултираат  $\frac{Q(Q-1)}{2} + Q$  модели. Како резултат на зголемениот број на модели кои треба да се евалуираат, *CLR* се карактеризира со зголемена пресметковна комплексност во однос на методот *парови на ознаки*.

Големиот број на ознаки во повеќезначниот класификациски проблем може значително да влијае на пресметковната комплексност на овој метод, слично како и кај методот *парови на ознаки*.

### 5.3.2 Брзо тежинско гласање за повеќезначна класификација и рангирање

За да направи рангирање на ознаките, алгоритмот за мнозинско гласање кој се користи кај методот *парови на ознаки* ги евалуира сите  $\frac{Q(Q-1)}{2}$  обучени модели. Ако бројот на ознаки во класификацискиот проблем е голем, пресметковната ефикасност на методот во процесот на предикција може значително да биде намалена поради квадратната зависност која постои меѓу бројот на модели и бројот на ознаки. За таа цел Park et al. [32] предложуваат поефикасен алгоритам за гласање наречен *брзо тежинско гласање* (анг. *Quick Weighted Voting Algorithm*). Овој метод, кој е оригинално предложен за повеќецелна класификација го искористува фактот дека во еден момент во процесот на евалуација, одредени ознаки немаат шанса да бидат прво рангирани.

Имено, максималниот број на гласови кои може да ги добие една ознака  $\lambda$  е  $Q - 1$  (моделите кои се обучени со примероците означени со оваа ознака). Ако разликата во бројот на гласови на ознаките  $\lambda_i$  и  $\lambda_j$  ( $\lambda_i$  е прво рангирана) е поголем од бројот на модели обучени со ознаката  $\lambda_j$  кои не се евалуирани, тогаш ознаката  $\lambda_j$  не може да ја надмине ознаката  $\lambda_i$  по бројот на гласови, дури и да ги добие сите гласови од сите нејзини модели кои сеуште не се евалуирани. Сите модели кои се обучени со примероци означени со ознаки кои немаат шанса да бидат прво рангирани, во процесот на тестирање истите не се евалуираат. На тој начин се зголемува ефикасноста на методот *парови на ознаки* во процесот на тестирање.

Mencia et al. [20] го адаптираат алгоритмот на *брзо тежинско гласање* за повеќезначна класификација и рангирање (*QWML*). Тие ја прилагодуваат оваа стратегија на гласање за *CLR* методот. Имено, за секој примерок, алгоритмот за *брзо тежинско гласање* се повторува се додека не се определат сите релевантни ознаки, односно се додека не се врати вештачката калибрациона ознака  $\lambda_0$ , што значи дека сите останати ознаки се нерелевантни за конкретниот примерок.

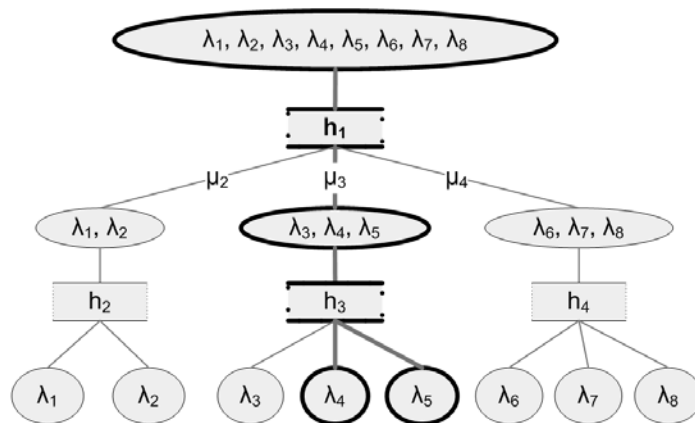
## 5.4 Методи множества на ознаки

### 5.4.1 Хиерархија од повеќезначни класификатори

*HOMER* (анг. *Hierarchy Of Multi-label classifiers*) [14] е метод за повеќезначна класификација и рангирање, кој од аспект на дизајн ја следи парадигмата *раздели на владеј*. Се користи за решавање на повеќезначни класификациски проблеми со голем број на ознаки. Клучната идеја се состои во трансформацијата на повеќезначниот класификациски проблем со голем број на ознаки  $Q$  во повеќе, поедноставни повеќезначни класификациски проблеми со мал број на ознаки  $k \ll Q$  организирани во хиерархиска структура т.е., дрво (Слика 5.4).

Секој јазел  $n$  од ова дрво содржи множество на ознаки  $\mathcal{L}_n \subseteq \mathcal{L}$ . Дрвото содржи вкупно  $|\mathcal{L}| = Q$  листови и секој од нив е претставен со единствена ознака  $\lambda_j$  која припаѓа на оригиналното множество на ознаки  $\lambda_j \in \mathcal{L}$  (*ознаките од интерес*). Множеството на ознаки на секој нетерминален (внатрешен) јазел  $n$  е унија од множествата на ознаки од неговите деца јазли  $\mathcal{L}_n = \cup_{c, c \in \text{children}(n)} \mathcal{L}_c$ . Коренот на дрвото ги содржи сите ознаки  $\mathcal{L}_{\text{root}} = \mathcal{L}$ .

За секој јазел  $n$  се дефинира концептот на *мета ознака*  $\mu_n$  која претставува дисјункција од ознаките содржани во јазелот  $\mu_n \equiv \bigvee \lambda_j, \lambda_j \in \mathcal{L}_n$ . При трансформацијата на податочното множество, секој примерок кој е означен барем со една ознака од  $\mathcal{L}_n$ , истиот ќе биде означен и со *мета ознаката*  $\mu_n$ .



Слика 5.4 Пример хиерархија кај *HOMER* ( $\lambda$  - ознака,  $\mu$  - мета ознака,  $h$  - повеќезначен класификатор)

Секој нетерминален јазел од хиерархијата, исто така содржи и повеќезначен класификациски модел  $h_n$  кој се обучува за предвидување на *мета ознаките* дефинирани во неговите деца. Множеството на ознаки за повеќезначниот класификациски модел  $h_n$  е дефинирано како  $M_n = \{\mu_c | c \in children(n)\}$ . Слика 5.4 прикажува пример хиерархија генерирана за повеќезначен класификациски проблем со 8 ознаки  $\{\lambda_1, \lambda_2, \dots, \lambda_8\}$ .

Класификацијата на нов примерок  $x$  започнува од коренот на дрвото  $\mathcal{L}_{root}$ . Процесот продолжува рекурзивно такашто примерокот  $x$  се класифицира со повеќезначниот класификациски модел  $h_c$  (сместен во дете јазелот  $c$ ), само ако примерокот е означен со ознаката  $\mu_c$  од класификацискиот модел  $h_{parent(c)}$ . Множеството на релевантни ознаки на примерокот  $x$  е составено од ознаките во листовите од дрвото во кои примерокот  $x$  е класификуван.

*HOMER* рекурзивно го гради дрвото со стандардниот пристап за градење на дрва од врвот надолу (анг. *top-down induction*), се разбира започнувајќи од коренот. Во секој јазел  $n$  се креираат  $k$  деца јазли ако  $|\mathcal{L}_n| > k$ , а во спротивно бројот на деца јазлите е еднаков на  $|\mathcal{L}_n|$ . Јазелот  $n$  за обучување на повеќезначниот класификациски модел  $h_n$  ги користи сите примероци од податочното множество кои се означени со најмалку една од ознаките на јазелот  $\mathcal{L}_n$  ( $E_n = \{(\mathbf{x}_i, \mathcal{Y}_i) | (\mathbf{x}_i, \mathcal{Y}_i) \in E_{parent(n)}, \mathcal{Y}_i \cap \mathcal{L}_n \neq \emptyset\}$ ). Коренот на дрвото го користи целото податочно множество  $E_{root} = E$ . Во секој нетерминален јазел се извршуваат последователно две постапки: а) Ознаките се групираат во  $k$  различни групи, една група за секое дете јазел; б) Се креира повеќезначен класификатор за предвидување на мета-ознаките на неговите деца јазли. Извршувањето продолжува се додека во јазлите има повеќе од една единствена ознака.

При градењето на дрвото најголемо значење има алгоритмот за кластерирање т.е. алгоритмот на групирање на ознаките  $\mathcal{L}_n$  во секој од нетерминалните јазли. Имено, од начинот на групирање на ознаките, директно зависат перформансите на предвидување и пресметковна комплексност на *HOMER*. Овој алгоритам всушност претставува алгоритам за хиерархиско кластерирање со чија помош се дефинираат мета-ознаките  $\mu_n$  во нетерминалните јазли од хиерархијата. Алгоритмот се обидува да ја открие хиерархиската поврзаност на податоците ако таа постои и да ја вметне истата во структурата на дрвото со цел да се подобрат перформансите на предвидување и да се намали пресметковната комплексност.

Авторите на овој метод предлагаат три алгоритми за конструирање на хиерархијата [14]: Случајно дефинирана хиерархија, хиерархија дефинирана со алгоритмот *k-средини* и хиерархија дефинирана со алгоритмот *балансиран к-средини*. Најдобри резултати се добиени со употреба на третиот алгоритам (*балансиран к-средини*).

## 5.5 Методи кои го адаптираат алгоритмот

### 5.5.1 *k* - Најблиски соседи за повеќезначна класификација и рангирање

*ML-kNN* (*Multi-label k – Nearest Neighbors*)[43] претставува адаптација на *kNN* алгоритмот за повеќезначна класификација и рангирање. Алгоритмот ги наоѓа *k*-те најблиски соседи (примероци) на тестирачкиот примерок и ги означува како позитивни ако се означени со ознаката  $\lambda$ , односно негативни ако не се означени. Да претпоставиме дека за даден примерок  $\mathbf{x}$  означен со ознаките од множеството  $\mathcal{Y} \subseteq \mathcal{L}$  бројот на соседи определен со овој алгоритам е  $k$ .  $\vec{y}_x$  е неговиот вектор на ознаки, а  $\vec{y}_x^\lambda$  е компонентата која покажува дали примерокот  $\mathbf{x}$  е означен со ознаката  $\lambda$  или не ( $\vec{y}_x^\lambda = 1$  ако примерокот е означен со ознаката  $\lambda$ , а  $\vec{y}_x^\lambda = 0$  ако примерокот не е означен со ознаката  $\lambda$ ). Дополнително, нека  $N(\mathbf{x})$  го претставува множеството од *k* најблиски соседи (примероци) од податочното множество за обучување. За ова множество на примероци може да се дефинира векторот на *бројност на ознаки*:

$$\vec{c}_x(\lambda) = \sum_{a \in N(\mathbf{x})} \vec{y}_a^\lambda, \lambda \in \mathcal{L} \quad (5.5)$$

каде  $\vec{c}_x^\lambda$  е бројот на соседи на  $\mathbf{x}$  кои се означени со ознаката  $\lambda$ .

За секој тестирачки примерок  $\mathbf{t}$ , *ML-kNN* прво ги наоѓа *k*-те најблиски соседи  $N(\mathbf{t})$  од множеството за обучување. Потоа дефинира три настани  $H_1^\lambda$ ,  $H_0^\lambda$  и  $E_j^\lambda$  ( $j \in \{0, 1, \dots, k\}$ ).  $H_1^\lambda$  е настанот примерокот  $\mathbf{t}$  да биде означен со ознаката  $\lambda$ , а  $H_0^\lambda$  е настанот тој да не биде означен со неа.  $E_j^\lambda$  е настанот точно *j* примероци од *k*-те најблиски соседи на примерокот  $\mathbf{t}$  да бидат означени со ознаката  $\lambda$ . Векторот на ознаки  $\vec{y}_t$  на примерокот  $\mathbf{t}$ , *ML-kNN* го определува преку постериорната веројатност:

$$\vec{y}_t^\lambda = \arg \max_{b \in \{0,1\}} P(H_b^\lambda | E_{\vec{c}_t}^\lambda), \lambda \in \mathcal{L} \quad (5.6)$$

Користејќи го Баесовото правило претходната равенка ја добива следната форма:

$$\begin{aligned} \vec{y}_t^\lambda &= \arg \max_{b \in \{0,1\}} \frac{P(H_b^\lambda) P(E_{\vec{c}_t}^\lambda | H_b^\lambda)}{P(E_{\vec{c}_t}^\lambda)} \\ &= \arg \max_{b \in \{0,1\}} P(H_b^\lambda) P(E_{\vec{c}_t}^\lambda | H_b^\lambda) \end{aligned} \quad (5.7)$$

Овој алгоритам дополнително се карактеризира и со способност за рангирање на ознаките преку подредување на постериорните веројатности.

### 5.5.2 C4.5 за повеќезначна класификација и рангирање

Clare и King [59][60] го адаптираат добро познатиот C4.5 алгоритам [61] за класификација и рангирање на повеќезначни податочни множества. Тие преку модификација на формулата за пресметување на ентропијата, дозволуваат во листовите од дрвото за одлучување да се појават повеќе ознаки со кои примерокот може да биде означен.

Пристапот на градење на ML-C4.5 е од врвот надолу. Во секој јазел во дрвото се избира по еден атрибут кој „најдобро“ ги дели примероците од множеството за обучување. Поделбата на примероците се прави врз основа на добивката на информација (анг. *information gain*) која претставува разлика меѓу ентропијата на множеството на примероци во соодветниот јазел и тежинската сума од сите ентропии на множествата на примероци добиени при поделбата со тој атрибут.

$$information\_gain(E) = entropy(E) - \sum_{v \in A} \frac{|E_v|}{|E|} * entropy(E_v) \quad (5.8)$$

кадешто  $A$  е атрибутот по кој е направена поделбата,  $E$  е множеството на примероци во соодветниот јазел, а  $E_v$  е подмножество од  $E$  со вредност  $v$  за атрибутот  $A$ .

Разликата меѓу ML-C4.5 алгоритамот и C4.5 е во начинот на пресметување на ентропијата во секој од јазлите на дрвото. Имено, равенката за пресметување на ентропијата кај C4.5

$$entropy(E) = - \sum_{i=1}^N p(\lambda_i) \log p(\lambda_i) \quad (5.9)$$

се заменува со следната:

$$entropy(E) = - \sum_{i=1}^N p(\lambda_i) \log p(\lambda_i) + q(\lambda_i) \log q(\lambda_i) \quad (5.10)$$

каде што  $p(\lambda_i)$  е веројатноста примерокот да биде означен со ознаката  $\lambda_i$  (нејзината релативна фреквенција), а  $q(\lambda_i) = 1 - p(\lambda_i)$  е веројатноста примерокот да не биде означен со ознаката  $\lambda_i$ .

### 5.5.3 Дрва за предиктивно кластерирање

Во рамката за предиктивно кластерирање, во која се имплементирани *дрвата за предиктивно кластерирање* (анг. *Predictive Clustering Trees – PCTs*), дрвото за одлучување се третира како хиерархија од кластери. На врвот на хиерархијата се наоѓа јазел кој одговара на кластер што ги содржи сите податоци. Движејќи се надолу низ дрвото, секој кластер понатаму рекурзивно се дели на повеќе помали кластери. *PCTs* се имплементирани во CLUS системот кој може да се преземе од следната адреса: <http://clus.sourceforge.net/> [62].

Досега *PCTs* се применети за следните проблеми/задачи: повеќецелна класификација (анг. *multi-target classification*) [56], хиерархиска повеќезначна класификација (анг. *hierarchical multi-label classification*) [63] и предикција на временски серии [64]. Во оваа дисертација, се користат *PCTs* за повеќецелна класификација. Пред да продолжиме со објаснување на *PCTs* за повеќецелна класификација, прво го претставуваме општиот алгоритам за градење *PCTs*.

#### 5.5.3.1 Општ алгоритам за градење на дрва за предиктивно кластерирање

*Дрвата за предиктивно кластерирање* се градат со стандарден алгоритам за градење дрва од врвот надолу (анг. *top-down induction of decision trees – TDIDT*) алгоритам [65]. Алгоритамот е претставен на Слика 5.5. Алгоритамот како влез прифаќа множество од примероци ( $E$ ), а како излез генерира дрво. Хевристиката ( $h$ ) која се користи за избор на тестовите ( $t$ ) е намалување на варијансата предизвикано од делењето на примероците ( $\mathcal{P}$ ) (види линија 4 од процедурата *BestTest* на Слика 5.5). Минимизирањето на варијансата ја максимизира хомогеноста на кластерот со што се подобруваат предиктивните перформанси.

procedure PCT( $E$ ) returns tree	procedure BestTest( $E$ )
1: $(t^*, h^*, \mathcal{P}^*) = \text{BestTest}(E)$	1: $(t^*, h^*, \mathcal{P}^*) = (\text{none}, \theta, \emptyset)$
2: if $t^* \neq \text{none}$ then	2: for each possible test $t$ do
3:     for each $E_k \in \mathcal{P}^*$ do	3: $\mathcal{P} =$ partition induced by $t$ on $E$
4: $tree_k = \text{PCT}(E_k)$	4: $h = \text{Var}(E) - \sum_{E_k \in \mathcal{P}} \frac{ E_k }{ E } \text{Var}(E_k)$
5:     return $\text{node}(t^*, \cup_k \{tree_k\})$	5:     if $(h > h^*) \wedge \text{Acceptable}(t, \mathcal{P})$ then
6: else	6: $(t^*, h^*, \mathcal{P}^*) = (t, h, \mathcal{P})$
7: return $\text{leaf}(\text{Prototype}(E))$	7: return $(t^*, h^*, \mathcal{P}^*)$

Слика 5.5 Општ алгоритам за градење на дрва за предиктивно кластерирање

Главната разлика меѓу алгоритмот за учење на *PCTs* и стандарден алгоритам за учење на дрва за одлучување (како на пример, *C4.5* алгоритмот предложен од Quinlan [61]), е во тоа што во *PCTs*, функцијата за варијанса и функцијата за прототип (која ја пресметува/предвидува ознаката за секој лист) се третираат како параметри кои може да се постават за даден проблем.

### 5.5.3.2 Дрва за предиктивно кластерирање за предвидување на повеќецелни променливи

Дрвата за предиктивно кластерирање кои се способни за симултано повеќецелно предвидување се викаат повеќецелни дрва на одлучување (анг. *multi-target decision trees* – *MTDTs*). *MTDTs* кои прават предикција на торка од континуални променливи (регресиски проблеми) се викаат повеќецелни регресиски дрва (анг. *multi-target regression trees* – *MRTs*), додека *MTDTs* кои предвидуваат торка од дискретни променливи се викаат повеќецелни класификациски дрва (анг. *multi-target classification trees* – *MTCTs*). Делот на CLUS системот за учење на повеќецелни дрва се нарекува *CLUS-MTDT*. Во оваа дисертација користиме *PCTs* за предикција на торка од дискретни променливи.

Функцијата на варијанса за *MTCTs* се пресметува како сума од Цини (анг. *Gini*) индексите на целните променливи,

$$\text{Var}(E) = \sum_{t=1}^q \text{Gini}(E, \mathcal{L}_t) \quad (5.11)$$

Дополнително како варијанса, може да се искористи сумата од ентропиите на целните променливи,

$$\text{Var}(E) = \sum_{t=1}^q \text{Entropy}(E, \mathcal{L}_t) \quad (5.12)$$

(оваа дефиниција веќе претходно се користела во контекст на предвидување на повеќе ознаки [60]).

Прототип функцијата враќа вектор од веројатности за секоја целна променлива за тоа дали даден примерок припаѓа на дадена класа (посебна веројатност за секоја класа на целната променлива). Со користење на овие веројатности може да се одреди најверојатната класа за секоја целна променлива. Како надополнување на претходно споменатите функции за пресметка на варијансата, во CLUS системот се имплементирани други функции за пресметка на варијансата: редуцирана грешка, информациска добивка, однос на добивка и *m*-оценка.

## 5.6 Методи кои градат ансамбли

### 5.6.1 Ансамбли од случајни шуми

*Случајните шуми* (анг. *Random Forests*) [66] претставуваат метод за креирање на ансамбл од класификатори. Пред да го опишеме овој метод ќе го воведеме поимот на *бутстреп* копија (анг. *bootstrap replicate*). Да претпоставиме дека податочното множество  $E$  се состои од  $N$  примероци. *Бутстреп* копија претставува секое множество  $E^i$  кое се добива преку последователно, случајно избирање на  $N$  примероци со замена од оригиналното множество  $E$ . Тоа значи дека еден примерок на множеството  $E$  може да биде избран повеќе пати и исто толку пати да се повторува во рамки на множеството  $E^i$ , додека друг примерок од множеството  $E$  може воопшто да не биде избран и да не учествува во множеството  $E^i$ . За потребите на методот *случајни шуми* се формираат онолку *бутстреп* копии од множеството за обука колку што е и предвидениот број на класификатори во ансамблот. Покрај промена на обучувачкото множество, овој метод воведува и динамичка промена на множеството на обележја. Попрецизно кажано, тоа значи дека примероците од множеството (*бутстреп* копија) во секој јазел во дрвото се опишани со случајно избрани обележја од оригиналното множество на примероци. Бројот на случајно избрани обележја може да биде различен (т.е.  $f(D) = 1, f(D) = \lfloor \sqrt{D} + 1 \rfloor, f(D) = \lfloor \log_2(D) + 1 \rfloor$ ) каде  $D$  е бројот на обележја во оригиналното множество на примероци, а  $f$  е функција која го дава бројот на случајно избрани обележја. Алгоритамот за учење на *случајни шуми* е даден во Слика 5.6.

Излезите на секој од обучените класификатори за невиден примерок од тестирачкото множество се проследуваат на различни комбинациони шеми, преку кои се донесува конечната одлука за ознаките на примерокот.

---

```
procedure RForest( $E, k, f(D)$ ) returns Forest
1:  $F = \emptyset$ 
2: for  $i = 1$  to  $k$  do
3:    $E_i = \text{bootstrap}(E)$ 
4:    $T_i = \text{BaseClassifier}(E_i, f(D))$ 
5:    $F = F \cup T_i$ 
6: return  $F$ 
```

---

**Слика 5.6** Алгоритамот *случајни шуми* за учење на ансамбли.  $E$  е множеството на примероци за тренирање,  $k$  е бројот на дрва, додека  $f(D)$  е големината на просторот на обележја кој се користи при учење на моделот

Во споредбата на методи за повеќезначна класификација и рангирање, ансамблот од *случајни шуми* беше вклучен со две различни имплементации:

1. *Случајни шуми* од C4.5 за повеќезначна класификација и рангирање (RFML-C4.5).
2. *Случајни шуми* од дрва за предиктивно кластерирање (RF-PCT).

Имплементацијата која го користи методот C4.5 за повеќезначна класификација и рангирање, предикциите на ознаките ги добива со користење на алгоритамот за мнозинско гласање. Во имплементацијата со дрва за предиктивно кластерирање

предикциите се добиваат со користење на гласање со дистрибуцијата на веројатности на начинот предложен од Waeg и Kohavi [67], за секоја ознака посебно.

## 5.6.2 Случајни $k$ - множества на ознаки

Случајни  $k$  – множества на ознаки (анг. *RAndom k-labEL sets - RAKEL*) е метод за повеќезначна класификација и рангирање предложен од Tsoumakas et al. [23]. Пред да бидат опишани алгоритмите за обучување и тестирање на овој метод, ќе биде воведен терминот  $k$  - множество на ознаки.

Да претпоставиме дека  $\mathcal{L} = \{\lambda_1, \dots, \lambda_Q\}$  е множеството на ознаки во повеќезначен класификациски проблем. Множеството  $Y \subseteq \mathcal{L}$ , со  $k = |Y|$  се нарекува *множество на ознаки со големина  $k$*  (анг. *k - labelset*). Со  $\mathcal{L}^k$  се означени сите различни *множества на ознаки со големина  $k$*  во  $\mathcal{L}$ . Вкупниот број на овие множества е определен со биномниот коефициент  $|\mathcal{L}^k| = \binom{Q}{k}$ .

*RAkEL* итеративно конструира ансамбл од  $m$  модели од типот *множества на ознаки*. Во секоја итерација  $i = 1, \dots, m$ , случајно избира *множество на ознаки со големина  $k$*  ( $Y_i$  од  $\mathcal{L}^k$ ) без замена, за кое обучува по еден *LP* модел. Процесот на обучување е прикажан на Слика 5.7:

---

```
procedure Training_RAKEL( $m, k, \mathcal{L}, E$ )
1:  $R \leftarrow \mathcal{L}^k$ 
2: for  $i = 1$  to  $\min(m, |\mathcal{L}^k|)$  do
3:    $Y_i \leftarrow k\text{LabelsetRandom}(R)$ 
4:    $\text{trainingLabelPowersetModel}(E, Y_i)$ 
5:    $R \leftarrow R \setminus Y_i$ 
```

---

Слика 5.7 Процедура за обучување на *RAkEL*.  $m$  е бројот на модели,  $k$  големината на *множеството на ознаки*,  $\mathcal{L}$  множеството на ознаки и  $E$  множеството на примероци за обучување

Бројот на итерации (членови на ансамблот)  $m$  е кориснички дефиниран параметар и може да прими вредности во опсегот од 1 до  $|\mathcal{L}^k|$ . Големината на множеството на ознаки  $k$  е исто така параметар кој го специфицира корисникот и може да прими вредности во опсегот од 2 до  $Q - 1$ . За  $k = 1$  и  $m = Q$  се добива методот *бинарна релевантност*.

Во процесот на тестирање секој модел од ансамблот носи одлука дали конкретен примерок  $\mathbf{x}$  е означен или не е означен со ознаката  $\lambda$  (1 – примерокот е означен, 0 – примерокот не е означен). *RAkEL* ја пресметува просечната вредност за секоја ознака  $\lambda$  од оргиналното множество на ознаки  $\mathcal{L}$  и носи позитивна одлука ако пресметаната просечна вредност е поголема од кориснички дефинирана прагова вредност. Процесот на тестирање е претставен на Слика 5.8.

---

```
procedure Testing_RAKEL( $m, \mathcal{L}, E$ )
1: for  $i = 1$  to  $Q$  do
2:    $\text{Sum}_i \leftarrow 0$ 
```

---

```
3:    $Votes_i \leftarrow 0$ 
4:   for  $i = 1$  to  $m$  do
5:     for each  $\lambda_j \in Y_i$  do
6:        $Sum_j \leftarrow Sum_j + M_i(\mathbf{x}, \lambda_j)$ 
7:        $Votes_j \leftarrow Votes_j + 1$ 
8:   for  $i = 1$  to  $Q$  do
9:      $Avg_i \leftarrow Sum_i / Votes_i$ 
10:    if  $Avg_i > t$  then
11:       $Result_i \leftarrow 1$ 
12:    else
13:       $Result_i \leftarrow 0$ 
```

Слика 5.8 Процедура за тестирање на *RAkEL*.  $m$  е бројот на модели,  $k$  големината на множеството на ознаки,  $\mathcal{L}$  множеството на ознаки,  $Y_i$  множество на ознаки со големина  $k$  и  $M_i$  модели од *LP* ансамблот

### 5.6.3 Ансамбли од верижни архитектури (ECC)

Методот за градење ансамбли од верижни архитектури (анг. *ensembles of classifier chains - ECC*) е предложен од Read et al. [12]. Како што кажува и неговото име, ансамбл од овој тип комбинира предикции добиени од различни класификатори базирани на верижната архитектура (CC) претставена во поглавјето 5.2.2. Имено, ECC обучува  $m$  CC класификатори  $C_1, C_2, \dots, C_m$ . Секој од овие класификатори  $C_k$  се обучува со:

- Случајно подмножество од оригиналното множество на примероци  $E$ .
- Случаен редослед на бинарните модели во верижната архитектура.

Во процесот на тестирање, предикциите од сите класификатори се сумираат по ознака, т.е., секоја ознака добива одреден број на гласови. За одредување на релевантните ознаки на примероците се користи прагова вредност.

За даден примерок, прво, секој класификатор  $C_k$  прави предикција на векторот на ознаки  $\vec{y}_k = (\lambda_1, \dots, \lambda_Q) \in \{0,1\}^Q$ . Сите предикции се сумираат по ознака и се добива векторот  $W = (l_1, \dots, l_Q) \in \mathbb{R}^Q$ . Секое  $l_i \in W$  ја претставува сумата на гласови за ознаката  $\lambda_i \in \mathcal{L}$ . Потоа, векторот на гласови  $W$  се нормализира ( $W^{norm}$  - дистрибуција на ознаките во опсегот  $[0,1]$ ) и ознаките се подредуваат според нормализираните вредности. На крајот се воведува прагова вредност за да се одредат финалните предикции (релевантните и нерелевантните ознаки) за секој примерок.

## 5.7 Експерименти

Во ова поглавје е прикажан експерименталниот дизајн (основни класификатори, поставување на параметри и статистичка евалуација) како и добиените резултати споредени и анализирани во однос на споредените методи за повеќезначна класификација и рангирање.

Целта на овој дел од истражувањето е да одговориме на следниве прашања:

- Кои се предностите и недостатоците на споредуваните методи за повеќезначна класификација и рангирање?

- Кои методи се поефикасни, а кои имаат подобри перформанси на предвидување?
- Кои методи се најсоодветни за различните типови на класификациски проблеми?
- Како влијаат карактеристиките на податочните множества (број на примероци, број на обележја, кардиналност и густина на ознаки) на перформансите на методите за повеќезначна класификација и рангирање?

За да одговориме на овие прашања направена е споредба на перформансите на 12 најмногу цитирани и најзначајни (анг. *state of the art*) методи за повеќезначна класификација и рангирање (истите се претставени во ова поглавје) на 11 бенчмарк повеќезначни класификациски проблеми претставени во претходната глава. За евалуација на перформансите на методите во однос на прецизноста на предвидување и ефикасноста, користени се 18 евалуациски мерки поделени во четири групи: евалуациски мерки базирани на примерок, евалуациски мерки базирани на ознака, евалуациски мерки базирани на рангирање и времињата на обучување и тестирање. Дополнително, за да ја процениме статистичката значајност на разликата на перформансите на споредуваните методи користен е *коригиран Friedman тест* [72] и *post-hoc Nemenyi тест* [73] како што е предложено во истражувањето на Demšar [74].

### 5.7.1 Експериментален дизајн

Експерименталниот дизајн опфаќа оптимизирање и поставување на параметрите на методите, избор на основни класификатори кај методите, како и изборот на тестови за статистичка евалуација на резултатите за определување на статистичката значајност на добиените перформанси.

При споредбата на методите за повеќезначна класификација и рангирање користени се повеќе различни системи за машинско учење: MULAN<sup>1</sup> [68] библиотека за повеќезначна и хиерархиска повеќезначна класификација и рангирање која како основна платформа ја користи библиотеката WEKA<sup>2</sup> [69], MEKA<sup>3</sup> екстензијата на WEKA и рамката за предиктивно кластерирање CLUS<sup>4</sup>. MULAN библиотеката се користи за методите: *BR*, *CLR*, *QWML*, *HOMER*, *ML-C4.5*, *RFML-C4.5*, *ML-kNN* и *RAkEL*; MEKA околината е користена за верижните архитектури *CC* и *ECC*, додека CLUS за *PCT* и *RF-PCT*. Сите експерименти беа направени на сервер со *Интел Ксеон* (анг. *Intel Xeon*) процесор на 2.50GHz и 64GB RAM меморија со *Федора 14* (анг. *Fedora 14*) оперативен систем. Во продолжение на ова поглавје прво се претставени основните класификатори кои се користат од методите за повеќезначна класификација и рангирање кои го трансформираат проблемот, а потоа е опишан начинот на поставување на параметрите на споредуваните методи.

---

<sup>1</sup> <http://mulan.sourceforge.net/>

<sup>2</sup> <http://www.cs.waikato.ac.nz/ml/weka/>

<sup>3</sup> <http://meqa.sourceforge.net/>

<sup>4</sup> <http://clus.sourceforge.net/>

### 5.7.1.1 Основни класификатори

Сите методи кои го трансформираат проблемот и методите за градење ансамбли вклучени во оваа експериментална евалуација користат два вида на основни класификатори: машини со носечки вектори и дрва за одлучување. За имплементацијата на машините со носечки вектори е користена LIBSVM<sup>1</sup> [70] библиотеката. Конкретно, користени се машини со носечки вектори со Гаусов кернел од страна на сите методи кои го трансформираат проблемот и ансамбл методите *RAkEL* и *ECC*. Кернел параметарот *гама* (анг. *gamma*) и казната *C* се определени преку 10-кратна вкрстена валидација на сите примероци од податочното множество за обучување. Исклучок од ова е само ансамбл методот *RAkEL* чии параметри се определени со 5-кратна вкрстена валидација за *tmc2007* и *mediamill* податочните множества поради неговата голема пресметковна комплексност за овие две податочни множества. Вредностите  $2^{-15}$ ,  $2^{-13}$ , ...,  $2^1$ ,  $2^3$  се користени за определување на вредноста на параметарот *гама*, додека вредноста на *C* е определена од вредностите  $2^{-5}$ ,  $2^{-3}$ , ...,  $2^{13}$ ,  $2^{15}$ . По изборот на оптималните вредности на параметрите посебно за секој метод во однос на секое податочно множество, методите се обучени со примероците од соодветното податочно множество за обучување и евалуирани со примероците од податочно множество за тестирање.

Во докторската дисертација се користени две имплементации на дрвата за одлучување: *ML-C4.5* од MULAN и *PCT* од CLUS. И двата класификатори користат методи за поткастрување. *ML-C4.5* го поткаструва дрвото по неговата изградба користејќи го факторот на доверба, додека *PCT* го поткаструва дрвото во текот на неговата изградбата користејќи *F*-тест (дали поделбата на примероците во даден јазел статистички значајно влијае на намалување на варијансата). Од друга страна, кога овие класификатори се користат како основни класификатори во ансамблите *RF-MLC4.5* и *RF-PCT* истите не се поткаструваат [67].

### 5.7.1.2 Поставување на параметри

Параметрите на сите методи во експерименталната евалуација беа поставени во согласност со препораките од литературата. Конкретно, за методите ансамбли базирани на дрвата за одлучување (*RF-C4.5* и *RF-PCT*), бројот на повеќезначни модели во ансамблите е 100 (број на модели препорачан од Bauer и Kohavi [67]). Притоа *RF-C4.5* користи  $f(D) = \lfloor \log_2(D) + 1 \rfloor$  обележја [66] во секој од јазлите при конструирање на основните модели, а *RF-PCT* користи  $f(D) = \lfloor 0.1 \cdot D + 1 \rfloor$  обележја како што е предложено во докторската дисертација на Kosev [57] ( $D$  е вкупниот број на обележја во податочното множество). Бројот на модели кај *ECC* ансамблот е поставен на 10, по препорака од авторите на методот [12]. *RAkEL* методот користи  $\min(2 \cdot Q, 100)$  (каде  $Q$  е вкупниот број на ознаки) модели за сите податочни множества, освен за податочните множества *mediamill*, *delicious* и *bookmarks* каде овој параметар е поставен на 10 како резултат на мемориските побарувања на овој метод. Покрај овој параметар кој го дефинира бројот на основни модели во ансамблот, постои уште еден параметар кај *RAkEL* методот кој треба да биде поставен: големината на *множество на ознаки* -  $k$ . За сите податочни множества овој параметар има вредност еднаква на половина од бројот на ознаки ( $Q/2$ ). Tsoumakas et al. [18] и Read et al. [12] имаат покажано дека оваа вредност претставува добар избор за големината на *множеството на ознаки*, поради

<sup>1</sup> <http://www.csie.ntu.edu.tw/~cjlin/libsvm/>

тоа што поставува баланс меѓу перформансите на предвидување и пресметковната комплексност.

*ML-C4.5* методот користи *издигнување на поддрво* (анг. *subtree raising*) [71] како метод за поткастрување по изградбата на дрвото со *фактор на доверба* од 0.25. Параметарот за минимален број на примероци кај *ML-C4.5* е поставен на 10. *PCT* користи поткастрување на дрвото во текот на неговата изградбата со *F*-тест за да одреди дали поделбата на примероците во даден јазел статистички значајно влијае на намалување на варијансата. Нивото на значајност на *F*-тест автоматски се одредува од предефинирана листа со 3-кратна вкрстена валидација. Бројот на соседи кај *ML-kNN* е одреден од вредностите 6 до 20 со чекор 2. *HOMER* исто како и *ML-kNN* има еден дополнителен параметар за поставување: бројот на кластери во хиерархијата. За овој параметар пробавме 5 различни вредности (2, 3, 4, 5 и 6). За сите методи се прикажани најдобрите резултати.

### 5.7.1.3 Статистичка евалуација

За да ја процениме статистичката значајност на разликата на перформансите на споредуваните методи користен е *коригиран Фридман* (анг. *Friedman*) *тест* [72] и *дополнителен Немењи* (анг. *post-hoc Nemenyi*) *тест* [73] како што е предложено во истражувањето на Demšar [74]. *Фридман* е непараметарски тест за тестирање на повеќе хипотези. Тој прво ги подредува сите методи во однос на секое податочно множество според нивните перформанси. Така најдобриот метод добива ранг 1, вториот најдобар ранг 2 итн. (ако перформансите на повеќе методи се еднакви тогаш тој доделува просечен ранг). Потоа *Фридман тестот* ги споредува просечните рангови на методите и пресметува  $\chi^2_F$  – *Фридман*-ова статистика која е дистрибуирана со  $k - 1$  степени на слобода ( $k$  е бројот на методи). Iman и Davenport [75] покажале дека *Фридман*-новата статистика е премногу конзервативна и предлагаат коригирана *Фридман (F)* статистика која е дистрибуирана со  $k - 1$  и  $(k - 1) \cdot (N - 1)$  степени на слобода ( $N$  е бројот на податочни множества).

Доколку е детектирана статистички значајна разлика во перформансите на методите, се користи *дополнителен* (анг. *post-hoc*) тест за да се открие меѓу кои од методите постои таа разлика. Како *дополнителен* тест во статистичката евалуација на добиените резултати е користен *Немењи* тестот. Кај овој тест, перформансите на две методи статистички значајно се разликуваат доколку растојанието меѓу просечните рангови на тие две методи е поголемо од некое критично растојание. Критичното растојание зависи од бројот на методи, бројот на податочни множества и критичната вредност (за дадено ниво на значајност -  $p$ ) која е базирана на Студентовата распределба и може да се најде во литературата за статистика (на пр. [76]).

Во продолжение се претставени резултатите од *дополнителниот Немењи* тест преку дијаграми на просечните рангови [74]. Истите се дадени во Слика 5.9 до Слика 5.16. Критичните дијаграми содржат нумерирана оска на која се означени просечните рангови на методите. Најдобрите методи се нанесени на десната страна на дијаграмите. Ако разликата на просечните рангови на методите е статистички незначајна (на ниво на значајност  $p = 0.05$ ), линиите кои ги означуваат ранговите се поврзани со хоризонтална линија.

Некои методи не генерираа модел за предвидување за големите податочни множества за време од една недела на ресурсите кои ги имавме на располагање. Во

табелите со резултати наместо постигнатите перформанси за овие методи е запишано **DNF (Did Not Finish)**. Имајќи го ова во предвид, статистичката анализа ја направивме два пати. Првиот пат во анализата се вклучени само оние податочни множества (8 податочни множества) за кои сите методи генерирале модел и дале резултати (Слика 5.9, Слика 5.11, Слика 5.13 и Слика 5.15). Во втората анализа (Слика 5.10, Слика 5.12, Слика 5.14 и Слика 5.16) вклучени се сите податочни множества, а оние методи кои не генерирале модел за конкретното податочно множество се казнети со доделување на најслабиот ранг во однос на сите евалуациски мерки.

## 5.7.2 Резултати и дискусија

Во ова поглавје се прикажани резултатите добиени од експерименталната евалуација. За сите типови на евалуациски мерки ги прикажуваме и анализираме критичните дијаграми добиени од статистичката евалуација на добиените резултати. Критичните дијаграми се групирани во четири групи според типот на евалуациските мерки кои тие ги прикажуваат. Дополнително во Додаток А се прикажани комплетните резултати за секоја евалуациска мерка посебно.

### 5.7.2.1 Резултати добиени од евалуациските мерки базирани на примерок

Како што споменавме претходно, евалуациските мерки базирани на примерок ги вклучуваат *Хаминг загуба*, *точност*, *прецизност*, *odsub*, *класификациска точност* и  $F_1$ . На Слика 5.9 и Слика 5.10 се прикажани критичните дијаграми за секоја евалуациска мерка посебно, додека комплетните резултати се прикажани во Табела А.1 до Табела А.6. Дополнително, во Табела 5.1 се прикажани  $p$ -вредностите за проценка на статистичката значајност на разликите во перформансите на методите со *Фридман* тестот.

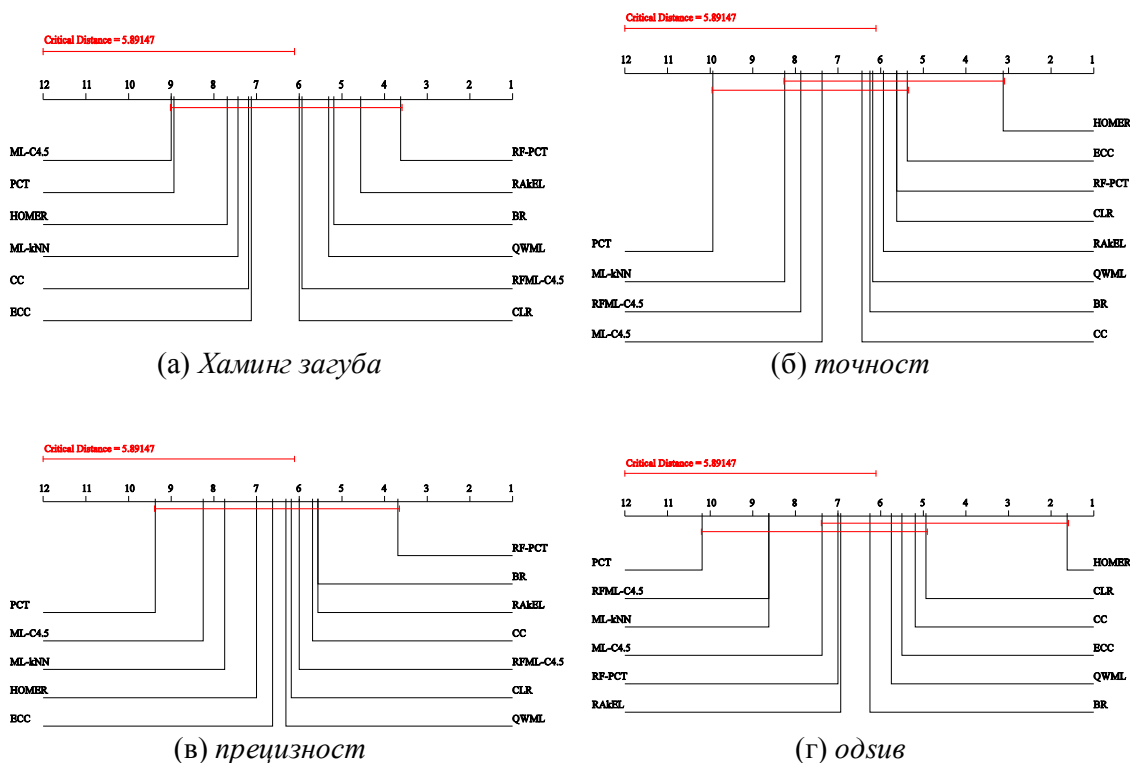
Врз основа на резултатите добиени на податочните множества за кои сите методи дале резултати (Слика 5.9) може да се направат неколку главни заклучоци. Првиот заклучок кој го привлече нашето внимание е фактот дека методот *HOMER* покажа најдобри резултати во однос на *odsub* евалуациската мерка, додека *RF-PCT* ансамблот покажа најдобри перформанси во однос на *прецизност* евалуациската мерка. Ова значи дека предикциите направени од *HOMER* се покомплетни: релевантните ознаки се предвидени како релевантни (подобрите вредности на *odsub* се резултат на малиот број на релевантни ознаки кои се погрешно предвидени). Од друга страна, помалата прецизност значи дека поголем број на ознаки кои оригинално се нерелевантни за примерокот се предвидени како релевантни. Поради тоа може да се заклучи дека *RF-PCT* е погзактен (ознаките предвидени како релевантни се навистина релевантни за примерокот). За разлика од *HOMER*, *RF-PCT* остава поголем број на релевантни ознаки со погрешна предикција.

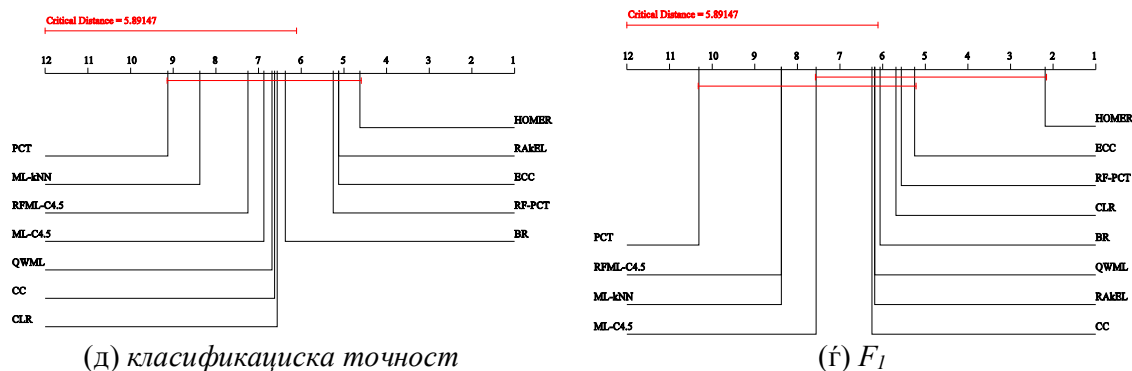
Во продолжение се анализирани перформансите на сите методи во однос на шесте евалуациски мерки. Од добиените резултати и статистичката евалуација генерално може да се заклучи дека најдобри перформанси имаат *RF-PCT* и *HOMER*, а веднаш позади нив се методите *BR* и *CC*. *RF-PCT* има најдобри перформанси во однос на *Хаминг загуба* евалуациската мерка и *прецизност* и е трето рангиран во однос на евалуациската мерка *точност* и  $F_1$ . *HOMER* е најдобар во однос на евалуациските мерки *класификациска точност*, *точност*, *odsub* и  $F_1$ . *HOMER* покажа особено лоши резултати на *Хаминг загуба* и *прецизност* (10-та и 9-та позиција на критичните дијаграми, соодветно), додека *RF-PCT* е најдобар на овие две мерки. Претпоставуваме

дека лошите перформанси на *HOMER* во однос на *Хаминг загуба* се резултат на начинот на градење на дрвото и дефинирање на хиерархијата.

Во однос на статистичката евалуација, критичните дијаграми покажуваат дека *HOMER* и *RF-PCT* се статистички значајно подобри од дрвата за одлучување *ML-C4.5* и *PCT* (на ниво на значајност од  $p = 0.05$ ). Од гледна точка на методите кои градат ансамбли, перформансите на *RF-PCT* се значително подобри од основниот модел *PCT* (целното обележје е множество на ознаки), слично како и за проблемот на регресија и повеќекласна класификација. Од друга страна подобрувањето на перформансите кај *ECC* и *RF-MLC4.5* не е толку очигледно. За некои евалуациски мерки перформансите на нивните основни модели ги надминуваат перформансите на ансамблите. За ова однесување на методите кои градат ансамбли и нивните основни модели имаме две претпоставки: прво, *CC* е стабилен класификатор и креирањето на ансамблот неможе многу да ги подобри перформансите на предвидување. Второ, резултатите на *RF-MLC4.5* се полоши во споредба со *RF-PCT*, затоа што големината на множеството на обележја кои се користат кај овој метод е логаритам од вкупниот број на обележја на класификацискиот проблем. Имајќи предвид дека областите што се користат во ова истражување (и други области од повеќезначната класификација) имаат голем број на обележја (обично повеќе од 500), логаритамската функција не обезбедува доволен број на обележја, односно недостасуваат некои корисни информации што можат да ги подобрат перформансите на класификацијата и рангирањето.

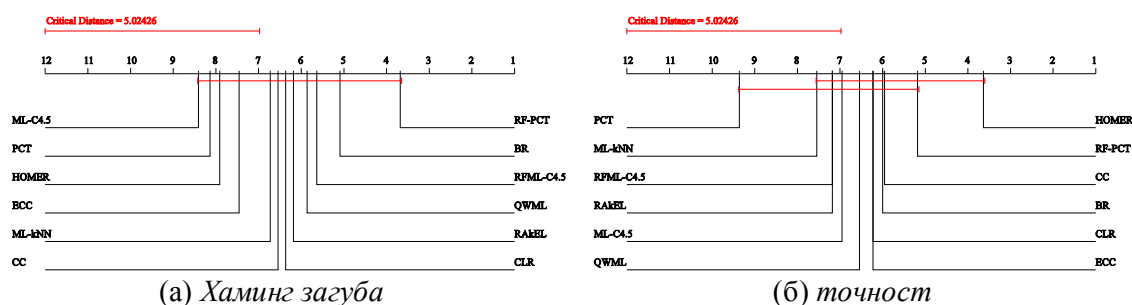
Следно, во рамките на оваа дискусија ќе се фокусираме на перформансите на различните типови на алгоритми за машинско учење. Резултатите покажуваат дека *ML-kNN* како претставник на методите за *мрзливо учење* (анг. *lazy learning*) има лоши перформанси во однос на сите евалуациски мерки базирани на примерок.

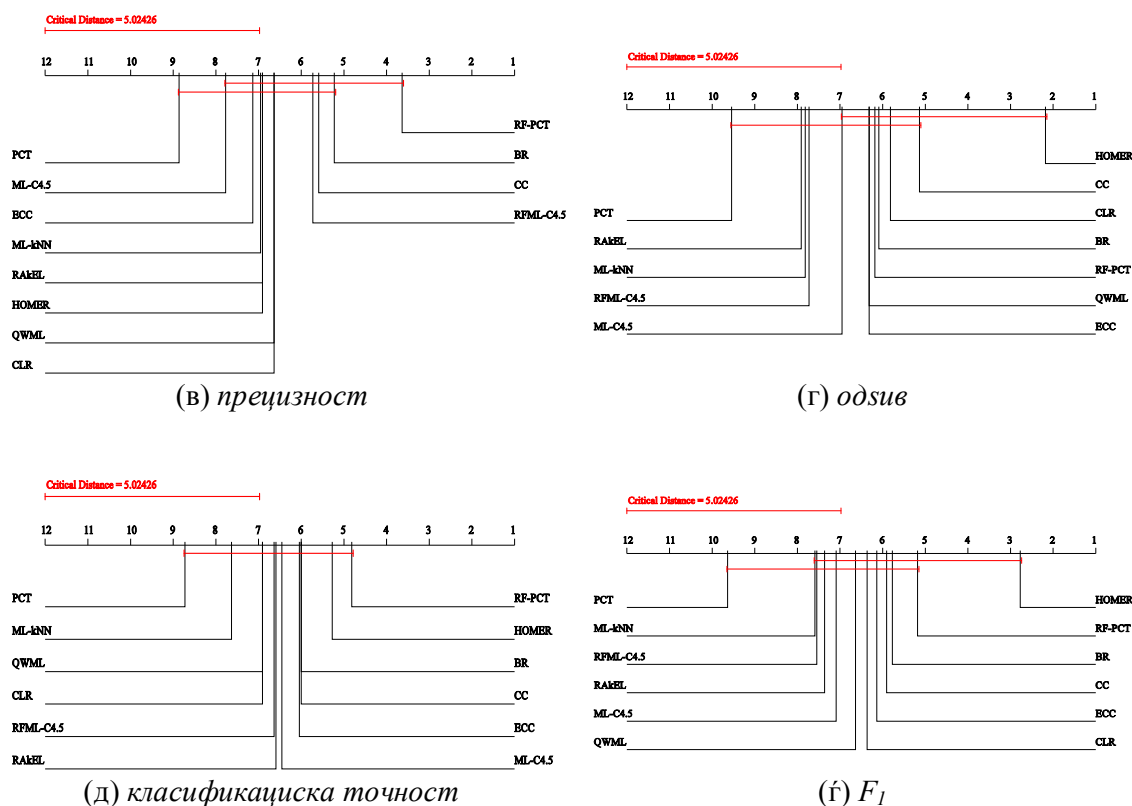




Слика 5.9 Критични дијаграми за мерките базирани на примерок. Резултати добиени од дополнителниот Немењи тест со ниво на значајност од 0.5 на податочните множества за кои сите методи дале резултати

*SVM*-базираниите методи покажуваат подобри перформанси на малите податочни множества (првата половина на податочни множества), додека методите базирани на дрвата за одлучување имаат подобри перформанси за големите податочни множества. Ова е последица на Гаусовиот кернел што го користат *SVM*-базираниите методи. Имено, тој добро се снаоѓа со податочните множества со мал број на примероци по ознака, за разлика од податочните множества кои имаат голем број на примероци, за кои неговите перформанси се приближуваат до перформансите на линеарен кернел. Исто така, *SVM*-базираниите методи покажуваат подобри перформанси за области кои имаат поголем број на обележја. На пример во проблемите на класификација на текст кои најчесто се карактеризираат со голем број на обележја, секој документ е претставен со комбинација од зборови како обележја и секое од тие обележја може да игра значајна улога во донесувањето на точната предикција. За разлика од нив, дрвата за одлучување користат само мало подмножество од сите обележја.





**Слика 5.10** Критични дијаграми за мерките базирани на примерок. Резултати добиени од дополнителниот Немењи тестот со ниво на значајност од 0.5 на сите податочните множества

**Табела 5.1**  $p$ -вредностите за проценка на перформансите на методите за повеќезначна класификација и рангирање со Фридман тестот во однос на евалуациските мерки базирани на примерок. Ознаката **Сите** се однесува на  $p$ -вредностите пресметани за резултатите на податочните множества за кои сите методи дале резултати. Ознаката **Подмножество** се однесува на  $p$ -вредностите пресметани за резултатите на сите податочни множества

	Сите	Подмножество
Хаминг загуба	0.089	0.047
Точност	0.077	0.037
Прецизност	0.117	0.19
Одсив	$5 \cdot 10^{-4}$	$7.9 \cdot 10^{-5}$
Класификациска точност	0.555	0.343
$F_1$	$5 \cdot 10^{-3}$	0.0013

За разлика од резултатите на податочните множества за кои сите методи генерирале модели за предикција, резултатите од сите податочни множества уште повеќе ја истакнуваат предноста на *RF-PCT* и *HOMER* во однос на останатите методи, како резултат на казните за необучените модели. Свкупно, *RF-PCT* има најдобри

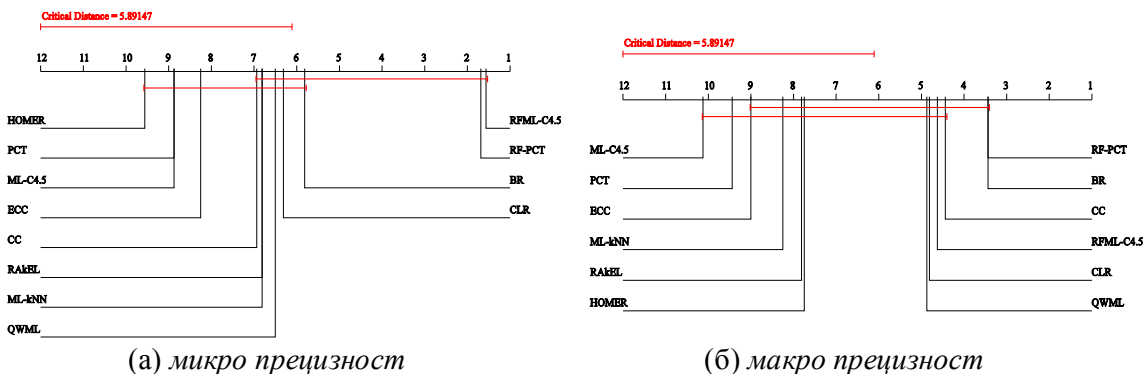
перформанси (највисок просечен ранг за сите мерки), а по него следат *HOMER*, *BR* и *CC*. *RF-PCT* е најдобар во однос на класификациска точност, Хаминг загуба и прецизност, додека *HOMER* е најдобар во однос на евалуациските мерки *точност*, *odsv* и  $F_1$ . *RF-PCT* е второ рангиран според *точност* и  $F_1$ , и има подобрени перформанси во однос на евалуациската мерка *odsv*.

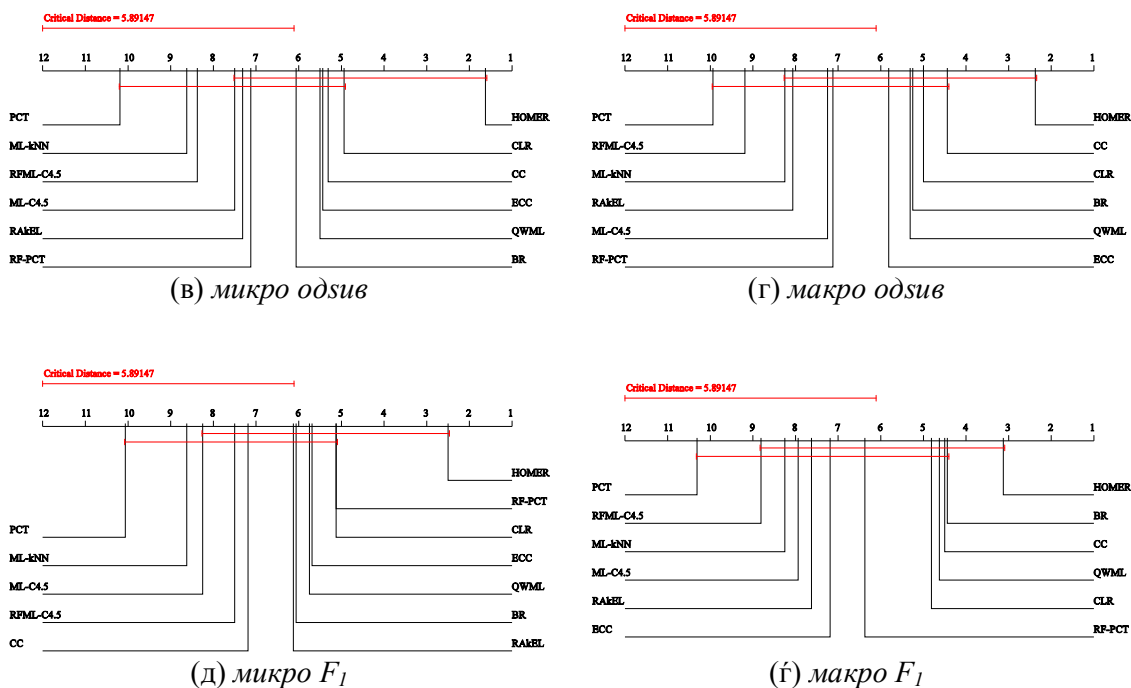
### 5.7.2.2 Резултати добиени од евалуациските мерки базирани на ознака

Евалуациските мерки базирани на ознака ги вклучуваат *микро прецизност*, *макро прецизност*, *микро odsv*, *макро odsv*, *микро  $F_1$*  и *макро  $F_1$* . На Слика 5.11 и Слика 5.12 се прикажани критичните дијаграми за секоја евалуациска мерка посебно, додека комплетните резултати се прикажани во Табела А.7 до Табела А.12. Дополнително, во Табела 5.2 се прикажани *p*-вредностите за проценка на статистичката значајност на разликите во перформансите на методите со Фридман тестот. Според критичните дијаграми на податочните множества за кои сите методи генерирале модел за предикција, најдобри перформанси покажаа *HOMER*, *RF-PCT*, *BR* и *CC*. *HOMER* има најдобри резултати за четири евалуациски мерки *микро odsv*, *макро odsv*, *микро  $F_1$*  и *макро  $F_1$* . *RF-PCT* покажа најдобри резултати на *макро прецизност* и е статистички значајно подобар од *HOMER* на *микро прецизност*, евалуациската мерка за која *RF-MLC4.5* покажа просечно најдобри резултати.

Слично како и кај евалуациските мерки базирани на примерок, така и кај мерките базирани на ознака очигледна е предноста на *RF-PCT* методот во однос на неговиот основен класификатор (*PCT*). Предноста на останатите два ансамбли (*ECC* и *RF-MLC4.5*) и кај овие мерки не е изразена во споредба со нивните основни класификатори. Перформансите на *CC* во просек дури и ги надминуваат перформансите на соодветниот ансамбл. Причините за лошите перформанси на овие два методи кои градат ансамбли се исти како и оние кај евалуациските мерки базирани на примерок.

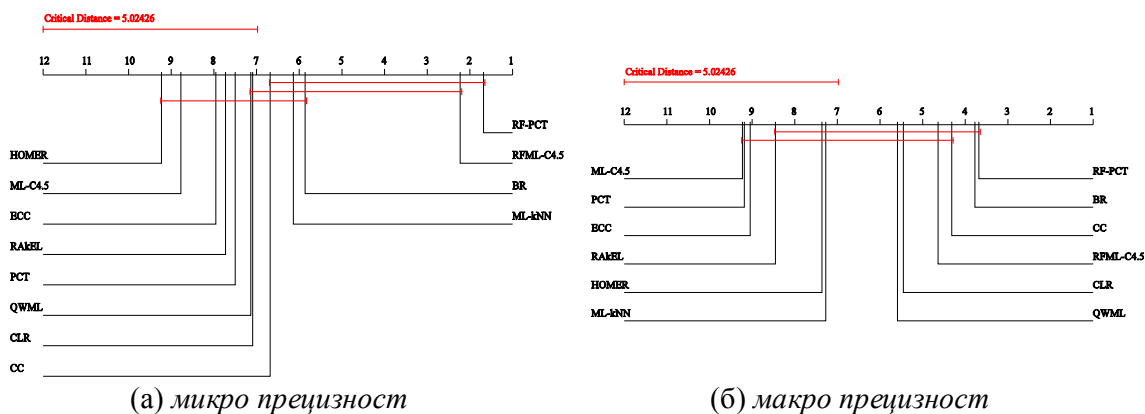
Во однос на перформансите на различните типови на методи за машинско учење, методите кои користат машини со носечки вектори повторно доминираат за помалите податочни множества и класификациските проблеми опишани со поголем број на обележја, додека дрвата за одлучување се подобри на поголемите податочни множества. Методот базиран на *k*-најблиски соседи покажа послаби резултати во однос на другите методи.

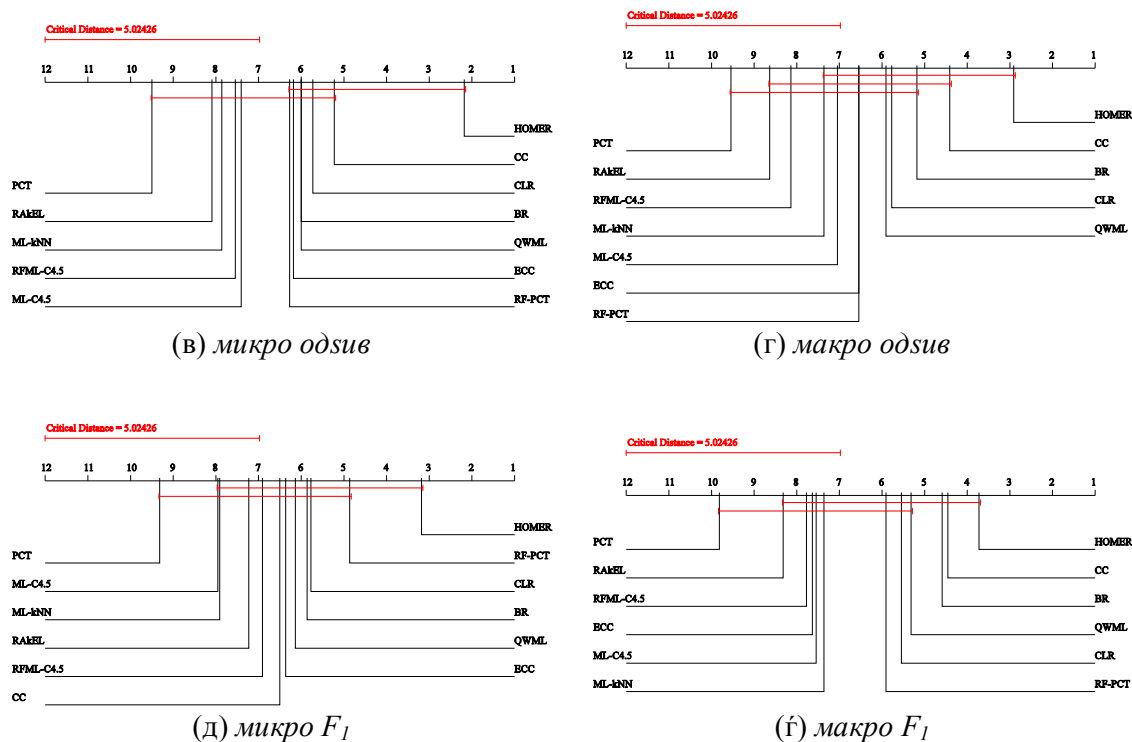




**Слика 5.11** Критични дијаграми за мерките базирани на ознака. Резултати добиени од дополнителниот Немењи тест со ниво на значајност од 0.5 на податочните множества за кои сите методи дале резултати

Анализата на статистичката евалуација покажа дека додавањето на податочните множества за кои сите методи не генерирале модел за предикција не ја променило состојбата премногу. Имено, предноста на методите со помала пресметковна комплексност (*RF-PCT*, *HOMER*, *CC* и *BR*) е уште поизразена. Заклучоците за релативните просечни перформанси остануваат многу слични како и за перформансите постигнати на подмножествата од податочни множества: *HOMER* е најдобар според *микро одсив*, *макро одсив*, *микро  $F_1$*  и *макро  $F_1$* , а *RF-PCT* според евалуациските мерки *макро прецизност* и *микро прецизност*.





Слика 5.12 Критични дијаграми за мерките базирани на ознака. Резултати добиени од дополнителниот Немеџи тест со ниво на значајност од 0.5 на сите податочните множества

Табела 5.2  $p$ -вредностите за проценка на перформансите на методите за повеќезначна класификација и рангирање со Фридман тестот во однос на евалуациските мерки базирани на ознака. Ознаката **Сите** се однесува на  $p$ -вредностите пресметани за резултатите на податочните множества за кои сите методи дале резултати. Ознаката **Подмножество** се однесува на  $p$ -вредностите пресметани за резултатите на сите податочни множества

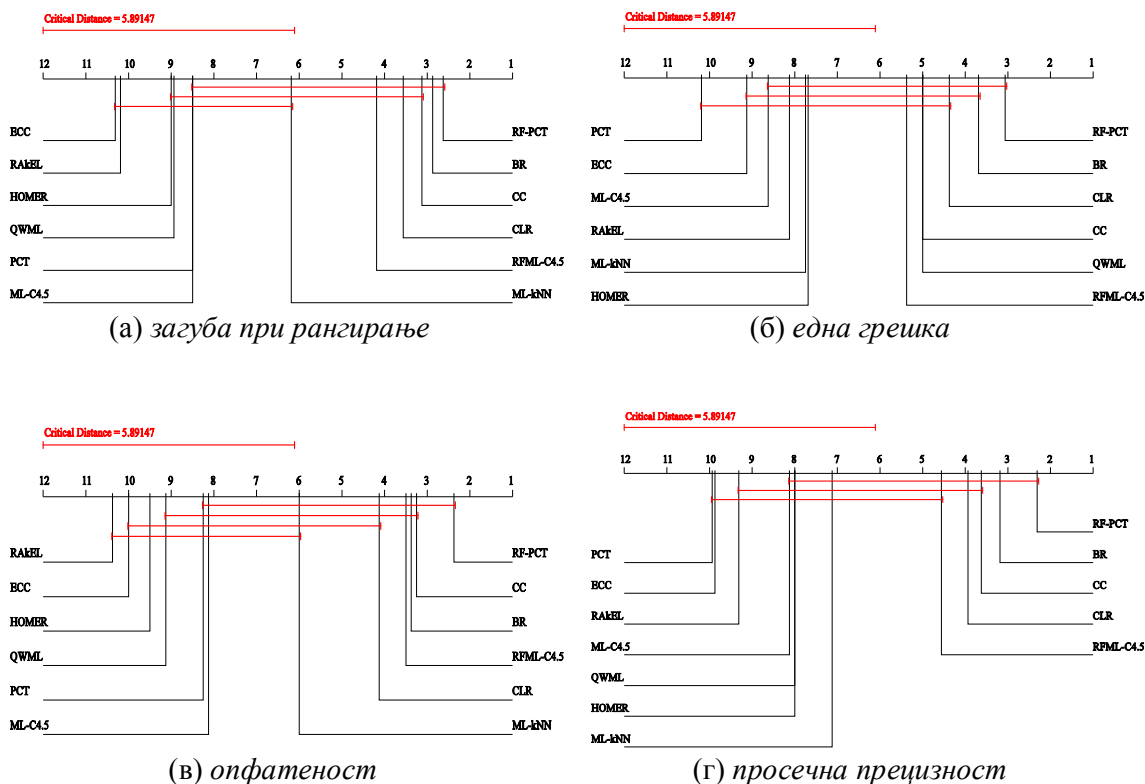
	Сите	Подмножество
Макро прецизност	$3.5 \cdot 10^{-7}$	$4.8 \cdot 10^{-7}$
Макро одзив	$2.8 \cdot 10^{-4}$	$1.1 \cdot 10^{-4}$
Макро $F_1$	$3.1 \cdot 10^{-4}$	$9.8 \cdot 10^{-5}$
Микро прецизност	$3.7 \cdot 10^{-9}$	$3.4 \cdot 10^{-8}$
Микро одзив	$3.6 \cdot 10^{-4}$	$7.3 \cdot 10^{-5}$
Микро $F_1$	0.011	0.002

### 5.7.2.3 Резултати добиени од евалуациските мерки базирани на рангирање

Евалуациските мерки базирани на рангирање ги вклучуваат мерките загуба при рангирање, една грешка, опфатеност и просечна прецизност. На Слика 5.13 и Слика 5.14 се прикажани критичните дијаграми за секоја евалуациска мерка посебно, додека

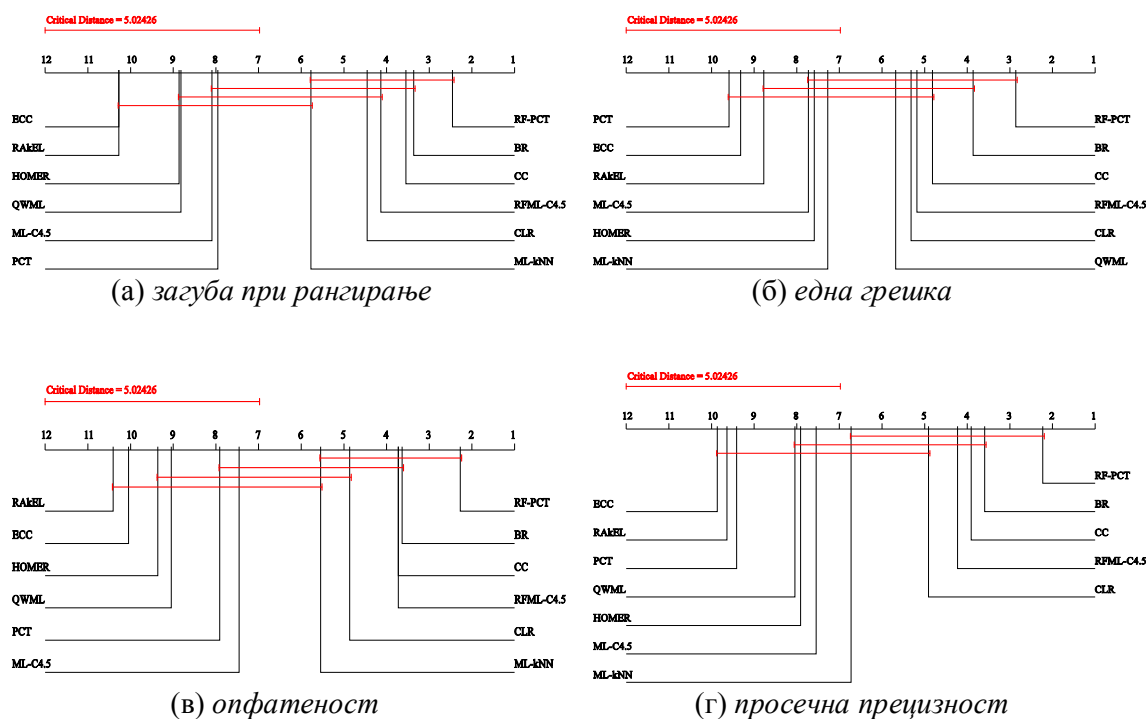
комплетните резултати се прикажани во Табела А.13 до Табела А.16. Дополнително, во Табела 5.3 се прикажани  $p$ -вредностите за проценка на перформансите на методите со *Friedman* тестот. Според критичните дијаграми на податочните множества за кои сите методи генерирале модел за предикција, најдобри перформанси покажа *RF-PCT*. По него следуваат методите *BR*, *CC* и *CLR*. *RF-PCT* има најдобри просечни резултати за сите евалуациски мерки, додека *BR* е втор според *загуба при рангирање*, една *грешка* и *просечна прецизност* и трет според евалуациската мерка *опфатеност*.

Заклучоците за перформансите на методите кои градат ансамбли во однос на нивните основни модели, како и заклучоците за типовите на методи за машинско учење остануваат исти како и кај мерките базирани на примерок и ознака.



**Слика 5.13** Критични дијаграми за мерките базирани на рангирање. Резултати добиени од дополнителниот *Немењи* тест со ниво на значајност од 0.5 на податочните множества за кои сите методи дале резултати

Дополнително, *RF-PCT* има статистички значајно подобри резултати од *HOMER* за евалуациските мерки *опфатеност* и *загуба при рангирање* на податочните множества за кои сите методи дале резултати. Разликата меѓу овие два метода станува статистички значајна и во однос на евалуациската мерка *просечна прецизност*, ако во анализата се вклучат сите податочни множества.



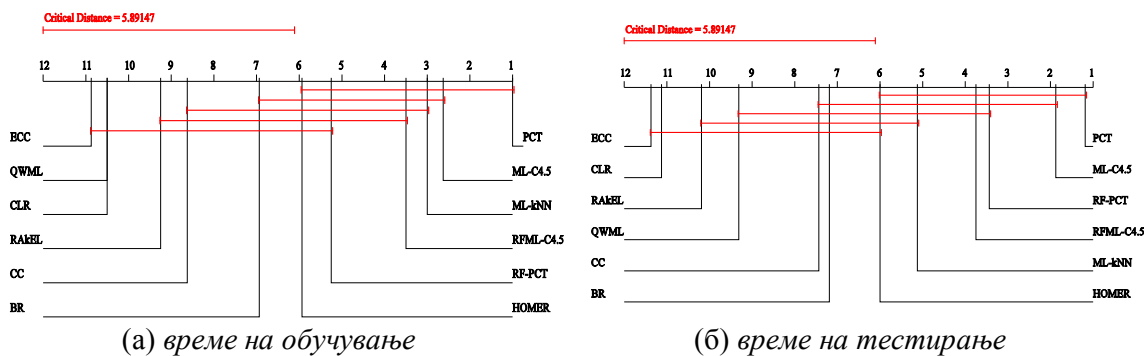
**Слика 5.14** Критични дијаграми за мерките базирани на рангирање. Резултати добиени од дополнителниот Немењи тест со ниво на значајност од 0.5 на сите податочни множества

**Табела 5.3**  $p$ -вредностите за проценка на перформансите на методите за повеќезначна класификација и рангирање со Фридман тестот во однос на евалуациските мерки базирани на рангирање. Ознаката **Сите** се однесува на  $p$ -вредностите пресметани за резултатите на податочните множества за кои сите методи дале резултати. Ознаката **Подмножество** се однесува на  $p$ -вредностите пресметани за резултатите на сите податочни множества

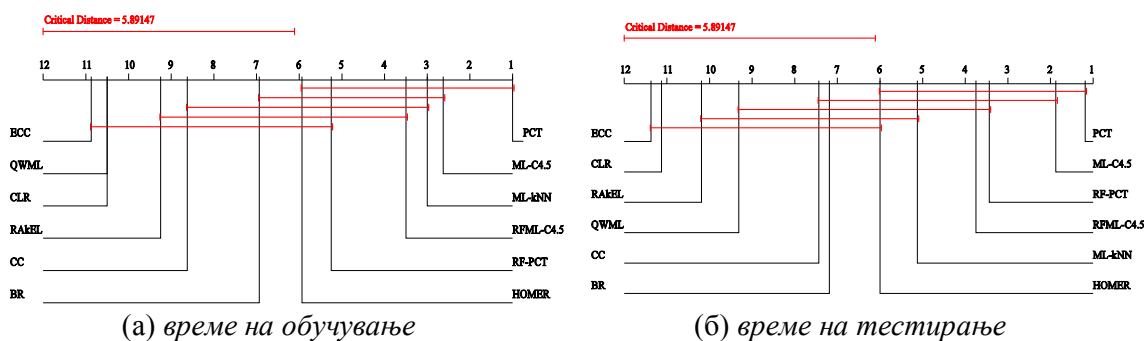
	Сите	Подмножество
<b>Една грешка</b>	$2.2 \cdot 10^{-7}$	$5.3 \cdot 10^{-6}$
<b>Опфатеност</b>	$1 \cdot 10^{-18}$	$2.3 \cdot 10^{-16}$
<b>Загуба при рангирање</b>	$1 \cdot 10^{-18}$	$1.2 \cdot 10^{-16}$
<b>Просечна прецизност</b>	$6.5 \cdot 10^{-14}$	$2 \cdot 10^{-11}$

#### 5.7.2.4 Времиња на обучување на моделите и тестирање

На крајот ги анализираме перформансите на методите во однос на времињата на обучување и тестирање. Критичните дијаграми добиени од статистичката евалуација се прикажани на Слика 5.15 и Слика 5.16. Комплетните резултати се прикажани во Табела А.17 до Табела А.18. Во Табела 5.4 се прикажани  $p$ -вредностите за проценка на перформансите на методите со Фридман тестот.



Слика 5.15 Критични дијаграми за мерките на ефикасност. Резултати добиени од дополнителниот Немењи тест со ниво на значајност од 0.5 на податочните множества за кои сите методи дале резултати



Слика 5.16 Критични дијаграми за мерките на ефикасност. Резултати добиени од дополнителниот Немењи тест со ниво на значајност од 0.5 на сите податочни множества

Табела 5.4  $p$ -вредностите за проценка на перформансите на методите за повеќезначна класификација и рангирање со Фридман тестот во однос на времињата на обучување и тестирање. Ознаката **Сите** се однесува на  $p$ -вредностите пресметани за резултатите на податочните множества за кои сите методи дале резултати. Ознаката **Подмножество** се однесува на  $p$ -вредностите пресметани за резултатите на сите податочни множества

	Сите	Подмножество
Време на обучување	$1 \cdot 10^{-18}$	$1 \cdot 10^{-18}$
Време на тестирање	$1 \cdot 10^{-18}$	$1 \cdot 10^{-18}$

Како што можеше да се очекува, методите базирани на дрвата за одлучување имаат значително повисока ефикасност од SVM-базираните методи. Имено, PCT е најефикасниот метод, а по него следат методите ML-C4.5 и ML-kNN. PCT за разлика од ML-C4.5 користи поткастрување во текот н градење на дрвото што резултира во поголема ефикасност на градењето на дрвото.

Во однос на методите кои покажаа најдобри перформанси за другите евалуациски мерки (базирани на примерок, ознака и рангирање), *RF-PCT* и *HOMER* имаат помали времиња на обучување и тестирање во споредба со другите методи. Притоа, *RF-PCT* е за нијанса подобар за малите податочни множества, додека за големите податочни множества оваа разлика станува поголема.

## 5.8 Заклучок

Во оваа глава беше прикажана експериментална евалуација на најзначајните методи за повеќезначна класификација и рангирање. Иако во последно време, темата на повеќезначна класификација и рангирање привлече значајно истражувачко внимание, ваква обемна експериментална евалуација во литературата сеуште не постои.

Направена е експериментална евалуација на најпопуларните методи за повеќезначна класификација и рангирање на податочни множества од различни области со различна големина и карактеристики во однос на голем број на евалуациски мерки. Конкретно, беа споредени 12-те најмногу цитирани и најзначајни методи за повеќезначна класификација и рангирање. Истите беа поделени во три главни групи: методи кои го трансформираат проблемот (5 методи), методи кои го адаптираат алгоритмот (3 методи) и методи кои градат ансамбли (4 методи). Методите користат три типови на основни алгоритми за машинско учење: машини со носечки вектори (7 методи), дрва за одлучување (4 методи) и  $k$ -најблиски соседи (1 метод). За да се направи комплетна анализа на перформансите на методите за повеќезначна класификација и рангирање од различни перспективи, многу е значајно споредбата на методите да биде направена врз основа на различни евалуациски мерки. Поради тоа анализата на перформансите на методите беше направена врз основа на 18 различни евалуациски мерки кои беа групирани во четири групи: 6 мерки базирани на ознака, 6 мерки базирани на примерок, 4 мерки базирани на рангирање и 2 мерки за ефикасноста на методите. На крај, во експерименталната евалуација беа вклучени 11 бенчмарк повеќезначни класификациски проблеми од 5 апликациски области на употреба: класификација на текст (6 податочни множества), класификација на слики (2 податочни множества), предикција на функција на гени (1 податочно множество), класификација на музика (1 податочен множество) и класификација на видео (1 податочно множество). За да ја процениме статистичката значајност на разликата на перформансите на споредуваните методи користен е *коригиран Фридман тест* и *дополнителниот Немењи тест*. Резултатите од тестовите се прикажани со критични дијаграми.

Врз основа на резултатите добиени од експериментите и статистичката евалуација генерално може да се заклучи дека во однос на евалуациските мерки базирани на примерок најдобри перформанси имаат *RF-PCT* и *HOMER*, а веднаш позади нив се методите *бинарна релевантност (BR)* и *верижниот метод (CC)*. *RF-PCT* ансамблот покажа најдобри перформанси во однос на *прецизност* евалуациската мерка, додека *HOMER* покажа најдобри резултати во однос на *одзивот*. Ова значи дека предикциите направени од *HOMER* се покомплетни, а оние на *RF-PCT* се погзактни. Предноста на *HOMER* во однос на *BR* и *CC* методите е резултат на хиерархијата која *HOMER* ја дефинира над податочното множество. Имено, *HOMER* прво се обидува да ја детектира хиерархиската законитост на означување на примероците во рамките на податочното множество, а потоа преку принципот на хиерархиска елиминација на ознаките ја

користи истата за да ја намали грешката на предвидување. Во однос на перформансите на различните типови на алгоритми за машинско учење, резултатите покажуваат дека *SVM*-базираните методи се подобри на малите податочни множества, додека методите базирани на дрва за одлучување имаат подобри перформанси за големите податочни множества. Исто така, *SVM*-базираните методи покажуваат подобри перформанси за области кои имаат поголем број на обележја.

Како и кај евалуациските мерки базирани на примерок, така и кај мерките базирани на ознака најдобри перформанси покажаа *HOMER*, *RF-PCT*, *BR* и *CC*. Кај овие мерки единствено разликата меѓу *HOMER* и *RF-PCT* е уште повеќе изразена во однос на евалуациските мерки *прецизност* и *odsub*. За евалуациската мерка *прецизност* *RF-PCT* е дури и статистички значајно подобар од *HOMER*.

Според четирите евалуациски мерки кои ги користевме, просечно најдобар рангирачки алгоритам е *RF-PCT*. По него следуваат методите *BR*, *CC* и *CLR*. За разлика од *RF-PCT*, *HOMER* покажа лоши перформанси при рангирањето на ознаките, што резултираше со статистичка значајност на разликата меѓу *RF-PCT* и *HOMER*.

Од анализата на перформансите на методите во однос на времињата на обучување и тестирање, може да се заклучи дека методите базирани на дрва за одлучување имаат значително повисока ефикасност од *SVM*-базираните методи. Имено, *PCT* и *ML-C4.5* се најефикасните методи, а по нив следува *ML-kNN* методот. *RF-PCT* е за нијанса подобар од *HOMER* за малите податочни множества, додека за големите податочни множества оваа разлика станува поголема.

## Глава 6

# Дво-нивовска архитектура за повеќезначна класификација и рангирање

Во оваа глава ќе биде претставена дво-нивовска ефикасна парови на ознаки архитектура (анг. *Two Stage Architecture - TSA*) за повеќезначна класификација и рангирање која е тесно поврзана со методот *калибрирано рангирање на ознака (CLR)* [31]. Основната идеја на оваа архитектура е да ја намали пресметковната комплексност на методите *парови на ознаки* (еден од најголемите недостатоци на овие методи) и притоа да ги задржи или подобри перформансите во однос на прецизноста на предвидување. Со намалување на бројот на модели за предикција, архитектурата ја зголемува пресметковната ефикасност во однос на методите *парови на ознаки*. За подобрување на перформансите во однос на прецизноста на предвидување, архитектурата користи мета-информации кои се добиваат од предикцијата на примероците од множеството за тренирање од соодветно избрани модели.

Во продолжение прво е претставена дво-нивовската архитектура и нејзиниот дизајн, а потоа се претставени три различни варијации за градење на оваа архитектура: Дво-нивовски метод со гласање, дво-нивовски верижен метод и дво-нивовски скратен верижен метод. На крај, направена е анализа на пресметковната комплексност на дво-нивовската архитектура во споредба со пресметковната комплексност на методот *калибрирано рангирање на ознака*.

Оваа глава завршува со анализа и заклучоци донесени врз основа на резултатите добиени од експерименталната евалуација на оваа архитектура. Во експерименталната евалуација се опфатени сите евалуациски мерки претставени во Глава 3, како и времињата на обучување и тестирање.

### 6.1 Методи базирани на дво-нивовска архитектура за повеќезначна класификација и рангирање

Како што беше претставено во Глава 4, конвенционалниот метод *парови на ознаки* [28] конструира  $\binom{Q(Q-1)}{2}$  бинарни класификациски модели  $M_{ij}$  за секој пар од ознаки  $(\lambda_i, \lambda_j)$ ,  $1 \leq i < j \leq Q$ . Моделите  $M_{ij}$  ( $1 \leq i < j \leq Q$ ) се нарекуваат модели на *парови на ознаки*. Секој од нив се обучува со примероците означени со  $\lambda_i$  како позитивни и примероците означени со  $\lambda_j$  како негативни. Примероците кои истовремено се означени со ознаките  $\lambda_i$  и  $\lambda_j$  како и примероците кои не се означени ниту со една од

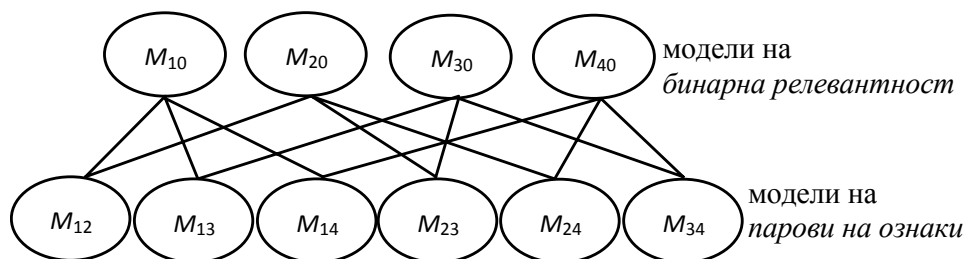
ознаките  $\lambda_i$  и  $\lambda_j$  не учествуваат во обучувањето на моделот  $M_{ij}$ . За да се примени овој метод во контекст на повеќезначна класификација и рангирање потребно е да се направи соодветна трансформација на оригиналното податочното множество која ја нарекуваме *двозначна трансформација* (анг. *two label transformation*).

За разлика од методот *парови на ознаки*, методот *калибрирано рангирање на ознака* [31] дополнително обучува  $Q$  класификациски модели. Иако овие модели се модели на *бинерна релевантност* и се истите кои се користат кај методот *бинерна релевантност* (со нотација  $M_k$  ( $1 \leq k \leq Q$ )), сепак во речникот на *CLR* тие претставуваат модели на *парови на ознаки* ( $M_{k0}$ ), кои се обучуваат со примероците означени со ознаката  $\lambda_k$  како позитивни примероци и сите останати како негативни примероци (негативните примероци се означуваат со вештачката калибрациона ознака  $\lambda_0$ ).

Исто како и методот *парови на ознаки*, во процесот на предвидување *CLR* го користи *алгоритмот за мнозинско гласање*. За да се донесе финалната одлука за означување на даден примерок потребно е да се консултираат  $\frac{Q(Q-1)}{2} + Q$  модели. Како резултат на зголемениот број на модели кои треба да се евалуираат, *CLR* се карактеризира со зголемена пресметковна комплексност во однос на методот *парови на ознаки*. Големиот број на ознаки во повеќезначниот класификациски проблем може значително да влијае на пресметковната комплексност на овој метод, слично како и кај методот *парови на ознаки*.

Дво-нивовската архитектура за повеќезначна класификација и рангирање има за цел да го намали бројот на модели потребни за предвидување на ознаките на даден примерок. Имено, со намалување на нивниот број, кој расте квадратно со бројот на ознаки во повеќезначниот проблем се очекува да се намали пресметковната комплексност на *CLR* методот.

Архитектурата е организирана во две нивоа (Слика 6.1). Првото ниво содржи  $Q$  модели на *бинерна релевантност*  $M_{k0}$ , додека второто ниво содржи  $\frac{Q(Q-1)}{2}$  модели на *парови на ознаки*  $M_{ij}$ . Секој модел  $M_{k0}$  од првото ниво е поврзан со  $Q - 1$  модели  $M_{ij}$  од второто ниво, каде  $k = i$  или  $k = j$  ( $1 \leq i \leq Q - 1, i + 1 \leq j \leq Q$ ). Овој модел ( $M_{k0}$ ) е директно одговорен за предвидување на ознаката  $\lambda_k$  (веројатноста со која примерокот треба да биде означен со ознаката  $\lambda_k$ , односно дали ознаката  $\lambda_k$  е релевантна за конкретниот примерок или не). Од друга страна, секој модел од второто ниво на архитектурата  $M_{ij}$  е поврзан со точно два модели на *бинерна релевантност* ( $M_{i0}$  и  $M_{j0}$ ), кои припаѓаат во првото ниво и истиот ( $M_{ij}$ ) е одговорен за предвидување на ознаките  $\lambda_i$  и  $\lambda_j$  ( $p(\lambda_i) = 1 - p(\lambda_j)$ ).



Слика 6.1 Дво-нивовска архитектура за повеќезначна класификација и рангирање

Во продолжение на оваа глава се претставени три различни варијации за градење на оваа архитектура (дво-нивовски метод со гласање (анг. *Two Stage Voting Method – TSVM*), дво-нивовски верижен метод (анг. *Two Stage Classifier Chains Method – TSCCM*) и дво-нивовски скратен верижен метод (анг. *Two Stage Pruned Classifier Chains Method – TSPCCM*) за повеќезначна класификација и рангирање. Сите методи ја градат истата дво-нивовска архитектура (истата организација, распоред и поврзаност на моделите на бинарна релевантност и парови на ознаки во двете нивоа). Поединечно, секој метод се карактеризира со различен начин на обучување на моделите и предвидување на ознаките. Во продолжение ќе бидат образложени поодделно секоја од варијациите на дво-нивовската архитектура.

### 6.1.1 Дво-нивовски метод со гласање

Кај дво-нивовскиот метод со гласање фазата на тренирање е иста како и кај *CLR* методот. Имено, во оваа фаза се обучуваат сите  $M_{k0}$  модели на бинарна релевантност и  $M_{ij}$  модели на парови на ознаки вклучени во дво-нивовската архитектура. Секој од овие модели независно се обучува со соодветните примероци од множество за тренирање. Процесот на обучување на моделите од првото и второто ниво е прикажано на Слика 6.2. Нотацијата која е користена за опишување на методот е следната:  $(\mathbf{x}_i, \vec{\mathcal{Y}}_i)$  се користи за претставување на  $i$ -тиот ( $i \in 1..N$ ) примерок, каде  $\mathbf{x}_i$  е векторот на влезни обележја, а  $\vec{\mathcal{Y}}_i$  е вектор на ознаки на примерокот.  $\mathcal{L} = \{\lambda_1, \dots, \lambda_Q\}$  е множеството на оригинални ознаки на повеќезначниот проблем.

---

```
procedure Training_FirstLayer( $E = \{(\mathbf{x}_1, \vec{\mathcal{Y}}_1), \dots, (\mathbf{x}_N, \vec{\mathcal{Y}}_N)\}, \mathcal{L}$ )
1: for  $k = 1$  to  $Q$  do
2:    $E_{k0} = \text{SingleLabelTransformation}(E, \lambda_k)$ 
3:    $M_{k0} = \text{TrainingModel}(E_{k0})$ 

procedure Training_SecondLayer( $E = \{(\mathbf{x}_1, \vec{\mathcal{Y}}_1), \dots, (\mathbf{x}_N, \vec{\mathcal{Y}}_N)\}, \mathcal{L}$ )
1: for  $i = 1$  to  $Q - 1$  do
2:   for  $j = i + 1$  to  $Q$  do
3:      $E_{ij} = \text{TwoLabelTransformation}(E, \lambda_i, \lambda_j)$ 
4:      $M_{ij} = \text{TrainingModel}(E_{ij})$ 
```

---

Слика 6.2 Процедура за обучување на *TSVM*.  $E$  множество на примероци,  $\mathcal{L}$  множество на ознаки

За разлика од методот за мнозинско гласање кој во фазата на предвидување ги евалуира сите обучени модели (метод користен за предвидување на ознаките кај *CLR*), дво-нивовскиот метод со гласање преку метод за елиминација ги отфрла сите модели за кои смета дека нема да придонесат значајно кон одлуката за релевантноста на определени ознаки. Тоа значи, дека преку намалување на бројот на евалуираните модели, овој метод ја намалува пресметковната комплексност на *CLR* методот во фазата на тестирање, а со тоа го намалува и времето за предвидување на ознаките на конкретен примерок.

Во фазата на тестирање кај дво-нивовската архитектура, прво се евалуираат сите модели  $M_{k0}$  ( $k \in 1 \dots Q$ ) кои се наоѓаат во првото ниво од архитектурата. Овие модели

ги определуваат релевантните ознаки за конкретен примерок. Секој модел  $M_{k0}$  ја дава веројатноста со која даден примерок е означен со ознаката  $\lambda_k$ . Доколку оваа веројатност е доволно мала (под некој предефиниран праг), соодветната ознака  $\lambda_k$  директно се прогласува за нерелевантна. Тоа значи дека ознаката  $\lambda_0$  ќе биде подобро рангирана од ознаката  $\lambda_k$ , под претпоставка дека сите модели  $M_{k0}$  донеле правилна одлука. Во овој случај може да се заклучи дека моделите на *парови на ознаки*  $M_{ij}$  во второто ниво на архитектурата кои се поврзани со моделот  $M_{k0}$  ( $i = k$  или  $j = k$ ) не мора да се евалуираат за конкретниот примерок, затоа што моделот на *бинарна релевантност*  $M_{k0}$  од првото ниво на архитектурата сугерирал дека ознаката  $\lambda_k$  припаѓа на множеството на нерелевантни ознаки. Имено, моделот  $M_{k0}$  сугерира дека евалуација на моделите на *парови на ознаки*  $M_{ij}$  ( $i = k$  или  $j = k$ ) не е потребна, затоа што неговата одлука е прилично сигурна.

Во однос на пресметковната комплексност, тоа значи дека за секоја ознака  $\lambda_k$  која ќе биде прогласена за нерелевантна ознака од моделот  $M_{k0}$ , бројот на модели на *парови на ознаки*  $M_{ij}$  од второто ниво на архитектурата кои треба да се евалуираат се намалува за  $Q - 1$ .

За да се донесе одлука кои ознаки се нерелевантни за конкретен примерок, т.е., кои модели на *парови на ознаки*  $M_{ij}$  од второто ниво на архитектурата не е потребно да се евалуираат, се воведува праг на одлука  $t$  ( $0 \leq t \leq 1$ ).

Како што беше претходно кажано, секој примерок прво ги евалуира сите модели  $M_{k0}$  од првото ниво на дво-нивовската архитектура. Потоа, излезот на секој  $M_{k0}$  модел се конвертира во веројатност на предвидување и се споредува со прагот на одлука  $t$ .

- Ако веројатноста на предвидување е поголема од прагот на одлука, примерокот за тестирање се проследува на моделите на *парови на ознаки*  $M_{ij}$  од второто ниво на архитектурата кои се поврзани со  $M_{k0}$  моделот.
- Ако веројатноста на предвидување е помала од прагот на одлука, примерокот за тестирање не се проследува на второто ниво од архитектурата, т.е., не се евалуираат моделите на *парови на ознаки*  $M_{ij}$  кои се поврзани со  $M_{k0}$  моделот.

Од гледна точка на моделите на *парови на ознаки*  $M_{ij}$ , можат да се појават три различни случаи во процесот на гласање од аспект на веројатностите на предвидување на моделите на *бинарна релевантност*  $M_{i0}$  и  $M_{j0}$ :

1. Веројатностите на предвидување на двата модели на *бинарна релевантност*  $M_{i0}$  и  $M_{j0}$  кои се поврзани со моделот на *парови на ознаки*  $M_{ij}$  се поголеми од прагот на одлука  $t$ .
2. Само една од веројатностите на предвидување на двата модели на *бинарна релевантност*  $M_{i0}$  и  $M_{j0}$  кои се поврзани со моделот на *парови на ознаки*  $M_{ij}$  е поголема од прагот на одлука  $t$ .
3. Веројатностите на предвидување на обата модели на *бинарна релевантност*  $M_{i0}$  и  $M_{j0}$  кои се поврзани со моделот на *парови на ознаки*  $M_{ij}$  се помали од прагот на одлука  $t$ .

Во првиот случај тестирачкиот примерок од првото ниво на архитектурата се проследува на второто ниво, се евалуира моделот на *парови на ознаки*  $M_{ij}$  и неговата

одлука (предикција) се претвора во глас за една од ознаките  $\lambda_i$  или  $\lambda_j$ . Во вториот случај, моделот на *парови на ознаки*  $M_{ij}$  не се евалуира и неговиот глас оди директно за ознаката чиј модел на *бинарна релевантност* има веројатност на предвидување поголема од прагот на одлука. Во третиот случај моделот на *парови на ознаки*  $M_{ij}$  не се евалуира и не гласа за ниту една од ознаките  $\lambda_i$  или  $\lambda_j$ . На крај, гласовите добиени од моделите на *бинарна релевантност*  $M_{k0}$  и моделите на *парови на ознаки*  $M_{ij}$  (за кои барем еден од моделите на *бинарна релевантност*  $M_{i0}$  или  $M_{j0}$  имал веројатност на предвидување поголема од прагот на одлука) се сумираат за да се донесе финалната одлука за рангот на ознаките преку бројот на освоените гласови. Процесот на тестирање е прикажан на Слика 6.3.

Сите ознаки  $\lambda_k$  кои имаат помалку гласови од вештачката калибрациона ознака  $\lambda_0$  се рангираат според веројатностите на предвидување добиени од моделите на *бинарна релевантност*  $M_{k0}$ .

---

```
procedure Testing_TSVM(x,t)
1: for k = 1 to Q do
2:   probabilities[Mk0] = classify(Mk0(x))
3:   if probabilities[Mk0] > 0.5 then
4:     votes[λk]++
5:   else
6:     votes[λ0]++
7:   for i = 1 to Q - 1 do
8:     for j = i + 1 to Q do
9:       if probabilities[Mi0] > t & probabilities[Mj0] > t
10:        //first case
11:        probabilities[Mij] = classify(Mij(x))
12:        if probabilities[Mij] > 0.5 then
13:          votes[λi]++
14:        else
15:          votes[λj]++
16:      else
17:        //second case
18:        if probabilities[Mi0] > t then
19:          votes[λi]++
20:        if probabilities[Mij] > t then
21:          votes[λj]++
22: order(votes[],probabilities[])
```

---

Слика 6.3 Процедура за тестирање на даден примерок  $x$  со дво-нивовскиот метод со гласање

Со зголемување на вредноста на прагот на одлука, бројот на модели на *парови на ознаки*  $M_{ij}$  кои се евалуираат се намалува. За  $t = 1$  примероците не се проследуваат во второто ниво од дво-нивовската архитектура и одлуката за ознаките се носи само од моделите од првото ниво. Во овој случај, сите ознаки за кои моделите на *бинарна релевантност*  $M_{k0}$  имале веројатност на предвидување поголема од 0.5 добиваат по

еден единствен глас (бидејќи моделите од второто ниво не се евалуираат). Вештачката калибрациона ознака  $\lambda_0$  ги добива сите останати гласови од  $M_{k0}$  моделите (оние кои имале веројатност на предвидување помала од 0.5). Ако бројот на такви модели е поголем од еден, тогаш ознаката  $\lambda_0$  е прво-рангирана и ниту една од оригиналните ознаки не се прогласува за релевантна. Бидејќи рангирањето на ознаките е направено според веројатностите на предвидување добиени од моделите на *бинарна релевантност*  $M_{k0}$ , перформансите на дво-нивовскиот метод со гласање се еднакви со перформансите на метод на *бинарна релевантност*, но само во однос на евалуациските мерки за рангирање.

Кога прагот на одлука  $t = 0.5$ , перформансите на дво-нивовскиот метод со гласање се еднакви со перформансите на методот на *бинарна релевантност* во однос на евалуациските мерки базирани на примерок и ознака. Имено, во овој случај вештачката калибрациона ознака  $\lambda_0$  ги добива гласовите од сите модели на *бинарна релевантност*  $M_{k0}$  што имале веројатност на предвидување помала од 0.5. Моделите на *бинарна релевантност*  $M_{k0}$  што имале веројатност на предвидување поголема од 0.5 ги пропуштаат соодветните примероци на второто ниво и гласаат за ознаките  $\lambda_k$ . Сметајќи ги гласовите од второто ниво кои ќе ги добијат, овие ознаки ќе бидат подобро рангирани од вештачката калибрациона ознака и истите ќе бидат прогласени за релевантни. Во споредба со методот *бинарна релевантност*, единствено може да се разликува предвидувањето за рангот на релевантните ознаки што доведува до разлика на перформансите во однос на евалуациските мерки за рангирање. За примерот прикажан на Слика 6.1, ако за конкретен примерок моделите  $M_{20}$  и  $M_{30}$  имаат веројатност на предвидување поголема од 0.5, ознаката  $\lambda_0$  ќе освои 2 гласа, а ознаките  $\lambda_2$  и  $\lambda_3$  ќе имаат 3 односно 4 гласа. За онаката  $\lambda_2$  сигурно ќе гласаат моделите  $M_{20}$ ,  $M_{12}$  и  $M_{24}$ , за ознаката  $\lambda_3$  ќе гласаат моделите  $M_{30}$ ,  $M_{13}$  и  $M_{34}$ , а моделот  $M_{23}$  ќе одлучи која од ознаките  $\lambda_2$  или  $\lambda_3$  ќе добие 4-ти глас и ќе биде прво рангирана.

Од друга страна, за вредност на прагот на одлука  $t = 0$ , сите модели на *парови на ознаки*  $M_{ij}$  се евалуираат и од дво-нивовскиот метод со гласање се добива *CLR* методот.

### 6.1.2 Дво-нивовски верижен метод

Како што претходно споменавме, дво-нивовскиот метод со гласање го намалува единствено бројот на евалуирани модели во процесот на предвидување на ознаките на тестирачките примероци. Меѓутоа, наместо предикциите на моделите од првото ниво да се користат само за вклучување или исклучување на одредени модели од второто ниво, идејата на овој метод е на моделите од второто ниво да им се пренесе и комплетната информација (одлука, веројатност) на моделите од првото ниво со надеж дека искористувајќи ја оваа дополнителна информација моделите од второто ниво ќе можат да донесат поквалитетни одлуки. За таа цел, дво-нивовскиот верижен метод воведува така наречена верига од класификатори која е поставена меѓу моделите од првото и второто ниво на дво-нивовската архитектура. Истата може да се опише како проширување на просторот на обележја (додавање на ново обележје) на секој примерок кој учествува во процесот на обучување или тестирање на моделите од второто ниво на архитектурата  $M_{ij}$  со веројатностите на предвидување на сите модели на *бинарна релевантност*  $M_{k0}$  од првото ниво. Тоа значи дека по секоја врска која постои меѓу моделите од првото и второто ниво се проследува веројатноста на предвидување на моделот на *бинарна релевантност*  $M_{k0}$  ( $k \in 1 \dots Q$ ) за конкретниот примерок. Поради

употребата на верижната структура, овој метод е наречен дво-нивовски верижен метод (*TSCCM*) за повеќезначна класификација и рангирање.

Сите модели од првото ниво на архитектурата во дво-нивовскиот верижен метод се обучуваат на ист начин како и во дво-нивовскиот метод со гласање. Единствената разлика меѓу *TSCCM* и *TSVM* е во фазата на обучување на моделите од второто ниво на архитектурата. Имено, во оваа фаза прво се евалуираат сите модели  $M_{k0}$  од првото ниво со примероците за обучување, а потоа просторот на влезни обележја на истите примероци се проширува со веројатностите на предвидување добиени од моделите  $M_{k0}$  (векторите на влезни обележја се прошируваат со  $Q$  нови обележја). На крај сите примероци со проширени вектори на обележја учествуваат (се користат) во процесот на обучување на моделите на *парови од ознака*  $M_{ij}$  од второто ниво на архитектурата. Користејќи ја истата нотација како и кај дво-нивовскиот метод со гласање, процесот на обучување е прикажан на Слика 6.4.

---

```
procedure Training_FirstLayer( $E = \{(\mathbf{x}_1, \vec{y}_1), \dots, (\mathbf{x}_N, \vec{y}_N)\}, \mathcal{L}$ )
1: for  $k = 1$  to  $Q$  do
2:    $E_{k0} = \text{SingleLabelTransformation}(E, \lambda_k)$ 
3:    $M_{k0} = \text{TrainingModel}(E_{k0})$ 
4:  $E' \leftarrow \{\}$ 
5: for  $(\mathbf{x}, \vec{y}) \in E$  do
6:    $\mathbf{x}' \leftarrow \{\mathbf{x}\}$ 
7:   for  $k = 1$  to  $Q$  do
8:     probabilities $[M_{k0}] = \text{classify}(M_{k0}(\mathbf{x}))$ 
9:      $\mathbf{x}' \leftarrow \mathbf{x}' \cup \{\text{probabilities}[M_{k0}]\}$ 
10:   $E' \leftarrow E' \cup (\mathbf{x}', \vec{y})$ 

procedure Training_SecondLayer( $E' = \{(\mathbf{x}'_1, \vec{y}'_1), \dots, (\mathbf{x}'_N, \vec{y}'_N)\}, \mathcal{L}$ )
1: for  $i = 1$  to  $Q - 1$  do
2:   for  $j = i + 1$  to  $Q$  do
3:      $E_{ij}' = \text{TwoLabelTransformation}(E', \lambda_i, \lambda_j)$ 
4:      $M_{ij}' = \text{TrainingModel}(E_{ij}')$ 
```

---

Слика 6.4 Процедура за обучување на *TSCCM*.  $E$  множество на примероци,  $\mathcal{L}$  множество на ознаки и  $E'$  е множеството на примероци со проширен вектор на влезни обележја

Истата верижна структура се користи и во процесот на тестирање. Тој е аналоген на процесот на обучување, односно, пред да се проследи тестирачкиот примерок од првото на второто ниво на архитектурата, прво се проширува неговиот вектор на влезни обележја со веројатностите на предвидување добиени од моделите  $M_{k0}$  од првото ниво, а потоа примерокот ги евалуира моделите од второто ниво и се донесува финална одлука за рангот на ознаките. Процесот на тестирање е прикажан на Слика 6.5.

---

```
procedure Testing_TSVM(x, t)
1: x' ← {x}
2: for k = 1 to Q do
3:   probabilities[Mk0] = classify(Mk0(x))
4:   if probabilities[Mk0] > 0.5 then
5:     votes[λk]++
6:   else
7:     votes[λ0]++
8:   x' ← x' ∪ {probabilities[Mk0]}
9: for i = 1 to Q − 1 do
10:  for j = i + 1 to Q do
11:    if probabilities[Mi0] > t & probabilities[Mj0] > t
12:      //first case
13:      probabilities[Mij] = classify(Mij(x'))
14:      if probabilities[Mij] > 0.5 then
15:        votes[λi]++
16:      else
17:        votes[λj]++
18:    else
19:      //second case
20:      if probabilities[Mi0] > t then
21:        votes[λi]++
22:      if probabilities[Mj0] > t then
23:        votes[λj]++
24: order(votes[], probabilities[])
```

---

Слика 6.5 Процедура за тестирање на даден примерок **x** со дво-нивовскиот верижен метод

### 6.1.3 Дво-нивовски скратен верижен метод

Како што претходно споменавме, кај дво-нивовскиот верижен метод, просторот на влезни обележја на примероците се проширува со веројатностите на предвидување на сите модели на *бинарна релевантност*  $M_{k0}$  ( $k \in 1 \dots Q$ ) од првото ниво. Во ова поглавје предлагаме модификација на дво-нивовскиот верижен метод во која просторот на влезни обележја на примероците кои учествуваат во процесот на обучување или тестирање на секој модел  $M_{ij}$  од второто ниво на архитектурата се проширува само со веројатностите на предвидување на моделите на *бинарна релевантност*  $M_{i0}$  и  $M_{j0}$  од првото ниво. Тоа значи дека просторот на влезни обележја наместо со  $Q$  предикции се проширува само со две предикции (веројатностите на предвидување на  $M_{i0}$  и  $M_{j0}$  моделите од првото ниво). Проширувањето на просторот на обележја само со овие две предикции е направено под претпоставка дека предикциите на  $M_{i0}$  и  $M_{j0}$  моделите се најрелевантни за  $M_{ij}$  моделот, затоа што единствено тој одлучува за ознаките  $\lambda_i$  и  $\lambda_j$ , односно со која од овие две ознаки конкретен примерок е означен. Овој метод е наречен дво-нивовскиот скратен верижен метод (*TSPCCM*) Користејќи ја нотацијата како и во претходните две методи, процесите на обучување и тестирање на *TSPCCM* се прикажани на Слика 6.6 и Слика 6.7 соодветно.

---

```
procedure Training_FirstLayer( $E = \{(\mathbf{x}_1, \vec{y}_1), \dots, (\mathbf{x}_N, \vec{y}_N)\}, \mathcal{L}$ )
1: for  $k = 1$  to  $Q$  do
2:    $E_{k0} = \text{SingleLabelTransformation}(E, \lambda_k)$ 
3:    $M_{k0} = \text{TrainingModel}(E_{k0})$ 

procedure Training_SecondLayer( $E' = \{(\mathbf{x}'_1, \vec{y}_1), \dots, (\mathbf{x}'_N, \vec{y}_N)\}, \mathcal{L}$ )
1: for  $i = 1$  to  $Q - 1$  do
2:   for  $j = i + 1$  to  $Q$  do
3:      $E' \leftarrow \{\}$ 
4:     for  $(\mathbf{x}, \vec{y}) \in E$  do
5:        $\mathbf{x}' \leftarrow \{\mathbf{x}\}$ 
6:       probabilities $[M_{i0}] = \text{classify}(M_{i0}(\mathbf{x}))$ 
7:        $\mathbf{x}' \leftarrow \mathbf{x}' \cup \{\mathbf{probabilities}[M_{i0}]\}$ 
8:       probabilities $[M_{j0}] = \text{classify}(M_{j0}(\mathbf{x}))$ 
9:        $\mathbf{x}' \leftarrow \mathbf{x}' \cup \{\mathbf{probabilities}[M_{j0}]\}$ 
10:     $E'_{ij} = \text{TwoLabelTransformation}(E', \lambda_i, \lambda_j)$ 
11:     $M_{ij} = \text{TrainingModel}(E'_{ij})$ 
```

---

Слика 6.6 Процедура за обучување на TSPCCM.  $E$  множеството на примероци,  $\mathcal{L}$  множеството на ознаки и  $E'$  е множеството на примероци со проширен вектор на влезни обележја

---

```
procedure Testing_TSPCCM( $\mathbf{x}, t$ )
1: for  $k = 1$  to  $Q$  do
2:   probabilities $[M_{k0}] = \text{classify}(M_{k0}(\mathbf{x}))$ 
3:   if probabilities $[M_{k0}] > 0.5$  then
4:     votes $[\lambda_k]++$ 
5:   else
6:     votes $[\lambda_0]++$ 
7:   for  $i = 1$  to  $Q - 1$  do
8:     for  $j = i + 1$  to  $Q$  do
9:       if probabilities $[M_{i0}] > t$  & probabilities $[M_{j0}] > t$ 
10:        //first case
11:         $\mathbf{x}' \leftarrow \{\mathbf{x}\}$ 
11:         $\mathbf{x}' \leftarrow \mathbf{x}' \cup \{\mathbf{probabilities}[M_{i0}]\}$ 
12:         $\mathbf{x}' \leftarrow \mathbf{x}' \cup \{\mathbf{probabilities}[M_{j0}]\}$ 
11:        probabilities $[M_{ij}] = \text{classify}(M_{ij}(\mathbf{x}'))$ 
12:        if probabilities $[M_{ij}] > 0.5$  then
13:          votes $[\lambda_i]++$ 
14:        else
15:          votes $[\lambda_j]++$ 
16:        else
17:          //second case
18:          if probabilities $[M_{i0}] > t$  then
19:            votes $[\lambda_i]++$ 
20:          if probabilities $[M_{j0}] > t$  then
```

---

```
21:          votes[λj++]  
22: order(votes[], probabilities[])
```

Слика 6.7 Процедура за тестирање на даден примерок  $x$  со дво-нивовскиот скратен верижен метод

## 6.2 Пресметковна комплексност на дво-нивовската архитектура за повеќезначна класификација и рангирање

Пресметковните комплексности на методот *калибрирано рангирање на ознака* и дво-нивовскиот метод со гласање во фазата на обучување се еднакви. Тие се еднакви од едноставна причина што моделите кои се обучуваат со овие два метода се практично идентични. Од друга страна, и покрај еднаквиот број на обучени модели, пресметковните комплексности на дво-нивовскиот верижен метод и дво-нивовскиот скратен верижен метод се поголеми како резултат на проширувањето на векторите на обележја на примероците при обучувањето на второто ниво на архитектурата.

Зголемувањето на пресметковната комплексност на дво-нивовскиот верижен метод во фазата на обучување споредено со *CLR* методот зависи од бројот на ознаки во повеќезначниот класификациски проблем. Ако овој број е мал во споредба со оригиналниот број на обележја во повеќезначниот класификациски проблем, тогаш пресметковната комплексност на дво-нивовскиот верижен метод во фазата на обучување е слична со пресметковната комплексност на *CLR* методот во фазата на обучување. Во поголемиот број на класификациски проблеми бројот на обележја е значително поголем од бројот на ознаки (повеќе од 10 пати), што значи дека пресметковната комплексност на дво-нивовскиот верижен метод е незначително поголема од пресметковната комплексност на *CLR* методот. Пресметковната комплексност на дво-нивовскиот скратен верижен метод е практично еднаква на пресметковната комплексност на *CLR* методот, затоа што во овој метод векторите на обележја на примероците за обучување се прошируваат само со две дополнителни обележја при обучувањето на секој модел  $M_{ij}$  од второто ниво на архитектурата.

Во продолжение на ова поглавје е анализирана пресметковната комплексност на дво-нивовската архитектура во фазата на предвидување, во споредба со пресметковната комплексност на *CLR* методот. Имајќи во предвид дека целта на оваа архитектура е да го намали бројот на евалуирани модели во фазата на предвидување, најголемиот придонес се очекува токму во оваа фаза. Токму затоа оваа анализа е направена во деталност и е квантитативно изразена. Во продолжение, наместо терминот „пресметковна комплексност во фазата на предвидување“ го употребуваме терминот „пресметковна комплексност“, со единствена цел да го олесниме изразувањето.

Пресметковната комплексност на дво-нивовската архитектура значително се разликува од пресметковната комплексност на *CLR* методот. Пресметковната комплексност на *CLR* методот ( $O_{CLR}$ ) може да биде дефинирана како сума од пресметковните комплексности на моделите на *бинарна релевантност* ( $O_{BR}$ ) и моделите на *парови на ознаки* ( $O_P$ ):

$$O_{CLR} = O_{BR} + O_P \quad (6.1)$$

Пресметковната комплексност на дво-нивовската архитектура ( $O_{TSA}$ ) може да биде дефинирана како сума од пресметковните комплексности на моделите кои се наоѓаат во првото ниво од архитектурата ( $O_{FL}$ ) и моделите кои се наоѓаат во второто ниво од архитектурата ( $O_{SL}$ ):

$$O_{TSA} = O_{FL} + O_{SL} \quad (6.2)$$

Пресметковната комплексност на моделите од првото ниво на дво-нивовската архитектура и пресметковната комплексност на моделите на *бинарна релевантност* од *CLR* методот се еднакви ( $O_{FL} = O_{BR}$ ). Во двете методи, моделите се практично исти. Тие се обучуваат со истите примероци за обучување и сите мора да се евалуираат од примероците за тестирање за тие да бидат означени со соодветни ознаки.

Главната разлика во пресметковните комплексности меѓу *CLR* методот и дво-нивовската архитектура е во моделите на *парови на ознаки* на *CLR* методот и второто ниво на архитектурата. Како што споменавме во претходното поглавје, ако прагот на одлука се постави на нула ( $t = 0$ ), тогаш сите модели од второто ниво на дво-нивовската архитектура ќе бидат евалуирани. Тоа значи дека пресметковните комплексности на моделите на *парови на ознаки* на *CLR* методот и второто ниво на архитектурата кај дво-нивовскиот метод со гласање се еднакви ( $O_{SL} = O_P$ ). Оваа еднаквост е само апроксимативна ( $O_{SL} \approx O_P$ ) кај двете методи базирани на верига структурата, како резултат на проширувањето на просторот на обележја на примероците во второто ниво на архитектурата кое дополнително ја зголемува комплексноста на моделите сместени во ова ниво. Ако вредноста на прагот на одлука се постави на еден ( $t = 1$ ), тогаш ниту еден модел од второто ниво на дво-нивовската архитектура нема да биде евалуиран. Во тој случај  $O_{SL} = 0$  и  $O_{TSA} = O_{FL} = O_{BR}$ . За вредности на прагот помеѓу нула и еден ( $0 < t < 1$ ),  $O_{SL} = r \cdot O_P$  каде  $r$  ( $0 < r < 1$ ) е *фактор на намалување* на пресметковната комплексност на второто ниво на дво-нивовската архитектура во однос на моделите на *парови на ознаки* на *CLR* методот. *Факторот на намалување* е тесно поврзан со концептот на кардиналност на ознаките [2], т.е., просечниот број на ознаки по примерок во дадено податочно множество.

Како што споменавме во претходното поглавје, ако модел на *бинарна релевантност* од првото ниво на дво-нивовската архитектура има веројатност на предвидување поголема од вредноста на прагот на одлука за конкретен примерок, тогаш примерокот се проследува на соодветните модели на *парови на ознаки* од второто ниво на архитектурата. Ако сите модели од првото ниво го проследат примерокот на второто ниво, тогаш вкупниот број на модели на *парови на ознаки* кои ќе бидат евалуирани е  $Q \cdot (Q - 1)/2$  (дво-нивовскиот метод со гласање станува *CLR*). Во случај само еден модел  $M_{k_0}$  ( $1 \leq k \leq Q$ ) од првото ниво да го проследи примерокот во второто ниво, тогаш тоа значи дека само ознаката  $\lambda_k$  е релевантна ознака за овој примерок (само  $\lambda_k$  е подобро рангирана од калибрационата ознака  $\lambda_0$ ) како резултат на добиените гласови од моделите  $M_{ik}$  и  $M_{kj}$  ( $1 \leq i < k, k \leq j < Q$ ) од второто ниво на архитектурата. Пресметковната комплексност во второто ниво од архитектурата во оваа ситуација е сеуште еднаква на нула ( $O_{SL} = 0$ ), затоа што моделите  $M_{ik}$  и  $M_{kj}$  не се евалуираат туку истите директно гласаат за ознаката  $\lambda_k$ . Ако два модели од првото ниво на дво-нивовската архитектура  $M_{m_0}$  и  $M_{n_0}$  ( $1 \leq m \leq Q, 1 \leq n \leq Q, m \neq n$ ) го

проследат примерокот на второто ниво, тогаш само моделот  $M_{mn}$  ( $m < n$ ) или  $M_{nm}$  ( $n < m$ ) се евалуира и ознаката која ќе го освои гласот од овој модел ( $\lambda_m$  или  $\lambda_n$ ) ќе биде прво рангирана, додека другата ознака ќе биде второ рангирана. Вештачката калибрациона ознака  $\lambda_0$  во овој случај ќе биде трето рангирана ознака.

Ако претпоставиме идеална ситуација во која моделите на *бинарна релевантност* од првото ниво на архитектурата направат идеална предикција (100% прецизност), бројот на евалуирани модели од второто ниво е  $rl \cdot (rl - 1)/2$  каде  $rl$  е бројот на релевантни ознаки, односно бројот на ознаки кои се подобро рангирани од калибрационата ознака  $\lambda_0$  за соодветниот примерок. Тоа значи дека просечниот број на евалуирани модели од второто ниво е  $l_c \cdot (l_c - 1)/2$ , каде  $l_c$  е кардиналноста на ознаките на повеќезначното податочно множество. Во овој идеален случај (прецизност на предвидување од 100% на моделите на *бинарна релевантност* од првото ниво), *факторот на намалување* е определен со следната релација:

$$r = \frac{l_c \cdot (l_c - 1)}{Q \cdot (Q - 1)} \quad (6.3)$$

Сепак реалната вредност на овој параметар е определена со  $r_{real}$ :

$$r_{real} = \frac{a_{brmf} \cdot (a_{brmf} - 1)}{Q \cdot (Q - 1)} \quad (6.4)$$

каде  $a_{brmf}$  е просечниот број на модели од првото ниво на архитектурата што имале поголема веројатност на предвидување од прагот на одлука  $t$ . Вредноста на  $r_{real}$  го покажува односот на евалуирани модели на *парови на ознаки* во процесот на тестирање кај дво-нивовската архитектура и *CLR* методот.

### 6.3 Експерименти

Во ова поглавје е прикажан експерименталниот дизајн (основни класификатори, поставување на параметри и статистичка евалуација) како и добиените резултати анализирани во однос на дво-нивовската архитектура за повеќезначна класификација и рангирање и нејзините три методи.

Целта на овој дел од истражувањето е да одговориме на следниве прашања:

- Дали дво-нивовската архитектура ги подобрува перформансите и пресметковната комплексност на *CLR* методот во фазата на тестирање?
- Дали методите *TSCCM* и *TSPCCM* се подобри од *TSVM* како резултат на мета-информациите кои се додаваат на моделите од второто ниво на дво-нивовската архитектура?
- Кој од методите *TSCCM* и *TSPCCM* е подобар и поефикасен?

- Каква е зависноста на перформансите на дво-нивовската архитектура од прагот на одлука?

За да одговориме на поставените прашања (конкретно првото и второто прашање) направена е споредба на перформансите на дво-нивовскиот метод со гласање, дво-нивовскиот верижен метод, дво-нивовскиот скратен верижен метод и методите *парови на ознаки* за повеќезначна класификација и рангирање на 9-те повеќезначни класификациски проблеми на кои методите *парови од ознака* генерираа модел. За евалуација на перформансите на методите во однос на прецизноста на предвидување и ефикасноста, користени се истите евалуациски мерки како и во претходната глава: евалуациски мерки базирани на примерок, евалуациски мерки базирани на ознака, евалуациски мерки базирани на рангирање и времињата на обучување и тестирање.

За да одговориме на третото прашање директно ги споредивме перформансите на двата метода (*TSCCM* и *TSPCCM*). Дополнително, користен е *Вилкоксон* (анг. *Wilcoxon*) *тестот* [77] за да се процени дали разликата во перформансите меѓу овие две методи е статистички значајна.

За да ја определиме зависноста која постои меѓу перформансите на дво-нивовската архитектура и прагот на одлука, анализирани се перформансите на трите методи во однос на сите евалуациски мерки и времињата на тестирање за вредностите на прагот на одлука 0.0 до 1.0 со чекор 0.1.

### 6.3.1 Експериментален дизајн

Експерименталниот дизајн опфаќа оптимизирање и поставување на параметрите на методите, избор на основни класификатори кај методите, како и изборот на тестови за статистичка евалуација на резултатите за определување на статистичката значајност на добиените перформанси.

При споредбата на дво-нивовската архитектура и методите *парови на ознаки* (*CLR* и *QWML*) користена е библиотеката за повеќезначна и хиерархиска повеќезначна класификација и рангирање *MULAN*<sup>1</sup> [68] која како основна платформа ја користи библиотеката *WEKA*<sup>2</sup> [69]. Сите експерименти беа направени на сервер со Интел Ксеон (анг. *Intel Xeon*) процесор на 2.50GHz и 64GB RAM меморија со Федора 14 (анг. *Fedora 14*) оперативен систем. Во продолжение на ова поглавје прво се претставени основните класификатори кои се користат од методите за повеќезначна класификација и рангирање кои го трансформираат проблемот, а потоа е опишан начинот на поставување на параметрите на споредуваните методи.

#### 6.3.1.1 Основни класификатори, поставување на параметри и статистичка евалуација

Споредуваните методи *парови на ознаки* користат машини со носечки вектори како основни класификатори. За имплементацијата на машините со носечки вектори е користена *LIBSVM*<sup>3</sup> [70] библиотеката. Конкретно, користени се машини со носечки вектори со Гаусов кернел. Кернел параметарот *гама* (анг. *gamma*) и казната *C* се определени преку 10-кратна вкрстена валидација на сите примероци од податочното

<sup>1</sup> <http://mulan.sourceforge.net/>

<sup>2</sup> <http://www.cs.waikato.ac.nz/ml/weka/>

<sup>3</sup> <http://www.csie.ntu.edu.tw/~cjlin/libsvm/>

множество за обучување. Вредностите  $2^{-15}$ ,  $2^{-13}$ , ...,  $2^1$ ,  $2^3$  се користени за определување на параметарот  $\gamma$ , додека  $C$  е определен од вредностите  $2^{-5}$ ,  $2^{-3}$ , ...,  $2^{13}$ ,  $2^{15}$ . По изборот на оптималните вредности на параметрите одделно за секој метод на *парови на ознаки* и секое податочо множество, методите се обучени со сите примероци од соодветното податочо множество за обучување и евалуирани со примероците од податочо множество за тестирање.

Единствен параметар што треба да се постави кај дво-нивовската архитектура е прагот на одлука  $t$ . Вредноста на прагот на одлука е определен со 5-кратна вкрстена валидација од вредностите 0.005 до 0.01 со чекор 0.001, 0.01 до 0.1 со чекор 0.01 и 0.1 до 1.0 со чекор 0.05, со цел да бидат постигнати оптимални резултати во однос на перформансите евалуациски мерки и времињата на тестирање. Користени се три различни опсежи со променлив чекор. Чекорите се помали во првите два опсежи затоа што во овие опсежи се очекуваат најголеми промени на перформансите на архитектурата. Останатите два методи на *парови на ознаки* ( $CLR$  и  $QWML$ ), освен кернел параметарот  $\gamma$  и казната  $C$  немаат други параметри за поставување.

За разлика од претходната глава во која беше направена статистичка евалуација на резултатите добиени од сите методи на сите податочни множества, во ова истражување за да ја процениме статистичката значајност на разликата на перформансите на дво-нивовската архитектура во однос на непосредните методи со кои таа е споредена, како што е предложено во истражувањето на Demšar [74], користен е *Вилкоксон тестот* [77]. Овој тест, исто како и *Фридман тестот* [72] е непараметарски и истиот се користи за да се процени дали постои статистички значајна разлика меѓу резултатите на две методи добиени на поголемо множество од податочни множества.

### 6.3.2 Резултати и дискусија

Резултатите кои се добиени од експерименталната евалуација на предложената архитектура ( $TSA$ ) и нејзините три метода ( $TSVM$ ,  $TSCCM$  и  $TSPCCM$ ),  $CLR$  и  $QWML$  се претставени во четири различни табели (Табела 6.2, Табела 6.3, Табела 6.4 и Табела 6.5). Во Табела 6.2 се прикажани перформансите на методите во однос на мерките базирани на примерок. Во Табела 6.3 се прикажани перформансите на методите во однос на мерките базирани на ознака, додека Табела 6.4 и Табела 6.5 ги прикажуваат перформансите во однос на мерките базирани на рангирање и времињата на обучување и тестирање, соодветно. Најдобрите резултати во табелите по евалуациска мерка и податочо множество се прикажани задебелено. Во Табела 6.1 се прикажани вредностите на прагот на одлука  $t$  за кои се прикажани резултатите на дво-нивовската архитектура, како и вредностите на  $a_{bmf}$  и  $r_{real}$ .

Во првата колона од четирите табели кои ги прикажуваат перформансите на споредуваните методи се означени евалуациските мерки. Во втората колона се прикажани имињата на споредуваните методи, а останатите колони ги прикажуваат резултатите добиени од експериментите, по една колона за секое од податочните множества.

Во Табела 6.6 се прикажани вредностите на  $p$  (ниво на значајност) според *Вилкоксон тестот* за секоја евалуациска мерка добиени со статистичка евалуација на перформансите на методите на дво-нивовската архитектура ( $TSVM$ ,  $TSCCM$  и  $TSPCCM$ ) во споредба со перформансите на  $CLR$  методот. Во првата колона се прикажани евалуациските мерки, додека во останатите колони се прикажани вредностите на

нивото на значајност  $p$  на статистичките резултати. Знакот  $>$  во третата, петтата и седмата колона покажува дека првиот метод е статистички подобар од вториот (прв метод *vs* втор метод, за секој од споредуваните парови), додека знакот  $<$  покажува дека вториот метод е статистички подобар од првиот. Вредностите на  $p$  помали од 0.05 покажуваат дека меѓу двете споредувани методи постои статистички значајна разлика и истите се прикажани задебелено.

**Табела 6.1** Вредности на параметрите  $a_{brmf}$ ,  $t$  и  $r_{real}$  за сите податочни множества за трите варијации на дво-нивовската архитектура

	T SVM	T SCCM	T SPCCM	T SVM	T SCCM	T SPCCM	T SVM	T SCCM	T SPCCM
	$a_{brmf}$			$t$			$r_{real}$		
<b>emotions</b>	3.658	3.143	3.143	0.20	0.25	0.25	0.32410	0.22451	0.22451
<b>scene</b>	1.394	1.394	1.394	0.20	0.20	0.20	0.01831	0.01831	0.01831
<b>yeast</b>	6.635	4.053	4.053	0.30	0.40	0.40	0.20543	0.06799	0.06799
<b>medical</b>	1.339	1.234	1.234	0.20	0.30	0.30	0.00023	0.00015	0.00015
<b>enron</b>	3.005	3.005	3.005	0.30	0.30	0.30	0.00219	0.00219	0.00219
<b>corel5k</b>	6.872	6.872	2.238	0.10	0.10	0.20	0.00029	0.00029	0.00002
<b>tmc2007</b>	2.351	2.351	2.502	0.30	0.30	0.20	0.00687	0.00687	0.00813
<b>mediamill</b>	4.510	4.510	3.454	0.20	0.20	0.30	0.00157	0.00157	0.00084
<b>bibtex</b>	1.499	1.949	1.949	0.30	0.20	0.20	0.00003	0.00007	0.00007

Од Табела 6.2, Табела 6.3 и Табела 6.4 јасно може да се забележи дека дво-нивовската архитектура ги надминува перформансите на останатите две методи *парови на ознаки* (*CLR* и *QWML*), во однос на скоро сите евалуациски мерки за сите податочни множества на кои овие методи се споредени. Предложената архитектура е апсолутен победник во оваа споредба. Таа има статистички подобри перформанси во споредба со *CLR* методот, а за евалуациските мерки *точност*, *odsub*,  $F_1$ , *класификациска точност*, *микро odsub*, *микро  $F_1$* , *макро odsub*, *макро  $F_1$*  и *една грешка* (Табела 6.6) *T SVM* и *T SCCM* покажуваат дури и статистички значајно подобри резултати од *CLR*. Единствено податочно множество за кое *CLR* методот покажа добри перформанси е *corel5k* податочното множество. Исто така, *CLR* покажа подобри перформанси во однос на помалите податочни множества (*emotions*, *scene* и *yeast*) наспроти оние кои ги постигна во однос на поголемите податочни множества (*tmc2007*, *mediamill* и *bibtex*). За поголемиот број од евалуациски мерки, *QWML* покажа слични перформанси со *CLR*. Разликата во перформанси меѓу овие две методи е најмногу изразена кај евалуациските мерки базирани на рангирање, што е резултат на помалиот број на евалуирани модели т.е., бројот на освоени гласови на ознаките и рангот на нерелевантните ознаки.

Табела 6.2 Перформанси на методите (*CLR*, *QWML*, *TSVM*, *TSCCM* и *TSPCCM*) во однос на евалуациските мерки базирани на примерок

		emotions	scene	yeast	medical	enron	corel5k	tmc2007	mediamill	libtex
<i>Hamming</i> загуба	CLR	0.257	0.080	<b>0.190</b>	0.017	0.048	0.012	0.014	0.043	<b>0.012</b>
	QWML	<b>0.254</b>	0.081	0.191	0.012	0.048	0.012	0.014	0.043	<b>0.012</b>
	TSVM	0.259	0.080	<b>0.190</b>	<b>0.011</b>	0.046	0.013	<b>0.013</b>	<b>0.032</b>	<b>0.012</b>
	TSCCM	0.259	0.080	0.192	<b>0.011</b>	0.046	<b>0.011</b>	<b>0.013</b>	<b>0.032</b>	<b>0.012</b>
	TSPCCM	0.260	<b>0.079</b>	0.192	<b>0.011</b>	<b>0.045</b>	<b>0.011</b>	<b>0.013</b>	<b>0.032</b>	<b>0.012</b>
точност	CLR	0.361	<b>0.686</b>	0.524	0.656	0.459	0.195	0.889	0.095	0.334
	QWML	<b>0.373</b>	0.683	0.523	0.658	0.388	0.195	0.889	0.095	0.338
	TSVM	0.369	0.680	0.526	<b>0.758</b>	0.478	0.200	0.892	0.407	<b>0.352</b>
	TSCCM	0.366	<b>0.686</b>	<b>0.530</b>	0.744	0.478	<b>0.202</b>	0.895	<b>0.411</b>	<b>0.352</b>
	TSPCCM	0.357	0.681	0.529	0.737	<b>0.481</b>	0.150	<b>0.897</b>	0.410	<b>0.352</b>
прецизност	CLR	0.538	<b>0.714</b>	0.719	0.695	0.650	<b>0.329</b>	0.937	0.201	0.488
	QWML	0.548	0.711	0.718	0.697	0.624	0.326	0.937	0.203	0.496
	TSVM	<b>0.550</b>	0.709	<b>0.721</b>	<b>0.792</b>	0.672	0.302	0.939	0.712	0.510
	TSCCM	0.540	<b>0.714</b>	0.711	0.784	<b>0.676</b>	<b>0.329</b>	0.943	0.721	<b>0.512</b>
	TSPCCM	0.533	0.710	0.711	0.780	0.675	0.289	<b>0.944</b>	<b>0.722</b>	<b>0.512</b>
одсиг	CLR	0.410	<b>0.712</b>	0.601	0.795	0.557	0.264	0.929	0.101	0.364
	QWML	<b>0.429</b>	0.709	0.600	0.801	0.453	0.264	0.929	0.101	0.366
	TSVM	0.412	0.704	0.602	<b>0.842</b>	0.568	<b>0.296</b>	0.930	0.464	<b>0.383</b>
	TSCCM	0.415	<b>0.712</b>	<b>0.616</b>	0.801	0.565	0.272	0.934	<b>0.468</b>	0.382
	TSPCCM	0.400	0.705	<b>0.616</b>	0.798	<b>0.572</b>	0.197	<b>0.936</b>	0.464	0.382
класификац. точност	CLR	0.144	<b>0.633</b>	<b>0.195</b>	0.486	0.117	0.010	0.767	0.044	0.183
	QWML	<b>0.149</b>	0.630	0.192	0.480	0.097	<b>0.012</b>	0.768	0.044	0.186
	TSVM	<b>0.149</b>	0.629	<b>0.195</b>	0.640	0.140	0.008	0.773	0.081	<b>0.198</b>
	TSCCM	<b>0.149</b>	0.632	0.193	<b>0.649</b>	<b>0.143</b>	0.011	<b>0.775</b>	<b>0.084</b>	0.197
	TSPCCM	0.144	0.629	0.192	0.634	0.138	0.006	<b>0.775</b>	0.083	0.197
<i>F1</i>	CLR	0.465	<b>0.713</b>	0.655	0.742	0.600	0.293	0.933	0.134	0.417
	QWML	<b>0.481</b>	0.710	0.654	0.745	0.525	0.292	0.933	0.135	0.421
	TSVM	0.471	0.706	0.656	<b>0.816</b>	0.616	<b>0.299</b>	0.935	0.562	<b>0.437</b>
	TSCCM	0.469	<b>0.713</b>	<b>0.660</b>	0.792	0.616	0.298	0.938	<b>0.567</b>	<b>0.437</b>
	TSPCCM	0.457	0.708	<b>0.660</b>	0.789	<b>0.620</b>	0.234	<b>0.940</b>	0.565	<b>0.437</b>

Табела 6.3 Перформанси на методите (*CLR*, *QWML*, *TSVM*, *TSCCM* и *TSPCCM*) во однос на евалуациските мерки базирани на ознака

			emotions	scene	yeast	medical	enron	corel5k	tmc2007	mediamill	libtex
макро прецизност	CLR	<b>0.685</b>	0.835	<b>0.729</b>	0.669	0.652	0.338	0.940	0.582	0.734	
	QWML	0.680	0.832	0.727	0.667	<b>0.687</b>	<b>0.339</b>	0.941	0.580	<b>0.736</b>	
	TSVM	0.673	0.839	<b>0.729</b>	0.778	0.672	0.312	0.944	0.717	<b>0.736</b>	
	TSCCM	0.673	0.838	0.717	<b>0.807</b>	0.677	0.320	<b>0.946</b>	0.725	0.733	
	TSPCCM	0.681	<b>0.844</b>	0.717	0.798	0.677	0.323	<b>0.946</b>	<b>0.727</b>	0.733	
макро прецизност	CLR	<b>0.677</b>	0.835	0.614	0.288	0.205	0.059	0.964	<b>0.140</b>	0.503	
	QWML	0.660	0.832	0.614	0.285	0.242	0.059	0.965	0.133	0.490	
	TSVM	0.602	0.839	<b>0.630</b>	<b>0.387</b>	<b>0.282</b>	<b>0.063</b>	0.970	0.115	0.530	
	TSCCM	0.614	0.838	0.582	0.379	0.267	<b>0.063</b>	<b>0.971</b>	0.131	<b>0.542</b>	
	TSPCCM	0.634	<b>0.843</b>	0.593	0.377	0.278	0.056	<b>0.971</b>	0.123	<b>0.542</b>	
макро одсув	CLR	0.409	0.695	0.595	0.782	0.532	0.258	0.920	0.066	0.322	
	QWML	<b>0.431</b>	0.692	0.595	0.787	0.438	0.258	0.920	0.066	0.328	
	TSVM	0.414	0.687	0.595	<b>0.830</b>	0.541	0.290	0.921	0.429	<b>0.341</b>	
	TSCCM	0.414	<b>0.695</b>	<b>0.610</b>	0.789	0.537	0.282	0.923	<b>0.433</b>	0.338	
	TSPCCM	0.396	0.689	0.609	0.785	<b>0.544</b>	0.197	<b>0.925</b>	0.429	0.338	
макро одлив	CLR	0.381	<b>0.704</b>	0.361	0.307	0.139	0.039	0.914	0.028	0.236	
	QWML	<b>0.398</b>	0.701	0.361	0.324	0.120	0.039	0.914	0.028	0.238	
	TSVM	0.383	0.697	0.359	<b>0.399</b>	0.148	<b>0.048</b>	0.917	0.058	<b>0.262</b>	
	TSCCM	0.380	<b>0.704</b>	<b>0.379</b>	0.387	<b>0.154</b>	<b>0.048</b>	0.922	<b>0.061</b>	0.261	
	TSPCCM	0.363	0.698	0.378	0.388	0.146	0.033	<b>0.924</b>	0.056	0.261	
макро $F_1$	CLR	0.512	0.758	0.655	0.721	0.585	0.293	0.930	0.118	0.448	
	QWML	<b>0.528</b>	0.756	0.654	0.722	0.535	0.293	0.930	0.119	0.454	
	TSVM	0.512	0.756	0.655	<b>0.803</b>	0.599	<b>0.301</b>	0.933	0.537	<b>0.466</b>	
	TSCCM	0.512	<b>0.760</b>	<b>0.659</b>	0.798	0.599	0.299	0.935	<b>0.542</b>	0.463	
	TSPCCM	0.501	0.758	<b>0.659</b>	0.791	<b>0.603</b>	0.245	<b>0.936</b>	0.540	0.463	
макро $F_1$	CLR	0.381	0.704	0.361	0.307	0.139	0.039	0.914	0.028	0.236	
	QWML	0.398	0.701	0.361	0.324	0.120	0.039	0.914	0.028	0.238	
	TSVM	<b>0.443</b>	0.759	0.388	<b>0.383</b>	0.169	<b>0.048</b>	0.943	0.064	<b>0.318</b>	
	TSCCM	0.442	<b>0.763</b>	<b>0.411</b>	0.370	<b>0.176</b>	<b>0.048</b>	0.945	<b>0.067</b>	<b>0.318</b>	
	TSPCCM	0.426	0.761	0.409	0.369	0.166	0.037	<b>0.947</b>	0.063	<b>0.318</b>	

Како што можеше да се очекува, *времињата на обучување* на споредените методи се многу слични. Имено, *времињата на обучување* на методите *CLR*, *QWML* и *TSVM* се исти, додека *времињата на обучување* на *TSCCM* и *TSPCCM* се незначително подолги (до, не повеќе од 5%). Подолгите *времиња на обучување* на овие методи се резултат на проширувањето на векторот на обележја на примероците во второто ниво од архитектурата. Од друга страна, *времињата на тестирање* драстично се разликуваат. Методите на дво-нивовската архитектура се статистички значајно побрзи во фазата на

тестирање во споредба со другите методи *парови на ознаки*. Времињата на тестирање на дво-нивовската архитектура се помали од времињата на тестирање на *CLR* во просек од 1.5 пати за најмалото податочно множество (*emotions*) до 57 пати за *corel5k* податочното множество. Во споредба со *QWML* методот, дво-нивовската архитектура е побрза во фазата на тестирање од 1.25 пати до 5.7 пати. Она што е особено интересно е фактот дека времињата на тестирање на *TSA* се многу слични со времињата на тестирање на *BR* методот (Табела А.18). Ова е резултат на малиот просечен број на евалуирани модели во второто ниво на архитектурата. Овој број може многу лесно да се пресмета за сите податочни множества со помош на параметарот  $a_{brmf}$  (Табела 6.1) со формулата  $a_{brmf}(a_{brmf} - 1) / 2$ . За разлика од *QWML*, помалиот број на евалуирани модели кај *TSVM*, *TSCCM* и *TSPCCM* не влијае на нивните перформанси во однос на останатите евалуациски мерки. На тој начин се избегнува шумот кој би можел да биде генериран во системот од моделите обучени со примероците кои се означени со нерелевантни ознаки за даден примерок. Притоа, шумот би можел драстично да биде зголемен имајќи го во предвид фактот дека просечниот број на релевантни ознаки по примерок е многу помал од просечниот број на нерелевантни ознаки за дадено податочно множество.

**Табела 6.4** Перформанси на методите (*CLR*, *QWML*, *TSVM*, *TSCCM* и *TSPCCM*) во однос на евалуациските мерки базирани на рангирање

		emotions	scene	yeast	medical	enron	corel5k	tmc2007	mediamill	bibtex
една грешка	<b>CLR</b>	0.391	0.190	<b>0.229</b>	0.168	0.231	<b>0.588</b>	0.033	0.586	0.388
	<b>QWML</b>	<b>0.337</b>	<b>0.181</b>	0.244	<b>0.117</b>	0.218	0.656	<b>0.018</b>	0.251	<b>0.341</b>
	<b>TSVM</b>	0.342	<b>0.181</b>	0.231	0.129	<b>0.212</b>	0.592	0.032	0.262	0.345
	<b>TSCCM</b>	<b>0.337</b>	0.183	0.240	0.120	0.216	0.590	0.019	<b>0.240</b>	<b>0.341</b>
	<b>TSPCCM</b>	<b>0.337</b>	<b>0.181</b>	0.244	<b>0.117</b>	0.218	0.656	<b>0.018</b>	0.251	<b>0.341</b>
офратеност	<b>CLR</b>	2.386	0.423	<b>6.286</b>	2.036	<b>11.763</b>	<b>91.506</b>	1.363	24.247	<b>18.540</b>
	<b>QWML</b>	2.807	0.631	8.659	1.832	22.746	206.880	2.796	28.982	57.343
	<b>TSVM</b>	2.356	<b>0.401</b>	6.335	1.544	12.563	105.376	1.317	20.532	20.963
	<b>TSCCM</b>	<b>2.351</b>	0.405	6.347	1.586	12.501	100.234	1.296	20.527	20.965
	<b>TSPCCM</b>	2.361	<b>0.401</b>	6.358	<b>1.532</b>	12.440	104.958	<b>1.291</b>	<b>20.470</b>	20.965
загуба при рангирање	<b>CLR</b>	0.264	0.065	<b>0.163</b>	0.028	<b>0.078</b>	<b>0.100</b>	0.005	0.092	<b>0.065</b>
	<b>QWML</b>	0.331	0.103	0.296	0.027	0.177	0.245	0.039	0.101	0.207
	<b>TSVM</b>	0.252	<b>0.060</b>	<b>0.163</b>	<b>0.020</b>	0.083	0.117	0.004	<b>0.061</b>	0.067
	<b>TSCCM</b>	<b>0.250</b>	0.061	0.166	0.021	0.083	0.116	<b>0.003</b>	<b>0.061</b>	0.068
	<b>TSPCCM</b>	0.253	<b>0.060</b>	0.166	<b>0.020</b>	0.083	0.116	<b>0.003</b>	<b>0.061</b>	0.068
просечна прецизност	<b>CLR</b>	0.718	0.886	0.768	0.864	0.699	0.352	0.972	0.450	0.579
	<b>QWML</b>	0.679	0.864	0.698	0.862	0.604	0.311	0.938	0.492	0.498
	<b>TSVM</b>	0.728	0.891	<b>0.769</b>	0.899	0.695	0.331	0.976	0.674	<b>0.598</b>
	<b>TSCCM</b>	<b>0.732</b>	0.891	0.765	<b>0.903</b>	0.698	<b>0.358</b>	0.984	<b>0.679</b>	<b>0.598</b>
	<b>TSPCCM</b>	0.731	<b>0.892</b>	0.764	<b>0.903</b>	<b>0.700</b>	0.307	<b>0.985</b>	0.677	<b>0.598</b>

**Табела 6.5** Перформанси на методите (*CLR*, *QWML*, *TSVM*, *TSCCM* и *TSPCCM*) во однос на времињата на обучување и тестирање мерени во секунди

		emotions	scene	yeast	medical	enron	corel5k	tmc2007	mediamill	bibtex
време на обучување	<b>CLR</b>	10.0	195.0	672.0	40.0	971.0	2388.0	52427.0	260156.0	13424.0
	<b>QWML</b>	10.0	195.0	672.0	40.0	971.0	2388.0	52427.0	260156.0	13424.0
	<b>TSVM</b>	10.0	195.0	672.0	40.0	971.0	2388.0	52427.0	260156.0	13424.0
	<b>TSCCM</b>	11.0	215.0	698.0	48.0	1038.0	2450.0	52688.0	261801.0	13982.0
	<b>TSPCCM</b>	11.0	208.0	685.0	44.0	1020.0	2430.0	52567.0	261132.0	13642.0
време на тестирање	<b>CLR</b>	3.0	87.0	153.0	90.0	634.0	2161.0	3282.0	76385.0	16733.0
	<b>QWML</b>	2.0	40.0	64.0	25.0	174.0	119.0	1543.0	20317.0	4710.0
	<b>TSVM</b>	1.6	<b>29.0</b>	42.0	<b>5.0</b>	<b>55.0</b>	<b>38.0</b>	<b>1078.0</b>	<b>7125.0</b>	<b>823.0</b>
	<b>TSCCM</b>	1.6	34.0	43.0	6.9	61.0	44.0	1101.0	7669.0	866.0
	<b>TSPCCM</b>	<b>1.5</b>	32.0	<b>40.0</b>	6.0	59.0	43.0	1132.0	7601.0	838.0

**Табела 6.6** Ниво на значајност  $p$  според *Викоксон тестот* за секоја евалуациска мерка во однос на перформансите на методите *CLR*, *TSVM*, *TSCCM* и *TSPCCM*. Знакот  $>$  покажува дека првиот метод е статистички подобар од вториот (прв метод vs втор метод), додека знакот  $<$  покажува дека вториот метод е статистички подобар од првиот. Вредностите на  $p$  помали од 0.05 покажуваат дека меѓу двата споредувани метода постои статистички значајна разлика

	<i>TSVM</i>		<i>TSCCM</i>		<i>TSPCCM</i>	
	vs		vs		vs	
	<i>CLR</i>		<i>CLR</i>		<i>CLR</i>	
<i>Хаминг загуба</i>	0.236	$>$	0.286	$>$	0.193	$>$
<i>точност</i>	<b>0.028</b>	$>$	<b>0.007</b>	$>$	0.155	$>$
<i>прецизност</i>	0.138	$>$	0.058	$>$	0.314	$>$
<i>odsiv</i>	<b>0.028</b>	$>$	<b>0.007</b>	$>$	0.343	$>$
<i>F<sub>1</sub></i>	<b>0.038</b>	$>$	<b>0.007</b>	$>$	0.286	$>$
<i>класификациска точност</i>	<b>0.038</b>	$>$	<b>0.038</b>	$>$	0.109	$>$
<i>микро прецизност</i>	0.213	$>$	0.441	$>$	0.314	$>$
<i>микро odsiv</i>	<b>0.028</b>	$>$	<b>0.007</b>	$>$	0.441	$>$
<i>микро F<sub>1</sub></i>	<b>0.033</b>	$>$	<b>0.007</b>	$>$	0.173	$>$
<i>макро прецизност</i>	0.213	$>$	0.514	$>$	0.514	$>$
<i>макро odsiv</i>	<b>0.044</b>	$>$	<b>0.015</b>	$>$	0.109	$>$
<i>макро F<sub>1</sub></i>	<b>0.038</b>	$>$	<b>0.013</b>	$>$	0.097	$>$
<i>една грешка</i>	<b>0.038</b>	$>$	<b>0.028</b>	$>$	0.173	$>$
<i>опфатеност</i>	0.678	$>$	0.767	$>$	0.722	$>$
<i>загуба при рангирање</i>	0.406	$>$	0.593	$>$	0.553	$>$
<i>просечна прецизност</i>	0.125	$>$	<b>0.021</b>	$>$	0.138	$>$
<i>време на обучување</i>	0.767	$>$	<b>0.007</b>	$<$	<b>0.007</b>	$<$
<i>време на тестирање</i>	<b>0.007</b>	$>$	<b>0.007</b>	$>$	<b>0.007</b>	$>$

Ако се споредат вредностите на параметарот  $a_{brmf}$  и кардиналноста на ознаките  $l_c$  за соодветните податочни множества, може да се забележи дека тие се доста слични. Вредностите на параметарот  $a_{brmf}$  се малку поголеми, со цел да се даде поголема шанса на ознаките за кои предвидувањата на моделите на *бинарна релевантност* не биле сигурни дали истите треба да бидат прогласени за релевантни или нерелевантни. Со пропуштањето на примероците на второто ниво од архитектурата, одлуката за релевантност на овие ознаки се препушта да биде донесена од соодветните модели на *парови на ознаки*.

Перформансите на двете верижни методи (*TSCCM* и *TSPCCM*) се доста слични. *TSCCM* просечно покажа малку подобри резултати во однос на евалуациските мерки базирани на примерок, ознака и рангирање, но и малку подолги времиња на обучување и тестирање. И двете методи покажаа подобри резултати од дво-нивовскиот метод со гласање.

Зависноста на перформансите на дво-нивовската архитектура од прагот на одлука  $t$  е прикажана на Слика Б.1 до Слика Б.27. На сликите се прикажани перформансите на трите методи *T SVM*, *TSCCM* и *TSPCCM* во однос на евалуациските мерки базирани на примерок, ознака и рангирање за вредности на прагот на одлука  $t$  од 0.0 до 1.0 со чекор 0.1. Како што може да се забележи, на сликите не е прикажана у оската, затоа што перформансите во однос на сите евалуациски мерки се скалирани во опсегот од 0 до 1 (вредностите на у оската би немале никакво значење). Целта на овие графици е да ја прикажат динамиката со која се менуваат перформансите на дво-нивовската архитектура во зависност од прагот на одлука. Ако се анализира оваа зависност, лесно може да се забележи дека за вредности на прагот на одлука помали од 0.5 резултатите во однос на перформансите евалуациски мерки се многу слични, а времињата на тестирање многу брзо се намалуваат и се скоро константни за вредности на прагот на одлука поголеми од 0.4. Најголемо намалување на времињата на тестирање се забележува за вредности на прагот на одлука од 0.0 до 0.1. Како што се очекуваше, рангирањето на ознаките најмалку зависи од прагот на одлука, а тоа е резултат на рангирањето на сите (прогласени) нерелевантни ознаки според предикциите на соодветните модели на *бинарна релевантност* од првото ниво на архитектурата.

## 6.4 Заклучок

Во оваа глава беше предложена *дво-нивовска ефикасна парови на ознаки архитектура* за повеќезначна класификација и рангирање. Претставени беа три методи за градење на оваа архитектура: Дво-нивовскиот метод со гласање, дво-нивовскиот верижен метод и дво-нивовскиот скратен верижен метод. Архитектурата е организирана во две нивоа: Во првото ниво се сместени  $Q$  модели на *бинарна релевантност*, додека во второто ниво се сместени  $Q(Q - 1)/2$  модели на *парови на ознаки*. Со воведување на прагот на одлука со кој динамички се намалува бројот на евалуирани модели од второто ниво на архитектурата во фазата на тестирање, беше постигнато значајно намалување на пресметковната комплексност, без притоа негативно да се влијае на резултатите добиени во однос на перформансите евалуациски мерки. Верижната структура која притоа беше воведена кај *TSCCM* и *TSPCCM*, уште повеќе ги подобри перформансите на дво-нивовската архитектура во однос на евалуациските мерки базирани на примерок, ознака и рангирање.

Во експериментална евалуација беше направена споредба на предложената архитектура и нејзините три варијации (дво-нивовскиот метод со гласање, дво-нивовскиот верижен метод и дво-нивовскиот скратен верижен метод) со двата метода парови на ознаки *CLR* и *QWML*. Методите ги споредивме на 9 повеќезначни класификациски проблеми од 5 апликациски области (класификација на текст, класификација на слики предикција на функција на гени, класификација на музика и класификација на видео) во однос на 18 различни евалуациски мерки (6 мерки базирани на ознака, 6 мерки базирани на примерок, 4 мерки базирани на рангирање и 2 мерки за ефикасноста на методите). За да ја процениме статистичката значајност на разликата на перформансите на споредуваните методи го користевме *Вилкоксон тестот*.

Врз основа на резултатите добиени од експериментите и статистичката евалуација може да се заклучи дека трите методи значително го намалуваат времето на тестирање на методите *парови на ознаки*. Во просек, намалувањето на времето на тестирање е од 1.5 до 57 пати во споредба со *CLR* методот и 1.25 до 5.7 пати во споредба со *QWML* методот. *Вилкоксон тестот* покажа дека оваа разлика е и статистички значајна.

Во однос на перформансите евалуациски мерки предложената архитектура има статистички подобри перформанси во споредба со *CLR* методот, а за евалуациските мерки *точност*, *odsub*,  $F_1$ , *класификациска точност*, *микро odsub*, *микро  $F_1$* , *макро odsub*, *макро  $F_1$*  и *една грешка*, *T SVM* и *TSCCM* покажуваат дури и статистички значајно подобри резултати од *CLR*.

Со менување на вредноста на прагот на одлука оваа архитектура дава можност за балансирање на перформансите на предикција и времињата на тестирање.

Во идните истражувања ќе бидат разгледани различни алгоритми за автоматско определување на прагот на одлука, а исто така ќе биде анализирана можноста за воведување на различен праг на одлука за секој од моделите на *бинарна релевантност* од првото ниво на архитектурата. Воведувањето на различни прагови на одлука веруваме дека значително може да ги подобри перформансите на оваа архитектура, поради фактот дека прагот на одлука значајно влијае на нејзините перформанси и пресметковна комплексност.

## Глава 7

# Хибридна архитектура за повеќезначна класификација и рангирање

Во оваа глава ќе биде претставена хибридна архитектура базирана на дрва за одлучување кои користат локални машини со носечки вектори за повеќезначна класификација и рангирање. Основната идеја на оваа архитектура е да ја намали пресметковната комплексност на методите кои го трансформираат проблемот и да ги задржи или подобри перформансите во однос на предвидувањата. Имено, користејќи го методот „раздели па владеј“, прво се трансформира глобалниот повеќезначен класификациски проблем во повеќе помали локални проблеми, а потоа со употреба на моќни локални класификатори се решаваат истите поефикасно и со поголема точност. Притоа за делење на проблемот се користат дрва за одлучување кои се карактеристични по својата висока ефикасност, а за решавање на локалните класификациски проблеми се користат локални машини со носечки вектори во *бинарна релевантност* архитектура. Имајќи ја во предвид брзината на дрвата за одлучување и прецизноста на методот *бинарна релевантност* се очекува неколкукратно зголемување на брзината на обучување и предвидување и подобрување на перформансите во однос на прецизноста на предвидување во однос на постоечките техники за повеќезначна класификација и рангирање.

Во продолжение прво е претставен методот за градење на хибридната архитектура и нејзиниот дизајн, а потоа се претставени и две дополнителни методи за градење на две нејзини модификации. Модификациите се однесуваат на начинот на поткастрување на дрвата за одлучување и употребата на машините со носечки вектори. На крај, направена е анализа на пресметковната комплексност на хибридната архитектура во споредба со пресметковната комплексност на глобалниот метод *бинарна релевантност*.

Оваа глава завршува со анализа и заклучоци донесени врз основа на резултатите добиени од експерименталната евалуација на оваа архитектура. Во експерименталната евалуација се опфатени сите евалуациски мерки претставени во Глава 3, како и времињата на тренирање и тестирање.

### 7.1 Комбинирање на дрва за одлучување и машини со носечки вектори

Во литературата досега се предложени неколку методи кои ги комбинираат дрвата за одлучување и машините со носечки вектори за повеќекласна класификација. Конкретно, Kumar et al. [78] предлага метод во кој ги комбинира овие два типа на класификатори за дво-класна класификација со цел да се подобрат перформансите на

машините со носечки вектори во однос на времињата на тестирање (пресметковната комплексност е главен недостаток на машините со носечки вектори), а притоа да се задржат нивните перформанси на предвидување. Имено, кај овој метод прво се обучува *SVM* класификатор со сите примероци од множеството на обучување. По неговото обучување сите примероци кои се блиску до маргината (под некој праг) на *SVM* класификаторот се селектираат и се означуваат со нова „трета“ класа. Ново добиениот трокласен класификациски проблем се решава со дрво за одлучување. Сите листови кои при обучувањето на дрвото за одлучување биле означени со „третата“ класа се заменуваат со *SVM* класификаторот обучен на сите примероци од податочното множество за обучување. Во овие листови, конечните одлуки за класите во кои припаѓаат тестирачките примероци ги носи *SVM* класификаторот.

Останатите методи како што се [79] [80] [81], користејќи ја структурата на дрвата за одлучување ја зголемуваат ефикасноста и прецизноста на предвидување на *SVM* базираните методи за повеќекласна класификација. Овие методи, прво го градат дрвото за одлучување, а потоа во неговите јазли сместуваат *SVM* класификатори за решавање на локалните класификациски проблеми дефинирани во соодветните јазли. *Бустинг* (анг. *Boosting*) ансамбл од комбинациите на дрва за одлучување и машини со носечки вектори се предложени од Ting и Zhu [82]. Gama [83] предлага архитектура од дрва за одлучување и бинарни класификатори која со помош на генерален пристап ги решава проблемите на повеќекласна класификација и регресија [84]. Тој предлага функционални дрва (анг. *functional trees*) во чии јазли се обучуваат класификатори или регресори (во зависност од проблемот). Обучениот класификатор или регресор се тестира со примероците од податочното множество на обучување кои се дел од соодветниот јазел. Добиените предикции се додаваат како нови обележја на соодветните примероци и истите се дел од процесот на делење на примероците во јазлите на дрвото на обучување. Имено, добиените предикции исто како и оригиналните обележја се вклучени во процесот на пресметување на ентропијата на множествата на примероци кои се формирани во однос на оригиналните и ново генерираните обележја. Сите овие методи покажуваат дека можат успешно да бидат комбинирани дрвата за одлучување и стабилните класификатори за повеќезначна класификација. Меѓутоа, досега не е објавен метод (не му е познат на авторот) кој ги комбинира дрвата за одлучување и машините со носечки вектори како стабилни класификатори за повеќезначна класификација и рангирање. Дополнително сеуште не е покажано како оваа комбинација може да влијае на перформансите на основните класификатори во однос на ефикасноста и перформансите на предвидување.

## **7.2 Ефикасна хибридна архитектура базирана на дрва за одлучување кои користат локални машини со носечки вектори**

За да се добие поефективна и пресметковно поефикасна предикција, еден пристап е прво глобалниот класификациски проблем да се подели на помали локални класификациски проблеми, а потоа „новите“ локални класификациски проблеми да се решат со помош на одделни локални класификатори (по еден класификатор за секој локален класификациски проблем). Во фазата на предвидување тоа значи дека потребно е прво да се определи на кој локален проблем припаѓа примерокот, а потоа со локалниот класификатор обучен со примероците кои припаѓаат на локалното податочно множество да се определи на кој класа примерокот припаѓа, односно со кои

ознаки примерокот треба да биде означен. На овој начин, прво се обидуваме да ја најдеме групата на која припаѓа примерокот, а потоа само со примероците од таа група да го означиме примерокот. Притоа претпоставуваме дека доколку примерокот е придружен на вистинската група, примероците на таа група најдобро (најбрзо и најефективно) можат да одлучат за ознаките со кои примерокот би требало да биде означен. За таа цел, предлагаме нова хибридна архитектура кој воведува локални модели базирани на машините со носечки вектори за ефикасно и ефективно решавање на повеќезначни класификациски проблеми.

Предложената архитектура ги комбинира *ML-C4.5* методот за дефинирање на локалните класификациски проблеми и *BR* методот за решавање на истите. *BR* методот ги користи машините со носечки вектори како основни класификатори за решавање на локалните бинарни класификациски проблеми. Главната идеја е да се искористат предностите на двете методи со цел да се подобрат перформансите на предвидување и времињата на обучување и тестирање. Предложената архитектурата во понатамошниот тест е референцирана како хибридна архитектура базирана на дрва за одлучување кои користат локални машини со носечки вектори за ефикасна повеќезначна класификација и рангирање (*ML-SVMDT*).

*BR* методот е често споменуван во литературата, но константно критикуван поради игнорирањето на зависностите кои постојат меѓу ознаките и примероците за обучување. Најголемиот проблем е тоа што *BR* методот за еден примерок може позитивно да предвиди две или повеќе ознаки кои во пракса неможат да бидат заеднички ознаки за еден примерок. За да се надмине оваа неконзистентност, се користи *ML-C4.5* методот во кој примероците се групираат во групи во кои некорелираноста на ознаките е елиминирана. Поделбата на примероците во секој од јазлите на дрвото се прави во согласност со некорелираноста на ознаките.

Имплементацијата на архитектурата се состои од три последователни фази:

1. Во првата фаза се гради дрвото за одлучување (Слика 7.1).
2. Во втората фаза се поткаструва дрвото за одлучување (Слика 7.2).
3. Во третата фаза прво се обучуваат локалните *SVM* модели, а потоа се интегрираат во дрвото за одлучување (Слика 7.3).

Во продолжение прво е воведена нотацијата што е користена во алгоритмите прикажани на Слика 7.1, Слика 7.2 и Слика 7.3, а потоа анализирани се трите фази поодделно. Прво, податочното множество со примероците за обучување се дели на множество на примероци за градење на дрвото за одлучување ( $E^{train}$ ) и множество на примероци за валидација ( $E^{val}$ ). Со  $f$  е означено едно обележје на податочното множество, а  $g$  е нормализираното количество на информација, пресметано како разлика меѓу ентропијата на дадено податочное множество  $E$  и ентропиите на податочните множества добиени со поделба на  $E$  според вредноста на обележјето  $f$ . Со  $\mathcal{P}$  е означена партицијата која се добива како резултат на поделбата на податочното множество според вредностите на обележјето  $f$ .  $g^*$  е максималното количество на информација кое се добива со поделба на податочното множество со обележјето  $f^*$ , а  $\mathcal{P}^*$  е партицијата која се добива со оваа поделба.

Процедурата на градење на дрвото за одлучување е дадена на Слика 7.1. Оваа процедура како влезен параметар го прима множеството на примероци за обучување

( $E^{train}$ ), а како излез го враќа изграденото дрво за одлучување. Процесот на градење на дрвото за одлучување започнува од коренот на дрвото со избирање на едно обележје ( $f$ ) кое најефективно го дели податочното множество на партиции ( $\mathcal{P}$ ). Изборот на обележјето  $f$  кое најефективно го дели податочното множество се прави со помош на процедурата *BestFeature* (ред 1 од *ML-SVMDT* процедурата, Слика 7.1). Критериумот при делење на податочното множество е нормализираното количество на информација - разликата во ентропијата  $g$  (ред 4 од *BestFeature* процедурата, Слика 7.1). Обележјето кое ја максимизира ентропијата се избира за да се направи поделбата на податочното множество а истото во алгоритмот е прикажано во редот 6 од процедурата *BestFeature* на Слика 7.1. Градењето на дрвото продолжува рекурзивно во секој јазел, се додека сите примероци во јазелот не се означени со една единствена ознака или ниту едно од обележјата повеќе не го зголемува количеството на информација. Исто така, градењето на дрвото во конкретен јазел престанува, ако во тој јазел биде надминат некој предефиниран минимален број на примероци. Тоа значи дека, ако поделбата на податочното множество во јазелот не може да се направи врз основа на ниту едно од обележјата или бројот на примероци во тој јазел е помал од минималниот број на примероци, градењето на дрвото престанува во тој јазел и истиот се прогласува за лист. Функцијата *Acceptable*( $f, \mathcal{P}$ ) (ред 5 од *BestFeature* процедурата, Слика 7.1) одредува дали поделбата на податочното множество за конкретна вредност на обележјето  $f$  е прифатлива или не.

<b>procedure</b> <i>ML-SVMDT</i> ( $E^{train}$ )	<b>procedure</b> <i>BestFeature</i> ( $E$ )
<b>returns</b> tree	
1: $(f^*, g^*, \mathcal{P}^*) = \text{BestFeature}(E^{train})$	1: $(f^*, g^*, \mathcal{P}^*) = (\text{none}, \theta, \emptyset)$
2: <b>if</b> $f^* \neq \text{none}$ <b>then</b>	2: <b>for each</b> feature $f$ <b>do</b>
3: <b>for each</b> $E_k^{train} \in \mathcal{P}^*$ <b>do</b>	3: $\mathcal{P} =$ partition induced by $f$ on $E$
4: $tree_k = \text{ML-SVMDT}(E_k^{train})$	4: $g = \text{entropy}(E) - \sum_{E_k \in \mathcal{P}} \frac{ E_k }{ E } \text{entropy}(E_k)$
5: <b>return</b> $\text{node}(f^*, E_k^{train}, \cup_k \{tree_k\})$	5: <b>if</b> $(g > g^*) \wedge \text{Acceptable}(f, \mathcal{P})$ <b>then</b>
6: <b>else</b>	6: $(f^*, g^*, \mathcal{P}^*) = (f, g, \mathcal{P})$
7: <b>return</b> $\text{leaf}(E^{train})$	7: <b>return</b> $(f^*, g^*, \mathcal{P}^*)$

Слика 7.1 Алгоритам за градење на *ML-SVMDT*

По фазата на градење, дрвото се поткаструва. Генералниот алгоритам за поткастрување на дрвото е претставен на Слика 7.2. За поткастрување на дрвото се користи *bottom up* пристап (од листовите кон коренот) со *post order* изминување (редови 2 и 3 од *PruneByValidation* процедурата, Слика 7.2). Поткаструвањето на дрвото се прави со податочно множество за валидација  $E^{val}$  т.е. со примероците од оригиналното податочното множество за обучување кои не се користат за градење на дрвото за одлучување. Сите примероци од ова податочно множество се проследуваат на дрвото за одлучување, истото го изминуваат и формираат локални податочни множества во секој од листовите. При поткаструвањето со валидација, за секој јазел се проценуваат две мерки: *грешка на лист* (анг. *leaf error*) и *грешка на дрво* (анг. *tree error*). *Грешка на лист* ја претставува грешката на јазелот ако тој биде прогласен за лист, додека *грешка на дрво* ја претставува грешката на јазелот како дел од дрвото (сума на грешките на сите подрва на конкретниот јазел). Ако *грешката на дрво* е поголема или еднаква на *грешката на лист* (ред 4 од *PruneByValidation* процедурата,

Слика 7.2), тогаш јазелот се прогласува за лист, а неговите поддрва се отфрлаат (поткаструваат).

---

```
procedure PruneByValidation(tree,  $E^{val}$ )
1: if tree is not leaf then
2:   for each descendent k do
3:     PruneByValidation(treek,  $E_k^{val}$ )
4:   if LeafError(tree,  $E^{val}$ ) ≤ TreeError(tree,  $E^{val}$ ) then
5:     tree = leaf(tree. $E^{train}$ )
```

---

Слика 7.2 Процедура за поткастрување со валидација на дрвото за одлучување по неговата изградба

---

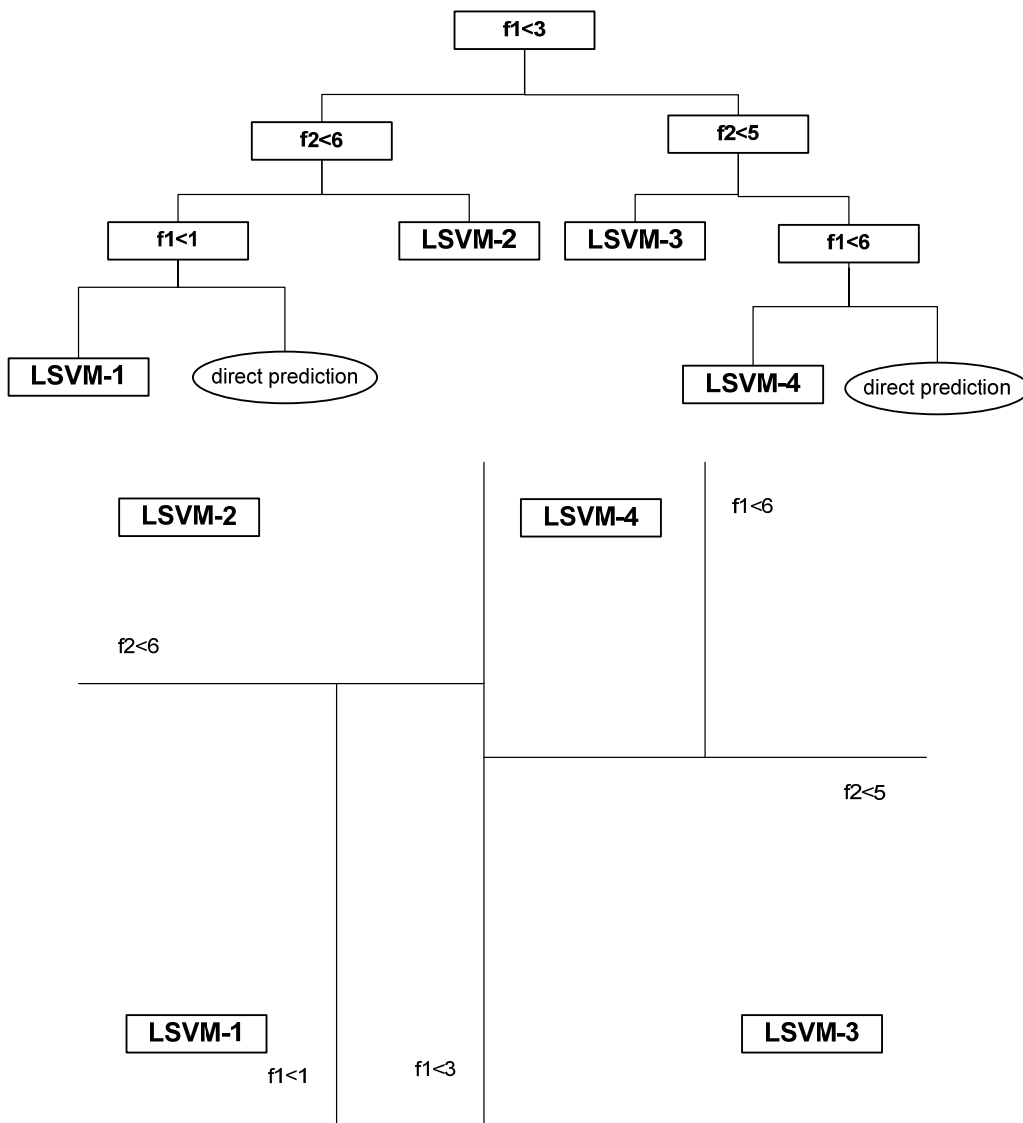
```
procedure BuildLocalModels(tree,  $E^{val}$ )
1: if tree is not leaf then
2:   for each descendent k do
3:     BuildLocalModels(treek,  $E_k^{val}$ )
4: else
5:   localSVM = TrainLocalModel(tree. $E^{train}$ )
6:   if LocalModelError(localSVM,  $E^{val}$ ) < LeafError(tree,  $E^{val}$ ) then
7:      $E$  = tree. $E^{train}$  +  $E^{val}$ 
8:     localSVM = TrainLocalModel( $E$ )
9:     tree = ReplaceNode(localSVM)
```

---

Слика 7.3 Процедура за обучување на локалните SVM модели

По поткаструвањето на дрвото, „новите“ класификациски проблеми дефинирани на примероците во листовите од дрвото се решаваат со прецизни локални класификатори (модел). Тоа значи, дека во секој лист од дрвото се обучува по еден локален модел со примероците од соодветниот лист со помош на *BR* методот (ред 3 од *BuildLocalModels* процедурата, Слика 7.3). Притоа, како основни класификатори во *BR* методот се користат машините со носечки вектори. По обучувањето на секој локален модел со примероците од податочното множество  $E^{train}$  кои припаѓаат на листот каде се наоѓа соодветниот локален модел, со помош на примероците од податочното множество за валидација  $E^{val}$  кои се дел од истиот лист се проценува грешката на локалниот модел (ред 6 од *BuildLocalModels* процедурата, Слика 7.3). Ако грешката на локалниот модел (анг. *local model error*) е помала од грешката на листот, прво се формира заедничко множество на примероци составено од примероците на множествата  $E^{train}$  и  $E^{val}$  во конкретниот лист (ред 7 од *BuildLocalModels* процедурата, Слика 7.3), а потоа листот се заменува со локален модел обучен со тие примероци (редови 8 и 9 од *BuildLocalModels* процедурата, Слика 7.3). Последниот чекор, обучувањето на локалните модели со примероците од множествата за градење на дрвото за одлучување и валидација се прави со цел да се подобрат перформансите на предвидување на локалните модели, поради фактот дека податочното множество за валидација е дел од оригиналното множество за обучување. Перформансите на предвидување на локалните модели се очекува да бидат подобрани, како резултат на нивното обучување со податочно множество кое е проширено со примероци кои имаат слични карактеристики на оригиналните примероци за обучување.

Со употребата на локалните модели, само во листовите каде тие покажуваат подобри перформанси од листот, очекуваме да се подобри пресметковната ефикасност на предложената архитектура во целина особено во фазата на тестирање. Оваа претпоставка се темели на фактот што *SVM* моделите имаат помала пресметковна ефикасност во споредба со дрвата за одлучување. Во наредното поглавје е направена математичка анализа за пресметковната ефикасност на предложената архитектура.



Слика 7.4 *ML-SVM*DT го дели оригиналното податочно множество на подмножества и обучува локален *SVM* модел за некои партиции

Најзначајниот момент во процесите на поткастрување на дрвото за одлучување и интеграцијата на локалните модели во неговите листови е проценката за грешката (евалуација на грешката). Имено, и во двата случаи грешката зависи од функцијата на загуба која се користи. Притоа, имајќи го во предвид бројот на евалуациски мерки кои се користат за повеќезначна класификација и рангирање, може да се употребат

различни функции на загуба: *Хаминг загуба*, *загуба при рангирање*, *точност*, *опфатеност*, итн., зависно од проблемот кој треба да се реши.

Слика 7.4 прикажува пример за градење на *ML-SVMDT*. Во примерот дрвото за одлучување се гради во согласност со вредностите на обележјата **f1** и **f2**. По поделбата на примероците според обележјата **f1** ( $f1 < 3$ ) во коренот од дрвото и **f2** ( $f2 < 6$ ) во првото ниво од левото поддрво, во десното поддрво на второто ниво се обучува локалниот *SVM* модел **LSVM-2**. Во левото поддрво на третото ниво се обучува локалниот модел **LSVM-1**, а во десното поддрво предвидувањето се прави врз основа на примероците и ознаките кои се дел од тој лист. Пристапот за градење на десното поддрво е ист како и оној во левото поддрво.

Тестирањето на примероците започнува од коренот на дрвото. Дрвото за одлучување го проследува секој примерок до еден од неговите листови. Ако листот содржи локален *BR* модел, тогаш *BR* моделот ја носи финалната одлука за означувањето на примерокот. Ако листот не содржи локален модел, тогаш на примерокот се придружуваат ознаките со кои се означени повеќе од 50 % од примероците за обучување кои припаѓаат на тој лист.

### 7.3 Хибридни архитектури базирани на поткастриени дрва за одлучување кои користат локални машини со носечки вектори

За да ја истражиме уште подлабоко врската меѓу дрвата за одлучување и машините со носечки вектори предложени се две модификации на хибридната архитектура. Во првата модификација не се користи поткастрување со валидација на изграденото дрво за одлучување. Во оваа модифицирана архитектура поткаструвањето се прави во фазата на градење на дрвото (анг. *pre-pruning*) и како критериум се користи минималниот број на примероци во јазлите. Имено, ако во конкретен јазел има помал број на примероци од претходно зададен минимален број јазелот се прогласува за лист. За разлика од *ML-SVMDT*, оваа архитектура сите листови ги заменува со локални *BR* модели, кои како основни класификатори ги користат машините со носечки вектори. Со воведување на параметарот за минимален број на примероци се контролира длабочината на дрвото за одлучување и пресметковната комплексност на локалните модели. Оваа модификација на хибридната архитектура е означена како *ML-SVMDT<sub>pre</sub>*.

Втората модификација користи *издигнување на поддрво* (анг. *subtree raising*) [71] како метод за поткастрување по изградбата на дрвото со *фактор на доверба*, наместо поткастрување со валидација. Оваа модификација исто како и *ML-SVMDT<sub>pre</sub>* ги користи сите примероци од податочното множество за обучување при градење на дрвото. По изградбата на дрвото истото се поткаструва при што сите листови се заменуваат со локални *BR* модели. Оваа модификација на хибридната архитектура во продолжение е означена како *ML-SVMDT<sub>post</sub>*.

### 7.4 Пресметковна комплексност на хибридните архитектури базирани на дрвата за одлучување

Во ова поглавје е анализирана пресметковната комплексност на хибридните архитектури базирани на дрвата за одлучување кои користат машини со носечки

вектори во споредба со пресметковната комплексност на глобалниот  $BR$  метод. Бидејќи во листовите од дрвото за одлучување се користат исто така  $BR$  модели, оваа споредба ја правиме со цел да укажеме на разликата која постои меѓу пресметковните комплексности на глобалниот  $BR$  модел обучен со сите примероци на оригиналното податочное множество и локалните  $BR$  модели обучени во листовите од дрвото за одлучување. Прво воведена е нотацијата која се користи, а потоа поодделно се анализирани пресметковните комплексности на фазите на обучување и тестирање.

Со  $V$  е означен бројот на листови во дрвото за одлучување.  $D$  е димензијата на влезните податоци т.е., бројот на обележја на примероците во податочното множество, а  $M$  е бројот на континуални обележја. Со  $N$  е означен вкупниот број на примероци, додека со  $N^i$  е означен бројот на примероци во  $i$ -тиот лист од дрвото за одлучување. Со  $N_{sv}$  е означен бројот на носечки вектори во глобалниот  $BR$  модел (моделот обучен на целото податочное множество за обучување), а  $N_{sv}^i$  е бројот на носечки вектори во локалните  $BR$  модели во листовите на дрвото.

#### 7.4.1 Пресметковна комплексност во фазата на обучување

Како резултат на поделбата на оригиналниот класификациски проблем на помали, локални класификациски проблеми, може да се очекува дека пресметковната комплексност на  $BR$  методот ќе биде намалена во однос на предложената архитектура. Локалните модели во листовите од дрвото, за обучување користат подмножества на примероци од оригиналното податочное множество. Тие се обучуваат побрзо од глобалниот  $BR$  модел, генерираат помал број на носечки вектори, што на крајот резултира и со побрза предикција.

Пресметковната комплексност на глобалниот  $BR$  модел според Burges [85] зависи од примероците на податочното множество, димензијата на влезните податоци, како и распределбата и бројот на носечките вектори определени во процесот на обучување. Кога поголемиот број на носечки вектори не се наоѓаат блиску до хипер-рамнината и  $N_{sv}/N \ll 1$  пресметковната комплексност е  $O(N_{sv}^3 + N_{sv}^2 N + DN_{sv}N)$ . Ако  $N_{sv}/N \approx 1$ , тогаш пресметковната комплексност е претставена со  $O(N_{sv}^3 + N_{sv}N + DN_{sv}N)$ . Кога поголемиот број на носечки вектори се наоѓаат блиску до хипер-рамнината и  $N_{sv}/N \ll 1$ , пресметковната комплексност се сведува на  $O(N_{sv}^2 + DN_{sv}N)$ . Конечно, ако  $N_{sv}/N \approx 1$  и поголемиот број на носечки вектори се наоѓаат блиску до хипер-рамнината, тогаш пресметковната комплексност е дадена со  $O(DN^2)$ .

Времето на обучување на хибридна архитектура може да се претстави како сума од времето на обучување на дрвото за одлучување и времињата на обучување на локалните  $BR$  модели. Времето на обучување на дрвото за одлучување е сума на времињата на трите фази кои се извршуваат во секој од јазлите на дрвото. Овие три фази се: подредување на вредностите на нумеричките дескриптивни обележја, пребарување на најдобрата поделба на примероците во јазлите и нивната поделба. Сортирањето на едно дескриптивно обележје во еден јазел чини  $O(N \log N)$ , додека сортирањето на сите дескриптивни обележја е  $O(MN \log N)$ . Пребарувањето на најдобрата поделба (издвојувањето на секое обележје и пресметувањето на целата потребна статистика за количеството на информација) е од ред на големина  $O(MN)$ . Поделбата на примероците во секој од јазлите чини уште  $O(N)$ . Тоа значи дека конструирањето на еден јазел чини  $O(MN \log N) + O(MN) + O(N)$ . Ако претпоставиме

дека дрвото е балансирано и пополнето, вкупната пресметковна комплексност на градење на дрвото може да се претстави како

$$O_{ML-C4.5} = O(MN \log^2 N) + O(MN \log N) + O(N \log N) \quad (7.1)$$

Горната граница е определена со првиот израз, т.е., пресметковната комплексност на дрвото за одлучување е од ред на големина  $O(MN \log^2 N)$ .

Пресметковната комплексност на локалните модели се пресметува на сличен начин како и на глобалниот  $BR$  модел. За  $i$ -тиот локален модел, ако поголемиот број на носечки вектори не се наоѓаат блиску до хипер-рамнината и  $N_{sv}^i/N^i \ll 1$  пресметковната комплексност е  $O(N_{sv}^{i3} + N_{sv}^{i2}N^i + DN_{sv}^iN^i)$ . Ако  $N_{sv}^i/N^i \approx 1$ , тогаш пресметковната комплексност е претставена со  $O(N_{sv}^{i3} + N_{sv}^iN^i + DN_{sv}^iN^i)$ . Кога поголемиот број на носечки вектори се наоѓаат блиску до хипер-рамнината и  $N_{sv}^i/N^i \ll 1$ , пресметковната комплексност е  $O(N_{sv}^{i2} + DN_{sv}^iN^i)$ . Конечно, ако  $N_{sv}^i/N^i \approx 1$  и поголемиот број на носечки вектори се наоѓаат блиску до хипер-рамнината, тогаш пресметковната комплексност е дадена со  $O(DN^{i2})$ . Различни локални модели имаат различна пресметковна комплексност и таа зависи од локалните класификациски проблеми кои треба да бидат решени. Вкупната пресметковна комплексност е од ист ред или повисока од пресметковната комплексност на дрвото за одлучување.

Во најлош случај пресметковната комплексност на хибридна архитектура е од ист ред со пресметковната комплексност на  $BR$  методот. Но, реално е да се очекува таа да биде пониска за ред на големина на вкупниот број на листови во дрвото, како резултат на помалата комплексност на класификациските проблеми во секој од листовите.

#### 7.4.2 Пресметковна комплексност во фазата на тестирање

Времето на тестирање на секој примерок може да се претстави како сума од времето на изминување на дрвото од коренот до еден од листовите и времето потребно за донесување на одлуката од страна на локалниот модел во тој лист, ако локален модел е изграден во тој лист.

Пресметковната комплексност на  $BR$  методот за секој примерок во фазата за тестирање според Burges [85] се претставува на следниот начин:

$$O_{BR} = O(TN_{sv}) \quad (7.2)$$

каде  $T$  е бројот на операции потребни да се евалуира кернелот. За Гаусов кернел  $T$  е  $O(D)$ , односно ред на големина на бројот на обележја на податочното множество.

Како што кажавме претходно, пресметковната комплексност на  $ML-SVMDT$  архитектурата во фазата на тестирање, генерално може да се претстави како сума на комплексноста на  $ML-C4.5$  и пресметковната комплексност на локалниот  $BR$  модел. Ако претпоставиме дека дрвото за одлучување е балансирано и пополнето, тогаш пресметковната комплексност на  $ML-SVMDT$  архитектурата може да се претстави како:

$$O_{ML-SVMDT} = O(\log N) + O(\overline{TN_{sv}^i}) \quad (7.3)$$

каде  $\overline{N_{sv}^i} = \frac{1}{V} \sum_{i=0}^{i<V} N_{sv}^i$ . Горната граница на пресметковната комплексност на  $ML-SVMDT$  е определена со вториот израз  $O(\overline{TN_{sv}^i})$ . Ако се претпостави дека  $N_{sv} = k\overline{N_{sv}^i}$ , може да се заклучи дека пресметковната комплексност на  $ML-SVMDT$  архитектурата во фазата на тестирање е  $k$  пати помала од пресметкованата комплексноста на глобалниот  $BR$  модел.

$$O_{BR} = kO_{ML-SVMDT} \quad (7.4)$$

Параметарот  $k$  директно зависи од распределбата (дистрибуцијата) и бројот на примероци во глобалниот класификациски проблем и распределбата и бројот на примероци во локалните класификациски проблем во секој од листовите на дрвото за одлучување. Статистички вредноста на параметарот  $k$  се очекува да биде приближно еднаква на бројот на листови во дрвото за одлучување ( $V$ ). Дополнително намалување на времето на тестирање може да се очекува како резултат на отсуството на локални  $BR$  модели во некои од листовите на дрвото.

## 7.5 Експерименти

Во овој поглавје се прикажани експерименталниот дизајн и резултатите добиени од експерименталната евалуација на хибридна архитектура,  $BR$  и  $ML-C4.5$  методите. Во експерименталниот дизајн се опишани основните класификатори што се користат од методите за повеќезначна класификација и рангирање кои го трансформираат проблемот, поставувањето на параметрите на методите и карактеристиките на статистичката евалуација.

Целта на овој дел од истражувањето е да одговориме на следниве прашања:

- Дали предложената хибридна архитектура има подобри перформанси и е поефикасна од методите кои таа ги интегрира?
- Дали и како влијае функцијата на загуба која се користи во процесот на валидација?

За да одговориме на поставените прашања направена е споредба на перформансите на хибридна архитектура и методите за повеќезначна класификација и рангирање кои таа ги интегрира ( $BR$  и  $ML-C4.5$ ) на 11-те повеќезначни класификациски проблеми претставени во Глава 3. За евалуација на перформансите на методите во однос на прецизноста на предвидување и ефикасноста, користени се 18-те евалуациски мерки, исто така претставени во Глава 3. Дополнително, користен е *Вилкоксон тестот* [77] за да се процени дали разликата во перформансите меѓу хибридна архитектура,  $BR$  и  $ML-C4.5$  е статистички значајна.

За да одговориме конкретно на второто прашање, во експерименталната евалуација на хибридна архитектура користени се две функции на загуба: *Хаминг загуба* и  $F_1$ .

Методите на предложената архитектура кои ги користат овие функции на загуба во продолжение се референцирани како  $ML-SVM_{DT_{HL}}$  односно  $ML-SVM_{DT_{F1}}$ .

### 7.5.1 Експериментален дизајн

При споредбата на хибридна архитектура и нејзините две модификации, *BR* методот и *ML-C4.5* користена е библиотеката за повеќезначна и хиерархиска повеќезначна класификација и рангирање *MULAN*<sup>1</sup> [68] која како основна платформа ја користи библиотеката *WEKA*<sup>2</sup> [69]. Сите експерименти беа направени на сервер со *Интел Ксеон* (анг. *Intel Xeon*) процесор на 2.50GHz и 64GB RAM меморија со *Федора 14* (анг. *Fedora 14*) оперативен систем. Во продолжение на ова поглавје прво се претставени основните класификатори кои се користат од методите за повеќезначна класификација и рангирање кои го трансформираат проблемот, а потоа е опишан начинот на нагудување на параметрите на споредуваните методи.

#### 7.5.1.1 Основни класификатори, поставување на параметри и статистичка евалуација

Машините со носечки вектори кои се употребуваат кај хибридна архитектура и методот *бинарна релевантност* користат Гаусов кернел. Кернел параметарот  $\gamma$  и казната  $C$  кај двете методи се определени преку 10-кратна вкрстена валидација на сите примероци од податочното множество за обучување. Вредностите  $2^{-15}$ ,  $2^{-13}$ , ...,  $2^1$ ,  $2^3$  се користени за определување на параметарот  $\gamma$  (анг. *gamma*), додека  $C$  е определен од вредностите  $2^{-5}$ ,  $2^{-3}$ , ...,  $2^{13}$ ,  $2^{15}$ . За имплементацијата на машините со носечки вектори е користена *LIBSVM*<sup>3</sup> [70] библиотеката. Од друга страна, *ML-C4.5* го користи методот за поткастрување на дрвото по неговата изградба со помош на факторот на доверба. По изборот на оптималните вредности на параметрите посебно за секој метод *парови на ознаки* во однос на секое податочно множество, методите се обучени со примероците од соодветното податочно множество за обучување и евалуирани со примероците од податочно множество за тестирање.

Предложениот метод (и двете негови имплементации  $ML-SVM_{DT_{HL}}$  и  $ML-SVM_{DT_{F1}}$ ) користат една четвртина од примероците од множеството за обучување, за валидација, додека останатите примероци се користат за градење на дрвото за одлучување. При експерименталната евалуација на  $ML-SVM_{DT_{pre}}$ , како минимален број на примероци во листовите од дрвото за одлучување се користат шест различни вредности (30-80 со чекор 10). *ML-C4.5* и  $ML-SVM_{DT_{post}}$  методите користат *издигнување на поддрво* како метод за поткастрување по изградбата на дрвото со *фактор на доверба* од 0.25. Параметарот за минимален број на примероци кај *ML-C4.5* е поставен на 10.

Како и во претходната глава така и во оваа, за да ја процениме статистичката значајност на разликата на перформансите на хибридна архитектура базирана на дрвата за одлучување во однос на непосредните методи со кои таа е споредена, користен е *Вилкоксон тестот* [77].

<sup>1</sup> <http://mulan.sourceforge.net/>

<sup>2</sup> <http://www.cs.waikato.ac.nz/ml/weka/>

<sup>3</sup> <http://www.csie.ntu.edu.tw/~cjlin/libsvm/>

## 7.5.2 Резултати и дискусија

Во Табела 7.1 до Табела 7.4 се прикажани резултатите добиени од експерименталната евалуација на предложената хибридна архитектура ( $ML-SVMDT_{HL}$  и  $ML-SVMDT_{F1}$ ), нејзините две варијации  $ML-SVMDT_{pre}$  и  $ML-SVMDT_{post}$ ,  $BR$  и  $ML-C4.5$  методите. Во Табела 7.1 се прикажани перформансите на методите во однос на мерките базирани на примерок. Во Табела 7.2 се прикажани перформансите на методите во однос на мерките базирани на ознака, додека Табела 7.3 и Табела 7.4 ги прикажуваат перформансите во однос на мерките базирани на рангирање и времињата на обучување и тестирање, соодветно. Најдобрите резултати во табелите по евалуациска мерка и податочно множество се прикажани задебелено. Во првата колона од овие табели се означени евалуациските мерки. Во втората колона се прикажани имињата на споредуваните методи, а останатите колони ги прикажуваат резултатите добиени од експериментите, по една колона за секое од податочните множества.

Во Табела 7.5 се прикажани вредностите на  $p$  (ниво на значајност) според *Вилкоксон тестот* за секоја евалуациска мерка добиени со статистичка евалуација на перформансите на методите на хибридна архитектура ( $ML-SVMDT_{HL}$  и  $ML-SVMDT_{F1}$ ) во споредба со перформансите на  $BR$  и  $ML-C4.5$  методите. Слично како и во претходното поглавје, во првата колона се прикажани евалуациските мерки, додека во останатите колони се прикажани вредностите на нивото на значајност  $p$  на статистичките резултати. Знакот  $>$  во третата, петтата, седмата и деветтата колона покажува дека првиот метод е статистички подобар од вториот (прв метод vs втор метод, за секој од споредуваните парови), додека знакот  $<$  покажува дека вториот метод е статистички подобар од првиот. Вредностите на  $p$  помали од 0.05 покажуваат дека меѓу двата споредувани метода постои статистички значајна разлика и истите се задебелено прикажани. Во Табела 7.6 се прикажани бројот на листови и локални  $BR$  модели во двете методи на предложената хибридна архитектура за сите податочни множества поодделно.

Добиените резултати јасно покажуваат дека хибридна архитектура нуди одличен баланс на перформансите и пресметковната ефикасност. Ако добро се анализираат резултатите може да се забележи дека перформансите на хибридна архитектура се најчесто подобри од перформансите на двата метода кои оваа архитектура ги интегрира или многу слични на перформансите на подобриот од нив. Како што можеше да се очекува  $ML-SVMDT_{HL}$  постигна најдобри резултати во однос на *Хаминг загуба* евалуациската мерка, додека  $ML-SVMDT_{F1}$  во однос на  $F_1$  евалуациската мерка, како резултат на функциите на загуба кои ги користат во процесот на градење на нивните модели. Хибридна архитектура покажа и статистички значајно подобри резултати за повеќето евалуациски мерки базирани на примерок и ознака, додека за евалуациските мерки базирани на рангирање,  $BR$  методот покажа најдобри перформанси. Послабите перформанси на хибридна архитектура во однос на овие евалуациски мерки е резултат на нерангирањето на ознаките кои не се дел од листот во кој се означува даден примерок. Имено, означувањето на даден примерок се прави во еден од листовите од дрвото за одлучување (директна предикција од самиот лист или локалниот  $BR$  модел) во кој само ознаките кои се дел од листот се рангираат, а сите останати ознаки кои не се дел од листот едноставно се игнорираат.

**Табела 7.1** Перформанси на методите ( $BR$ ,  $ML-C4.5$ ,  $ML-SVMDT_{HL}$ ,  $ML-SVMDT_{F1}$ ,  $ML-SVMDT_{pre}$  и  $ML-SVMDT_{post}$ ) во однос на евалуациските мерки базирани на примерок

		emotions	scene	yeast	medical	enron	corel5k	tmc2007	mediamill	bibtex	delicious	bookmarks
<i>Hamming загуба</i>	<b>BR</b>	0.257	0.079	<b>0.190</b>	0.077	<b>0.045</b>	0.017	0.013	0.032	0.012	0.018	DNF
	<b>ML-C4.5</b>	<b>0.247</b>	0.141	0.234	0.013	0.053	0.010	0.093	0.044	0.016	0.019	0.009
	<b>ML-SVMDT<sub>HL</sub></b>	<b>0.247</b>	<b>0.077</b>	<b>0.190</b>	<b>0.011</b>	0.046	<b>0.009</b>	0.013	<b>0.030</b>	<b>0.011</b>	<b>0.017</b>	<b>0.008</b>
	<b>ML-SVMDT<sub>F1</sub></b>	0.258	0.099	0.212	0.012	0.049	<b>0.009</b>	<b>0.009</b>	0.031	0.012	0.018	0.009
	<b>ML-SVMDT<sub>pre</sub></b>	0.254	0.097	0.205	<b>0.011</b>	0.049	<b>0.009</b>	0.011	0.032	0.012	0.018	0.009
	<b>ML-SVMDT<sub>post</sub></b>	0.257	0.099	0.204	<b>0.011</b>	0.050	<b>0.009</b>	0.011	0.032	0.012	0.018	0.009
<i>точност</i>	<b>BR</b>	0.361	<b>0.689</b>	<b>0.520</b>	0.206	<b>0.446</b>	0.030	0.891	0.403	<b>0.348</b>	0.136	DNF
	<b>ML-C4.5</b>	<b>0.536</b>	0.569	0.480	0.730	0.418	0.002	0.110	0.052	0.108	0.001	0.237
	<b>ML-SVMDT<sub>HL</sub></b>	0.377	<b>0.689</b>	0.493	0.739	0.429	<b>0.058</b>	0.893	0.406	0.320	<b>0.148</b>	0.228
	<b>ML-SVMDT<sub>F1</sub></b>	0.386	0.656	0.494	0.737	0.423	0.052	0.901	<b>0.407</b>	0.321	0.147	0.226
	<b>ML-SVMDT<sub>pre</sub></b>	0.443	0.638	0.498	0.732	0.428	0.052	0.910	0.404	0.311	0.143	0.238
	<b>ML-SVMDT<sub>post</sub></b>	0.435	0.632	0.501	<b>0.740</b>	0.421	0.055	<b>0.912</b>	0.404	0.319	0.144	<b>0.246</b>
<i>прецизност</i>	<b>BR</b>	0.550	<b>0.718</b>	<b>0.722</b>	0.211	<b>0.703</b>	0.042	0.941	<b>0.731</b>	<b>0.515</b>	0.443	DNF
	<b>ML-C4.5</b>	<b>0.606</b>	0.592	0.620	0.797	0.623	0.005	0.146	0.056	0.123	0.001	0.271
	<b>ML-SVMDT<sub>HL</sub></b>	0.548	<b>0.718</b>	0.692	<b>0.806</b>	0.670	<b>0.142</b>	0.937	<b>0.731</b>	0.495	<b>0.506</b>	0.277
	<b>ML-SVMDT<sub>F1</sub></b>	0.549	0.628	0.705	0.777	0.662	0.139	<b>0.962</b>	<b>0.731</b>	0.482	0.497	0.278
	<b>ML-SVMDT<sub>pre</sub></b>	0.583	0.668	0.698	0.790	0.669	0.127	0.946	0.727	0.484	0.486	<b>0.281</b>
	<b>ML-SVMDT<sub>post</sub></b>	0.556	0.662	0.700	0.798	0.660	0.135	0.949	0.729	0.490	0.477	<b>0.281</b>
<i>одбив</i>	<b>BR</b>	0.409	<b>0.711</b>	0.591	0.735	0.497	0.055	0.928	0.450	0.373	0.155	DNF
	<b>ML-C4.5</b>	0.703	0.582	<b>0.608</b>	0.740	0.487	0.002	0.111	0.052	0.111	0.001	0.244
	<b>ML-SVMDT<sub>HL</sub></b>	0.697	<b>0.711</b>	0.566	0.754	<b>0.547</b>	<b>0.149</b>	0.936	0.457	<b>0.392</b>	<b>0.236</b>	0.248
	<b>ML-SVMDT<sub>F1</sub></b>	<b>0.706</b>	0.641	0.574	0.718	0.521	0.147	0.948	<b>0.481</b>	0.388	0.235	<b>0.256</b>
	<b>ML-SVMDT<sub>pre</sub></b>	0.545	0.655	0.581	0.760	0.486	0.054	<b>0.952</b>	0.452	0.327	0.159	0.246
	<b>ML-SVMDT<sub>post</sub></b>	0.535	0.648	0.586	<b>0.765</b>	0.480	0.056	0.948	0.449	0.338	0.161	0.255
<i>класификац. точност</i>	<b>BR</b>	0.129	<b>0.639</b>	<b>0.190</b>	0.000	<b>0.149</b>	0.000	0.772	0.080	<b>0.194</b>	0.004	DNF
	<b>ML-C4.5</b>	<b>0.277</b>	0.533	0.158	0.646	0.140	0.000	0.078	0.049	0.095	0.001	0.209
	<b>ML-SVMDT<sub>HL</sub></b>	0.183	<b>0.639</b>	0.168	<b>0.655</b>	0.145	<b>0.008</b>	0.776	<b>0.092</b>	0.183	0.002	0.205
	<b>ML-SVMDT<sub>F1</sub></b>	0.180	0.612	0.166	0.649	0.142	<b>0.008</b>	0.789	<b>0.092</b>	0.181	0.002	0.204
	<b>ML-SVMDT<sub>pre</sub></b>	0.178	0.590	0.165	0.643	0.140	<b>0.008</b>	0.796	0.091	0.179	0.004	0.210
	<b>ML-SVMDT<sub>post</sub></b>	0.198	0.585	0.166	<b>0.655</b>	0.138	<b>0.008</b>	<b>0.802</b>	0.091	0.179	<b>0.005</b>	<b>0.215</b>
<i>F<sub>1</sub></i>	<b>BR</b>	0.469	<b>0.714</b>	<b>0.650</b>	0.328	0.582	0.047	0.934	0.557	0.433	0.230	DNF
	<b>ML-C4.5</b>	<b>0.651</b>	0.587	0.614	0.768	0.546	0.003	0.126	0.054	0.117	0.001	0.257
	<b>ML-SVMDT<sub>HL</sub></b>	0.614	<b>0.714</b>	0.623	0.779	<b>0.602</b>	<b>0.145</b>	0.936	0.562	<b>0.438</b>	<b>0.322</b>	0.262
	<b>ML-SVMDT<sub>F1</sub></b>	0.618	0.634	0.633	0.746	0.583	0.143	<b>0.955</b>	<b>0.580</b>	0.430	0.319	<b>0.267</b>
	<b>ML-SVMDT<sub>pre</sub></b>	0.563	0.661	0.634	0.775	0.563	0.076	0.949	0.557	0.390	0.240	0.262
	<b>ML-SVMDT<sub>post</sub></b>	0.545	0.655	0.638	<b>0.781</b>	0.556	0.079	0.949	0.556	0.400	0.240	<b>0.267</b>

Дополнително, интересно е да се напомене дека перформансите на  $ML-SVMDT_{F1}$  се послаби за помалите податочни множества како резултат на поткаструвањето (анг. *over pruning*) што го предизвикува  $F_1$  валидациската функција поради малиот број на примероци во листовите од дрвото. Методите кои директно ги интегрираат  $ML-C4.5$  и

локалните  $BR$  модели во секој од листовите на дрвото ( $ML-SVMDT_{pre}$  и  $ML-SVMDT_{post}$ ) нудат за нијанса послаби перформанси од предложената хибридна архитектура.

**Табела 7.2** Перформанси на методите ( $BR$ ,  $ML-C4.5$ ,  $ML-SVMDT_{HL}$ ,  $ML-SVMDT_{F1}$ ,  $ML-SVMDT_{pre}$  и  $ML-SVMDT_{post}$ ) во однос на евалуациските мерки базирани на ознака

		emotions	scene	yeast	medical	enron	core15k	tmc2007	mediamill	bibtex	delicious	bookmarks
макро прецизност	BR	<b>0.684</b>	<b>0.843</b>	0.733	0.225	<b>0.721</b>	0.061	0.947	0.742	0.753	0.658	DNF
	ML-C4.5	0.607	0.619	0.618	0.796	0.613	0.160	0.940	0.597	0.359	0.000	0.632
	ML-SVMDT <sub>HL</sub>	0.683	<b>0.843</b>	<b>0.738</b>	<b>0.843</b>	0.669	<b>0.698</b>	0.942	0.755	0.787	<b>0.692</b>	0.798
	ML-SVMDT <sub>F1</sub>	0.643	0.798	0.730	0.839	0.648	0.657	<b>0.977</b>	<b>0.760</b>	0.788	0.679	0.793
	ML-SVMDT <sub>pre</sub>	0.634	0.787	0.695	0.834	0.668	0.664	0.947	0.749	<b>0.789</b>	0.662	<b>0.855</b>
	ML-SVMDT <sub>post</sub>	0.628	0.782	0.698	0.830	0.657	0.662	0.951	0.745	0.764	0.641	0.755
макро прецизност	BR	<b>0.721</b>	<b>0.844</b>	<b>0.628</b>	<b>0.399</b>	0.258	0.052	0.972	0.112	<b>0.528</b>	0.299	DNF
	ML-C4.5	0.602	0.635	0.377	0.263	0.142	0.004	0.925	0.046	0.128	0.000	0.292
	ML-SVMDT <sub>HL</sub>	0.654	<b>0.844</b>	0.559	0.364	0.246	<b>0.055</b>	0.980	0.261	0.513	<b>0.319</b>	<b>0.526</b>
	ML-SVMDT <sub>F1</sub>	0.632	0.805	0.551	0.377	0.250	<b>0.055</b>	<b>0.986</b>	<b>0.270</b>	0.480	0.318	0.495
	ML-SVMDT <sub>pre</sub>	0.625	0.785	0.515	0.379	0.266	<b>0.055</b>	0.984	0.258	0.495	0.312	0.485
	ML-SVMDT <sub>post</sub>	0.596	0.777	0.545	0.320	<b>0.307</b>	<b>0.055</b>	0.983	0.201	0.520	0.290	0.485
макро одбие	BR	0.406	0.694	0.587	0.725	0.464	0.057	0.917	0.415	<b>0.328</b>	0.143	DNF
	ML-C4.5	<b>0.712</b>	0.570	<b>0.603</b>	0.720	0.440	0.002	0.073	0.004	0.053	0.000	0.170
	ML-SVMDT <sub>HL</sub>	0.562	<b>0.698</b>	0.587	0.730	<b>0.529</b>	<b>0.109</b>	0.927	0.428	0.296	<b>0.182</b>	0.176
	ML-SVMDT <sub>F1</sub>	0.549	0.683	0.582	0.719	0.474	0.067	0.932	<b>0.437</b>	0.302	0.173	0.174
	ML-SVMDT <sub>pre</sub>	0.539	0.638	0.577	0.735	0.457	0.055	<b>0.943</b>	0.416	0.281	0.148	0.170
	ML-SVMDT <sub>post</sub>	0.534	0.631	0.579	<b>0.738</b>	0.457	0.058	0.940	0.411	0.289	0.149	<b>0.178</b>
макро одбие	BR	0.378	<b>0.703</b>	0.355	<b>0.423</b>	0.120	<b>0.023</b>	<b>0.915</b>	0.049	<b>0.250</b>	<b>0.072</b>	DNF
	ML-C4.5	<b>0.702</b>	0.573	<b>0.375</b>	0.249	0.107	0.005	0.085	0.002	0.034	0.000	0.098
	ML-SVMDT <sub>HL</sub>	0.556	<b>0.703</b>	0.343	0.323	0.128	0.020	0.906	0.062	0.223	<b>0.072</b>	0.095
	ML-SVMDT <sub>F1</sub>	0.538	0.635	0.331	0.328	0.114	0.019	0.912	<b>0.071</b>	0.217	0.069	0.094
	ML-SVMDT <sub>pre</sub>	0.511	0.646	0.344	0.313	0.119	0.014	0.903	0.060	0.198	0.066	0.103
	ML-SVMDT <sub>post</sub>	0.503	0.639	0.348	0.339	<b>0.130</b>	0.014	0.902	0.061	0.203	0.064	<b>0.111</b>
макро F1	BR	0.509	<b>0.761</b>	<b>0.652</b>	0.343	0.564	0.059	0.932	0.533	<b>0.457</b>	0.234	DNF
	ML-C4.5	<b>0.655</b>	0.593	0.610	0.756	0.512	0.004	0.135	0.007	0.093	0.000	0.268
	ML-SVMDT <sub>HL</sub>	0.617	<b>0.764</b>	<b>0.654</b>	<b>0.782</b>	<b>0.591</b>	<b>0.188</b>	0.934	0.546	0.430	<b>0.288</b>	<b>0.288</b>
	ML-SVMDT <sub>F1</sub>	0.592	0.736	0.648	0.774	0.547	0.121	<b>0.954</b>	<b>0.555</b>	0.437	0.276	0.285
	ML-SVMDT <sub>pre</sub>	0.583	0.705	0.630	0.781	0.543	0.102	0.945	0.535	0.414	0.242	0.283
	ML-SVMDT <sub>post</sub>	0.577	0.698	0.633	0.781	0.539	0.106	0.946	0.530	0.419	0.242	<b>0.288</b>
макро F1	BR	0.440	<b>0.765</b>	<b>0.392</b>	<b>0.361</b>	0.143	0.021	<b>0.942</b>	0.056	0.307	0.096	DNF
	ML-C4.5	<b>0.630</b>	0.596	0.370	0.250	0.115	0.008	0.124	0.003	0.045	0.000	0.119
	ML-SVMDT <sub>HL</sub>	0.601	<b>0.765</b>	0.379	0.337	0.155	<b>0.036</b>	0.940	0.081	<b>0.311</b>	<b>0.105</b>	<b>0.161</b>
	ML-SVMDT <sub>F1</sub>	0.535	0.705	0.373	0.343	0.148	0.032	0.935	<b>0.132</b>	0.258	0.102	0.158
	ML-SVMDT <sub>pre</sub>	0.525	0.704	0.372	0.331	0.141	0.020	0.939	0.075	0.249	0.089	0.144
	ML-SVMDT <sub>post</sub>	0.510	0.698	0.377	0.339	<b>0.159</b>	0.020	0.939	0.076	0.255	0.087	0.156

**Табела 7.3** Перформанси на методите ( $BR$ ,  $ML-C4.5$ ,  $ML-SVMDT_{HL}$ ,  $ML-SVMDT_{F1}$ ,  $ML-SVMDT_{pre}$  и  $ML-SVMDT_{post}$ ) во однос на евалуациските мерки базирани на рангирање

		emotions	scene	yeast	medical	enron	corel5k	tmc2007	mediamill	bibtex	delicious	bookmarks
една зрешка	BR	0.386	<b>0.180</b>	<b>0.236</b>	<b>0.135</b>	<b>0.237</b>	0.660	0.029	0.188	<b>0.346</b>	<b>0.354</b>	DNF
	ML-C4.5	<b>0.347</b>	0.394	0.312	0.198	0.309	0.762	0.145	0.194	0.529	0.411	0.643
	ML-SVMDT <sub>HL</sub>	0.376	<b>0.180</b>	0.267	0.153	<b>0.237</b>	0.655	0.021	0.180	0.384	0.375	0.599
	ML-SVMDT <sub>F1</sub>	0.370	0.192	0.266	0.155	0.256	0.660	0.014	<b>0.175</b>	0.387	0.375	0.590
	ML-SVMDT <sub>pre</sub>	0.371	0.259	0.253	0.156	0.264	<b>0.652</b>	<b>0.013</b>	0.183	0.407	0.372	0.594
	ML-SVMDT <sub>post</sub>	0.396	0.259	0.246	0.162	0.273	0.666	0.016	0.184	0.382	0.376	<b>0.577</b>
опфатеност	BR	2.307	<b>0.399</b>	<b>6.330</b>	<b>1.610</b>	12.530	104.800	<b>1.311</b>	20.481	<b>20.926</b>	<b>530.126</b>	DNF
	ML-C4.5	<b>2.069</b>	0.945	7.105	3.033	17.010	279.900	2.671	22.096	58.016	620.155	58.353
	ML-SVMDT <sub>HL</sub>	2.331	<b>0.399</b>	6.482	1.886	12.926	102.345	1.437	19.188	26.993	535.346	45.266
	ML-SVMDT <sub>F1</sub>	2.232	0.408	6.453	1.923	<b>12.520</b>	<b>100.342</b>	1.437	<b>19.102</b>	26.567	538.789	<b>40.323</b>
	ML-SVMDT <sub>pre</sub>	2.163	0.578	6.626	2.114	13.420	107.776	1.438	19.292	29.094	546.584	46.867
	ML-SVMDT <sub>post</sub>	2.144	0.599	6.549	2.108	13.575	108.250	1.453	19.212	25.648	541.022	42.855
згуба при рангирање	BR	0.246	<b>0.060</b>	<b>0.164</b>	<b>0.021</b>	<b>0.084</b>	<b>0.117</b>	<b>0.003</b>	0.061	<b>0.068</b>	<b>0.114</b>	DNF
	ML-C4.5	<b>0.210</b>	0.169	0.225	0.048	0.120	0.479	0.043	0.073	0.260	0.174	0.194
	ML-SVMDT <sub>HL</sub>	0.269	<b>0.060</b>	0.182	0.027	0.089	<b>0.117</b>	0.006	0.058	0.098	0.116	0.146
	ML-SVMDT <sub>F1</sub>	0.270	0.075	0.183	0.026	0.090	<b>0.117</b>	0.004	<b>0.056</b>	0.092	0.117	<b>0.136</b>
	ML-SVMDT <sub>pre</sub>	0.226	0.096	0.184	0.031	0.092	0.120	0.005	0.058	0.107	0.119	0.152
	ML-SVMDT <sub>post</sub>	0.230	0.100	0.179	0.031	0.096	0.119	0.006	0.058	0.091	0.117	0.137
просечна прецизност	BR	0.721	<b>0.893</b>	<b>0.768</b>	<b>0.896</b>	<b>0.693</b>	0.303	0.978	0.686	<b>0.597</b>	0.351	DNF
	ML-C4.5	<b>0.759</b>	0.751	0.706	0.823	0.629	0.196	0.842	0.669	0.392	0.321	0.378
	ML-SVMDT <sub>HL</sub>	0.721	<b>0.893</b>	0.760	0.872	0.681	<b>0.312</b>	0.979	0.703	0.551	<b>0.364</b>	0.420
	ML-SVMDT <sub>F1</sub>	0.720	0.788	0.750	0.856	0.643	0.307	<b>0.987</b>	<b>0.715</b>	0.555	0.358	0.426
	ML-SVMDT <sub>pre</sub>	0.747	0.843	0.746	0.872	0.668	0.306	0.985	0.698	0.536	0.362	0.421
	ML-SVMDT <sub>post</sub>	0.742	0.840	0.749	0.871	0.658	0.304	0.983	0.697	0.557	0.362	<b>0.435</b>

Во однос на времињата на обучување и тестирање,  $ML-C4.5$  покажа најдобри резултати и според *Вилкоксон тестот* истите се статистички значајни.  $ML-C4.5$  има најмали времиња на обучување и тестирање за сите податочни множества, освен за  $corel5k$  и  $delicious$ . За овие две податочни множества,  $ML-SVMDT_{pre}$  е побрз во фазата на обучување, а тоа е резултат на методот на поткастрување што тој го користи (поткастрување во текот на градење на дрвото). Методите за градење на хибридна архитектура се побрзи за 1.2 до 42 пати во фазата на обучување и 3 до 40 пати во фазата на тестирање од глобалниот  $BR$  метод.

**Табела 7.4** Перформанси на методите (*BR*, *ML-C4.5*, *ML-SVMDT<sub>HL</sub>*, *ML-SVMDT<sub>F1</sub>*, *ML-SVMDT<sub>pre</sub>* и *ML-SVMDT<sub>post</sub>*) во однос на евалуациските мерки базирани на рангирање

		emotions	scene	yeast	medical	enron	core5k	tmc2007	mediamill	bibtex	delicious	bookmarks
време на обучување	<b>BR</b>	4.0	71.0	145.0	18.0	318.0	926.0	42645.0	85468.0	11013.0	57053.0	DNF
	<b>ML-C4.5</b>	<b>0.3</b>	<b>8.0</b>	<b>14.0</b>	<b>3.0</b>	<b>15.0</b>	369.0	<b>469.0</b>	<b>2030.0</b>	<b>566.0</b>	2738.0	<b>4039.0</b>
	<i>ML-SVMDT<sub>HL</sub></i>	4.0	29.0	86.0	14.0	97.0	732.0	920.0	10345.0	1304.0	1945.0	160981.0
	<i>ML-SVMDT<sub>F1</sub></i>	2.3	28.0	52.0	10.0	83.0	793.0	1469.0	14035.0	2156.0	2314.0	17194.0
	<i>ML-SVMDT<sub>pre</sub></i>	2.0	25.0	42.0	8.0	77.0	<b>274.0</b>	888.0	9015.0	767.0	<b>1168.0</b>	53737.0
	<i>ML-SVMDT<sub>post</sub></i>	2.0	25.0	49.0	8.0	70.0	645.0	862.0	10789.0	769.0	1358.0	137660.0
време на тестирање	<b>BR</b>	1.0	25.0	23.0	4.0	50.0	25.0	927.0	6152.0	654.0	2045.0	DNF
	<b>ML-C4.5</b>	<b>0.0</b>	<b>1.0</b>	<b>0.1</b>	<b>0.1</b>	<b>0.2</b>	<b>1.0</b>	<b>1.7</b>	<b>1.0</b>	<b>6.5</b>	<b>19.0</b>	<b>21.0</b>
	<i>ML-SVMDT<sub>HL</sub></i>	0.4	8.0	4.0	0.7	7.0	8.0	18.0	353.0	48.0	102.0	1480.0
	<i>ML-SVMDT<sub>F1</sub></i>	0.2	1.3	4.3	1.4	7.5	9.0	28.0	425.0	69.0	169.0	3074.0
	<i>ML-SVMDT<sub>pre</sub></i>	0.6	12.0	10.0	1.8	18.0	9.0	74.0	398.0	84.0	189.0	4189.0
	<i>ML-SVMDT<sub>post</sub></i>	0.6	13.0	11.0	1.7	16.0	14.0	65.0	470.0	84.0	182.0	8022.0

**Табела 7.5** Ниво на значајност *p* според *Вилкоксон тестот* за секоја евалуациска мерка во однос на перформансите на методите *CLR*, *TSVM*, *TSCCM* и *TSPCCM*. Знакот > покажува дека првиот метод е статистички подобар од вториот (прв метод vs втор метод), додека знакот < покажува дека вториот метод е статистички подобар од првиот. Вредностите на *p* помали од 0.05 покажуваат дека меѓу двата споредувани метода постои статистички значајна разлика

	<i>ML-SVMDT<sub>HL</sub></i> vs <i>BR</i>	<i>ML-SVMDT<sub>F1</sub></i> vs <i>BR</i>	<i>ML-SVMDT<sub>HL</sub></i> vs <i>ML-C4.5</i>	<i>ML-SVMDT<sub>F1</sub></i> vs <i>ML-C4.5</i>
<i>Хаминг загуба</i>	<b>0.028</b> >	0.959 <	<b>0.005</b> >	<b>0.028</b> >
<i>точност</i>	0.508 >	0.878 <	<b>0.036</b> >	<b>0.036</b> >
<i>прецизност</i>	0.918 >	0.721 >	<b>0.012</b> >	<b>0.022</b> >
<i>одсив</i>	<b>0.028</b> >	0.114 >	<b>0.016</b> >	<b>0.025</b> >
<i>F<sub>1</sub></i>	<b>0.028</b> >	0.092 >	<b>0.013</b> >	<b>0.022</b> >
<i>класификациска точност</i>	0.415 >	0.646 >	<b>0.046</b> >	0.060 >
<i>микро прецизност</i>	0.152 >	0.575 >	<b>0.005</b> >	<b>0.005</b> >
<i>микро одсив</i>	<b>0.028</b> >	0.241 >	<b>0.047</b> >	0.060 >
<i>микро F<sub>1</sub></i>	<b>0.025</b> >	0.114 >	<b>0.009</b> >	<b>0.016</b> >
<i>макро прецизност</i>	0.508 <	0.338 <	<b>0.005</b> >	<b>0.005</b> >
<i>макро одсив</i>	0.610 <	0.203 <	0.092 >	0.092 >
<i>макро F<sub>1</sub></i>	0.241 >	0.878 <	<b>0.012</b> >	<b>0.022</b> >
<i>една грешка</i>	0.415 <	0.114 <	<b>0.009</b> >	<b>0.009</b> >
<i>опфатеност</i>	0.202 <	0.444 <	<b>0.007</b> >	<b>0.007</b> >
<i>загуба при рангирање</i>	<b>0.025</b> <	<b>0.019</b> <	<b>0.037</b> >	<b>0.037</b> >
<i>просечна прецизност</i>	0.759 <	0.203 <	<b>0.009</b> >	<b>0.022</b> >
<i>време на обучување</i>	<b>0.005</b> >	<b>0.005</b> >	0.059 <	<b>0.032</b> <
<i>време на тестирање</i>	<b>0.005</b> >	<b>0.005</b> >	<b>0.005</b> <	<b>0.005</b> <

Табела 7.6 Број на листови и локални *BR* модели во *ML-SVM*DT архитектурата

		emotions	scene	yeast	medical	enron	corel5k	tmc2007	mediamill	bibtex	delicious	bookmarks
<b>ML-SVM</b> DT <sub>HL</sub>	листови	15	37	22	11	41	23	38	61	24	44	130
	локални модели	8	20	7	6	21	21	36	49	16	37	74
<b>ML-SVM</b> DT <sub>F1</sub>	листови	10	11	15	5	25	20	46	56	16	57	134
	локални модели	3	1	7	3	10	17	45	43	15	56	98

## 7.6 Заклучок

Во оваа глава беше предложена хибридна архитектура базирана на дрва за одлучување кои користат локални машини со носечки вектори за повеќезначна класификација и рангирање. Претставени беа две методи за градење на оваа архитектура (*ML-SVM*DT<sub>HL</sub> и *ML-SVM*DT<sub>F1</sub>) и две модификации (*ML-SVM*DT<sub>pre</sub> и *ML-SVM*DT<sub>post</sub>).

Во експериментална евалуација беше направена споредба на предложената архитектура и нејзините две модификации со методите кои оваа архитектура ги интегрира *BR* и *ML-C4.5*. Методите ги споредивме на 11 повеќезначни класификациски проблеми од 5 апликациски области (класификација на текст, класификација на слики предикција на функција на гени, класификација на музика и класификација на видео) во однос на 18 различни евалуациски мерки (6 мерки базирани на ознака, 6 мерки базирани на примерок, 4 мерки базирани на рангирање и 2 мерки за ефикасноста на методите). За да ја процениме статистичката значајност на разликата на перформансите на споредуваните методи го користевме *Вилкоксон тестот*.

Врз основа на резултатите добиени од експериментите и статистичката евалуација може да се заклучи дека хибридната архитектура може да биде вистинскиот избор за повеќезначен класификатор кога се работи за големи податочни множества. Имено, таа нуди одличен баланс на перформансите во однос на предвидувањето и пресметковната ефикасност. Нејзините перформанси на предвидување најчесто се подобри од перформансите на двата метода кои оваа архитектура ги интегрира или многу слични на перформансите на подобриот од нив. Во однос на времињата на обучување и тестирање, хибридната архитектура е статистички значајно побрза од глобалниот *BR* метод според *Вилкоксон тестот* (конкретно 1.2 до 42 пати во фазата на обучување и 3 до 40 пати во фазата на тестирање).

Во идните истражувања ќе бидат разгледани неколку различни статистички тестови кои би можеле да се употребат како методи за поткастрување на дрвото во текот на неговата изградба, наместо да се користи поткастрување со валидација. Ќе биде анализирано влијанието на различни техники за градење на ансамбли од хибридната архитектура (*случајни шуми*, бегинг (анг. *bagging*), *бустинг*...) во однос на нејзините перформанси на предвидување и пресметковната комплексност.

## Глава 8

# Интеграција на HOMER и дрва за предиктивно кластерирање

Во оваа глава ќе биде претставен уште еден метод за повеќезначна класификација и рангирање. Имајќи ги во предвид предностите и недостатоците на различните методи за повеќезначна класификација и рангирање кои беа споредени во Глава 5, направена е интеграција на двете методи (*HOMER* и ансамблиите од дрва за предиктивно кластерирање – *RF-PCT*) кои покажаа најдобри перформанси во однос на точноста на предвидувањата и пресметковната комплексност. Иако овие две методи покажаа најдобри перформанси, сепак интересен е фактот што секој од овие две методи е доминантен за различна евалуациска мерка. Имено, методот *HOMER* постигна најголем *odsv* во однос на евалуациските мерки базирани на примерок и ознака, додека *RF-PCT* беше *најпрецизен* и покажа извонредни резултати за големите податочни множества. Дополнително, како резултат на машините со носечки вектори кои беа користени како основен класификатор, *HOMER* покажа добри перформанси за помалите податочни множества и податочните множества со голем број на обележја, а *RF-PCT* се истакна со својата пресметковна *ефикасност* на големите податочни множества, што е резултат на ефикасноста која ја поседуваат дрвата за одлучување.

При анализата на резултатите во Глава 5, еден од заклучоците кои беа донесени се однесуваше на причините за подобриот *odsv* на методот *HOMER* во однос на другите методи кои користат машини со носечки вектори како основен класификатор. Тој заклучок се однесуваше на вештачката хиерархија која *HOMER* ја дефинира над податочното множество со помош на алгоритам за хиерархиско кластерирање. Имено, според нашите претпоставки токму како резултат на хиерархијата која се дефинира, *HOMER* ги подобрува перформансите во однос на *odsvom* и останатите евалуациски мерки базирани на примерок и ознака. Доколку постои хиерархиска законитост на означување на примероците во рамките на податочното множество (јасно издвојување на групи на примероци со заеднички карактеристики во секој од јазлите на хиерархиската структура), *HOMER* успева да ја детектира таа законитост и да ги подобри своите перформанси.

Со оглед на тоа што *PCTs* можат да се применат за решавање не само на повеќезначни туку и на хиерархиски повеќезначни проблеми преку дефинирање на варијансата и прототипот, хиерархијата која *HOMER* ја дефинира над податочното множество може да се искористи како дополнителен влез во *PCTs*. Имено, по организирањето на ознаките (предикциите) на секој од примероците во согласност со вештачката хиерархија дефинирана од *HOMER*, хиерархискиот повеќезначен класификациски проблем добиен со соодветна трансформација на оригиналниот повеќезначен класификациски проблем може да се решава со помош на *PCTs* за хиерархиска повеќезначна класификација. На тој начин, ново дефинираната хиерархија

над податочното множество добива можност да влијае на процесот на дефинирање на кластерите и пресметувањето на вредноста на варијансата во секој од јазлите на дрвото (*PCT*). Доколку хиерархијата правилно ја отсликува вистинската организација во податочното множество, се очекува таа позитивно да влијае на перформансите и пресметковната комплексност на *PCT*.

Во продолжение се претставени методот кој го користи *HOMER* за дефинирање на вештачката хиерархија над податочните множества и дрвата за предиктивно кластерирање (*PCTs*) кои се користат за хиерархиска повеќезначна класификација (пресметување на варијансата и функцијата на прототип).

## 8.1 Креирање на вештачка хиерархија

*HOMER* користи хиерархија од повеќезначни класификатори за решавање на проблемот на хиерархиски организираното податочно множество. Авторите на овој метод предлагаат три алгоритми за конструирање на хиерархијата [14]: Случајно дефинирана хиерархија, хиерархија дефинирана со алгоритмот *k-средини* и хиерархија дефинирана со алгоритмот *балансирани k-средини*. Најдобри резултати се добиени со употреба на третиот алгоритам (*балансирани k-средини*). Овој алгоритам се користи во процесот на интеграција на *HOMER* и дрвата за предиктивно кластерирање за решавање на хиерархиски повеќезначни класификациски проблеми.

**Балансирани *k-средини*** е алгоритам за балансирано хиерархиско кластерирање. Се базира на добро познатиот алгоритам за кластерирање *k-средини* [86]. За разлика од *k-means*, предложениот алгоритам воведува експлицитни ограничувања во однос на големината на секој кластер. Кластерирањето се прави само во однос на ознаките, т.е., за примероците  $(\mathbf{x}_i, \mathcal{Y}_i)$  кои се наоѓаат во конкретен јазел од хиерархиската структура (Слика 5.4), алгоритмот при кластерирањето го користи само делот со ознаките  $\mathcal{Y}_i$  ( $(\mathbf{x}_i, \mathcal{Y}_i) \in E_n$ , каде  $E_n$  е множеството на примероци во јазлите од хиерархиската структура). Во првото ниво од хиерархијата (коренот на дрвото) множеството на примероци  $E_n$  е идентично со целото податочно множество ( $E_n = E$ ). Множеството на ознаки на конкретен примерок во даден јазел ( $w_i$ ) се дефинира како пресек од множеството на ознаки во конкретниот јазел ( $\mathcal{L}_n$ ) и множеството на ознаки за конкретниот примерок ( $\mathcal{Y}_i$ )  $w_i = \mathcal{Y}_i \cap \mathcal{L}_n$ . За даден јазел, алгоритмот за хиерархиско кластерирање како влез ги прима множеството на ознаки од јазелот  $\mathcal{L}_n$ , множествата на ознаки од сите примероци кои се дел од тој јазел  $\mathbf{W}_n$ , бројот на кластери  $k$  и бројот на итерации  $it$ . Како излез се добиваат  $k$  дисјунктни подмножества од  $\mathcal{L}_n$  со приближно еднаква големина. Алгоритмот е претставен на Слика 8.1.

Клучниот елемент на алгоритмот се однесува на листата од ознаки  $C_l$  (подредена во растечки редослед според растојанието до центарот на кластерот  $c_l$ ) во секој кластер  $l$ . Доколку со додавањето на ознака на соодветна позиција во подредената листа од ознаки за даден кластер, се надмине максималниот број на ознаки по кластер (приближно еднаков на бројот на ознаки во јазелот поделен со бројот на кластери), последната ознака (најоддалечената) се додава во листата на ознаки на следниот најсоодветен кластер. Во најлош случај, ова може да води до  $k - 1$  дополнителни додавања. Во овој алгоритам бројот на итерации се ограничува преку параметарот  $it$ , дефиниран од корисникот, за разлика со алгоритмот *k-средини* кој конвергенцијата ја гарантира на друг начин.

---

```
procedure BalancedkMeans( $k, \mathcal{L}_n, \mathbf{W}_n, it$ )
1: for  $i \leftarrow 1$  to  $k$  do
2:   //initialize clusters and clusters centers
3:    $C_i \leftarrow \emptyset$ ;
4:    $c_i \leftarrow$  random member of  $\mathcal{L}_n$ ;
5: while  $it > 0$  do
6:   for each  $\lambda \in \mathcal{L}_n$  do
7:     for  $i \leftarrow 1$  to  $k$  do
8:        $d_{\lambda i} \leftarrow$  distance( $\lambda, c_i, \mathbf{W}_n$ );
9:        $finished \leftarrow$  false;
10:       $v \leftarrow \lambda$ ;
11:      while not  $finished$  do
12:         $j \leftarrow \operatorname{arg\,min}_i d_{vi}$ 
13:        Insert sort( $v, d_v$ ) to sorted list  $C_j$ ;
14:        if  $|C_j| > \lceil |\mathcal{L}_n|/k \rceil$  then
15:           $v \leftarrow$  remove last element from  $C_j$ ;
16:           $d_{vj} \leftarrow \infty$ 
17:        else
18:           $finished \leftarrow$  true;
19:        recalculate centers;
20:         $it \leftarrow it - 1$ ;
21: return  $C_1, \dots, C_k$ 
```

---

Слика 8.1 Балансирани  $k$ -средини.

За да се пресметаат центрите на кластерите и растојанието на ознаките до секој од тие кластери, прво множеството на ознаки  $\mathbf{W}_n = \{w_1, w_2, \dots, w_{|E_n|}\}$  се претставува со помош на бинарни вектори  $w_{ij}, i=1 \dots |E_n|, j=1 \dots |\mathcal{L}_n|$ , каде  $w_{ij} = 1$  ако  $\lambda_j \in w_i$  и  $w_{ij} = 0$  во спротивната ситуација. Потоа, центрите се пресметуваат како аритметичка средина од векторите на ознаките во секој кластер, а за пресметување на растојанието меѓу центрите на кластерите и векторите на ознаките се користи Евклидово растојание (формула (8.1)).

$$distance(\lambda_j, c_l, \mathbf{W}) = \sqrt{\sum_{i=1}^{|\mathcal{L}_n|} (c_{li} - w_{ij})^2} \quad (8.1)$$

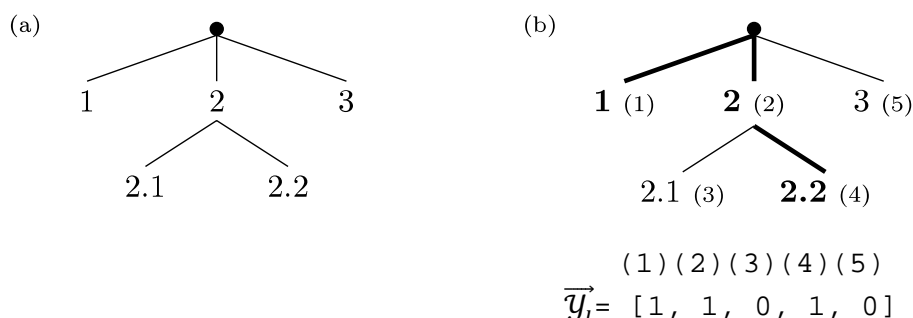
## 8.2 Дрва за предиктивно кластерирање за хиерархиска повеќезначна класификација и рангирање

Како што спомнавме во Глава 5, дрвата за предиктивно кластерирање (*PCTs*) се применети за повеќецелна класификација (анг. *multi-target classification*) [56], хиерархиска повеќезначна класификација (анг. *hierarchical multi-label classification*) [63] и предикција на временски серии [64]. За потребите на овој метод се користат *PCTs* за хиерархиска повеќезначна класификација. Во продолжение се опишани дрвата

за предиктивно кластерирање за хиерархиска повеќезначна класификација и начинот на пресметување на варијансата и функцијата на прототип кај овој вид на дрва за предиктивно кластерирање.

### 8.2.1 Дрва за предиктивно кластерирање за хиерархиска повеќезначна класификација

За да се овозможи примена на *PCTs* за хиерархиска повеќезначна класификација и рангирање, варијансата и прототипот може да се дефинираат на начинот опишан во [63]. Множеството на ознаки за секој примерок ги вклучува сите јазли од хиерархијата на ознаки и е претставено како вектор од бинарни компоненти;  $n$ -тата компонента од векторот е „1“ ако примерокот е означен со ознаката  $\lambda_n$ , а во спротивен случај „0“ (Слика 8.2).



**Слика 8.2** Пример хиерархија кај *PCTs* за хиерархиска повеќезначна класификација и векторот на ознаки  $\vec{Y}_i$

Варијансата на дадено множество од примероци  $E$  е дефинирана како средна вредност од квадратот на растојанието меѓу векторот на ознаките ( $\vec{Y}_i$ ) за секој примерок и средниот вектор на класите за целото множество ( $\vec{Y}$ ):

$$Var(E) = \frac{\sum_i d(\vec{Y}_i, \vec{Y})^2}{|E|} \quad (8.2)$$

Во контекстот на хиерархиска повеќезначна класификација, сличноста во повисоките нивоа од хиерархијата е позначајна отколку во пониските. Ова е земено предвид преку растојанието кое се користи во формулата (8.2) – утежнето Евклидово растојание:

$$d(\vec{Y}_1, \vec{Y}_2) = \sqrt{\sum_k w(\lambda_k) \cdot (\vec{Y}_1(\lambda_k) - \vec{Y}_2(\lambda_k))^2} \quad (8.3)$$

каде  $\vec{Y}_i(\lambda_k)$  е  $k$ -тата компонента на векторот на ознаки  $\vec{Y}_i$  (која се однесува на ознаката  $\lambda_k$ ) за даден примерок  $\mathbf{x}$ . Тежината за ознаката  $w(\lambda)$  се намалува со длабочината на ознаката во хиерархијата. Попрецизно,  $w(\lambda) = w_0 \cdot avg_j\{w(p_j(\lambda))\}$ , каде со  $p_j(\lambda)$  е означен  $j$ -тиот родител на ознаката  $\lambda$ , и  $0 < w_0 < 1$ .

## 8.2.2 Предвидувања на ознаките

Класификациското дрво во своите листови ја зачувува мнозинската ознака за соодветниот лист, која всушност ќе биде предвидувањето за сите примероци кои ќе пристигнат во тој лист. Во случајот на хиерархиска повеќезначна класификација, даден примерок може да биде означен со повеќе ознаки, а со тоа и нотацијата „мнозинска ознака“ не може едноставно да се примени. Наместо ова, средната вредност  $\bar{U}$  на векторите на ознаките за примероците во соодветниот лист се зачувува како предвидување. Притоа  $n$ -тата компонента од  $\bar{U}$  се интерпретира како веројатноста за даден примерок, кој пристигнал во тој лист, да биде означен со ознаката  $\lambda_n$ .

Предвидувањето за даден примерок кој пристигнува во даден лист може да се добие со примена на кориснички дефинирана прагова вредност  $\tau$  над веројатноста. Ако  $n$ -тата компонента од  $\bar{U}$  е над  $\tau$ , тогаш примероците треба да бидат означени со ознаката  $\lambda_n$ . При предвидувањето на ознаките се почитува ограничувањето наметнато од хиерархијата (предвидувањата на ознаките се потчинуваат на родител-дете релациите од хиерархијата). Затоа вредностите на праговите  $\tau$  се избираат на следниот начин:  $\tau_i \leq \tau_j$ , во случај кога  $\lambda_i \leq_h \lambda_j$  ( $\lambda_i$  е предок на  $\lambda_j$ ). Праговата вредност се избира во зависност од контекстот. Корисникот може да ја постави праговата вредност да добие висока *прецизност* на сметка на слаб *odsub* или обратно, максимизирање на  $F_1$ , максимизирање на интерпретабилноста или прифатливоста на дадениот модел итн.

## 8.3 Употреба на ансамблот случајни шуми од PCTs за хиерархиска повеќезначна класификација и методот HOMER за решавање на проблемот на повеќезначна класификација и рангирање

### 8.3.1 Дизајн

За да се применат *PCTs* за хиерархиска повеќезначна класификација, хиерархиската структура, која ја дефинира *HOMER* над податоците треба да го запази правилото за хиерархиската повеќезначна класификација: Даден примерок, кој е означен со одредена ознака  $\lambda$ , автоматски е означен со сите супер-ознаки на ознаката  $\lambda$  (супер-ознаките се наоѓаат повисоко во хиерархијата од ознаката  $\lambda$ ).

По дефинирањето на хиерархијата на ознаките во согласност со правилото за хиерархиска повеќезначна класификација со помош на алгоритмот за балансирано хиерархиско кластерирање, целото податочно множество се трансформира (пресликува) во ново податочно множество. За разлика од дескриптивните обележја на примероците кои остануваат непроменети во новото податочно множество, сите целни обележја се променуваат. Тие се пресликуваат во хиерархиската структура дефинирана со алгоритмот *балансиран к-средини*. При пресликувањето единствено треба да се запази правилото за хиерархиската повеќезначна класификација (ако примерокот оригинално е означен со дадена ознака  $\lambda$ , тој се означува и со сите супер-ознаки на таа ознака). Во Табела 8.1 е претставено пример на повеќезначно податочно множество чии примероци оригинално се означени со ознаките од множеството  $\mathcal{L} = \{\lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5, \lambda_6, \lambda_7, \lambda_8\}$ . За хиерархијата прикажана на Слика 5.4 (добиена со алгоритмот за балансирано хиерархиско кластерирање), податочното множество кое се добива со трансформација (пресликување) на оригиналното повеќезначно податочно

множество (со помош на алгоритмот за балансирано хиерархиско кластерирање) е претставено во Табела 8.2.

**Табела 8.1** Пример за оригинално повеќезначно податочно множество

Примерок	Атрибути/обележја	Множество на ознаки
<b>X<sub>1</sub></b>	$x_{11}, x_{12}, \dots, x_{1D}$	$\{\lambda_1, \lambda_4, \lambda_8\}$
<b>X<sub>2</sub></b>	$x_{21}, x_{22}, \dots, x_{2D}$	$\{\lambda_3, \lambda_6\}$
<b>X<sub>3</sub></b>	$x_{31}, x_{32}, \dots, x_{3D}$	$\{\lambda_1\}$
<b>X<sub>4</sub></b>	$x_{41}, x_{42}, \dots, x_{4D}$	$\{\lambda_2, \lambda_3, \lambda_4, \lambda_8\}$
<b>X<sub>5</sub></b>	$x_{51}, x_{52}, \dots, x_{5D}$	$\{\lambda_1, \lambda_4, \lambda_7\}$

**Табела 8.2** Повеќезначно податочно множество добиено со трансформација на оригиналното повеќезначно податочно множество прикажано во Табела 8.1 со помош на алгоритмот за балансирано хиерархиско кластерирање

Примерок	Атрибути/обележја	Множество на ознаки
<b>X<sub>1</sub></b>	$x_{11}, x_{12}, \dots, x_{1D}$	$\{\mu_1/\mu_2/\lambda_1, \mu_1/\mu_3/\lambda_4, \mu_1/\mu_4/\lambda_8\}$
<b>X<sub>2</sub></b>	$x_{21}, x_{22}, \dots, x_{2D}$	$\{\mu_1/\mu_2/\lambda_3, \mu_1/\mu_4/\lambda_6\}$
<b>X<sub>3</sub></b>	$x_{31}, x_{32}, \dots, x_{3D}$	$\{\mu_1/\mu_2/\lambda_1\}$
<b>X<sub>4</sub></b>	$x_{41}, x_{42}, \dots, x_{4D}$	$\{\mu_1/\mu_2/\lambda_2, \mu_1/\mu_3/\lambda_3, \mu_1/\mu_3/\lambda_4, \mu_1/\mu_4/\lambda_8\}$
<b>X<sub>5</sub></b>	$x_{51}, x_{52}, \dots, x_{5D}$	$\{\mu_1/\mu_2/\lambda_1, \mu_1/\mu_3/\lambda_4, \mu_1/\mu_4/\lambda_7\}$

По трансформацијата на податочното множество, ново генерираниот хиерархиски повеќезначен класификациски проблем се решава со помош на ансамлот *случајни шуми* од *PCTs* за хиерархиска повеќезначна класификација. Конечната предикција за секој примерок од новото хиерархиско податочно множество донесена од ансамлот од *PCTs* за хиерархиска повеќезначна класификација се добива со комбинирање на предикциите од сите основни предиктивни модели од кои тој е составен. Таа се добива со пресметување на средна вредност од поединечните предикции со примена на праговата вредност претходно опишана во 8.2.2. Притоа треба да се забележи дека предикциите за примероците од секој предиктивен модел од ансамлот се во согласност на хиерархиската структура дефинирана над податочното множество. Како што споменавме во 8.2.2, *PCTs* ја предвидуваат веројатноста даден примерок да биде означен со конкретна ознака, што значи дека предвидувањето на ознаките во хиерархискиот повеќезначен проблем се однесува на сите ознаки од хиерархијата (ознаките од оригиналниот повеќезначен класификациски проблем и вештачки генерираните ознаки – мета ознаките).

За да се направи евалуација на перформансите на системот во контекст на оригиналниот повеќезначен класификациски проблем, потребно е да се направи соодветна трансформација на конечните предикции. Во конкретниот случај, ознаките дефинирани во оригиналниот повеќезначен класификациски проблем (ознаките од интерес) се всушност ознаките од последното ниво на хиерархијата (види 5.4.1, Слика 5.4) од ново дефинираниот хиерархиски повеќезначен класификациски проблем. Поради тоа, трансформацијата на конечните предикции се прави на тој начин што се евалуираат само предикциите на ознаките од интерес (ознаките од последното ниво на хиерархијата), додека предикциите на сите останати ознаки (мета ознаките) едноставно се игнорираат.

### 8.3.2 Интеграција на методот HOMER и PCTs за хиерархиска повеќезначна класификација

---

```
procedure RF-HHPCT( $E^{train}, E^{test}$ ) returns performance
1:  $\mathbf{W}_H$  = DefineHOMERHierarchy( $\mathbf{W}$ );

2: //transform multi-label dataset to hierarchical
3: //multi-label dataset according to balanced k-means
4:  $E_H^{train}$  = MLToHMLTrainDataset( $E^{train}, \mathbf{W}_H$ );
5:  $E_H^{test}$  = MLToHMLTestDataset( $E^{test}, \mathbf{W}_H$ );

6: //solve transformed hierarchical multi-label problem with
7: //Random Forest of PCTs for HMLC
8: Ensemble = RF-PCT( $E_H^{train}$ );
9: //generate predictions
10:  $P_H$  = Ensemble( $E_H^{test}$ );

11: //transform hierarchical multi-label predictions  $P_H$  to
12: //flat multi-label predictions  $P$ 
13:  $P$  = TransformHMLPredictions( $P_H, \mathbf{W}$ );
14: return EvaluatePredictions( $P$ );
```

---

Слика 8.3 RF-HHPCT (анг. *Random Forest - Homer Hierarchy Predictive Clustering Trees*)

Целиот процес кој ги вклучува дефинирањето на хиерархијата со *HOMER*, решавањето на хиерархискиот повеќезначен класификациски проблем со помош на ансамблот *случајни шуми* од *PCTs* за хиерархиска повеќезначна класификација и евалуацијата на предикциите во контекст на оригиналниот повеќезначен класификациски проблем го нарекуваме *RF-HHPCT* (анг. *Random Forest – Homer Hierarchy PCTs*). Во продолжение го користиме овој термин за предложената интеграција на *HOMER* и *RF-PCT*. Процесот на интеграција е претставен со псевдокод на Слика 8.3.  $E^{train}$  е податочното множество за обучување,  $E^{test}$  е податочното множество за тестирање, додека  $\mathbf{W}$  е множеството на ознаки на сите примероци од  $E$ .  $\mathbf{W}_H$  е новото множество на хиерархиски дефинираните ознаки со помош на алгоритмот *балансиран  $k$ -средини*.  $E_H^{train}$  и  $E_H^{test}$  се хиерархиските податочни множества за обучување и тестирање соодветно, добиени со трансформација на оригиналните податочни множества  $E^{train}$  и  $E^{test}$  и множество на хиерархиски дефинираните ознаки  $\mathbf{W}_H$ .  $P_H$  се хиерархиските предикции за примероците од  $E_H^{test}$  кои се добиваат со решавање на вештачки генерираниот хиерархиски повеќезначен класификациски проблем, додека  $P$  се предикциите за  $E^{test}$  на оригиналните ознаки од повеќезначниот проблем. Предикциите  $P$  се добиваат со трансформација на предикциите  $P_H$ .

## 8.4 Експерименти

Во ова поглавје се прикажани резултатите од експериментите направени со предложениот метод (*RF-HHPCT*). Неговите перформанси се споредени со методите

кои тој ги интегрира *HOMER* и *RF-PCT*, кои воедно се и методите кои покажаа најдобри просечни перформанси на обемната експериментална евалуација прикажана во Глава 5.

Целта на овој дел од истражувањето е да одговориме на следниве прашања:

- Дали предложениот метод има подобри перформанси и е поефикасен од методите кои тој ги интегрира?
- Дали вештачки дефинираната хиерархија позитивно влијае на перформансите на предвидување?
- Дали и како вештачки дефинираната хиерархија влијае на пресметковната комплексност на дрвата за предиктивно кластерирање?

За да одговориме на поставените прашања измерени се перформансите на трите методи на сите повеќезначни класификациски проблеми претставени во Глава 3, во однос на 18-те евалуациски мерки (6 евалуациски мерки базирани на примерок, 6 евалуациски мерки базирани на ознака, 4 евалуациски мерки базирани на рангирање и времињата на обучување и тестирање) користени во обемната експериментална евалуација. Дополнително, користен е *Вилкоксон тестот* [77] за да се процени дали разликата во перформансите меѓу *RF-HHPCT*, *HOMER* и *RF-PCT* е статистички значајна.

За да одговориме на второто и третото прашање, поодделно се анализирани и споредени перформансите на *RF-HHPCT* при решавање на вештачки дефинираниот хиерархиски повеќезначен проблем и *RF-PCT/HOMER* при решавање на оригиналниот повеќезначен класификациски проблем.

Пред да бидат прикажани резултатите и анализата на постигнатите перформанси, опишани се експерименталниот дизајн и статистичката евалуација која се користи за проценување на значајноста на разликата меѓу перформансите на споредуваните методи.

#### 8.4.1 Експериментален дизајн

При споредбата на методите користени се библиотеката *MULAN*<sup>1</sup> [68] и рамката за предиктивно кластерирање *CLUS*<sup>2</sup>. *MULAN* библиотеката се користи за обучување и тестирање на *HOMER* методот, додека *CLUS* за *RF-PCT*. За предложениот метод *RF-HHPCT*, се користи *CLUS-HMC* (со соодветните растојанија и прототипи) на дрвата за предиктивно кластерирање (*PCTs*) за хиерархиска повеќезначна класификација. Сите експерименти се направени на сервер со *Интел Ксеон* (анг. *Intel Xeon*) процесор на 2.50GHz и 64GB RAM меморија со *Федора 14* (анг. *Fedora 14*) оперативен систем.

##### 8.4.1.1 Основни класификатори, поставување на параметри и статистичка евалуација

Параметрите на сите методи во експерименталната евалуација беа поставени во согласност со препораките од литературата. За *RF-PCT* и *HOMER* се користени истите параметри како во Глава 5. Конкретно, при обучување на *RF-PCT* се користени

---

<sup>1</sup> <http://mulan.sourceforge.net/>

<sup>2</sup> <http://clus.sourceforge.net/>

случајни шуми од 100 непоткастрени предиктивни дрва за кластерирање за повеќецелна класификација (број на модели препорачан од Bauer и Kohavi [67]). При обучување на основните класификатори (*PCT*) бројот на дескриптивните обележја кои се задржуваат во секој од јазлите при обучување на ансамлот го поставивме на 10% од вкупниот број на дескриптивните обележја. Истите вредности се користени и за *RF-HHPCT* (хиерархиската варијанта на ансамлот на дрвата за предиктивно кластерирање). Исто како и за *HOMER*, така и за *RF-HHPCT* потребно е да се оптимизира бројот на кластери во хиерархијата која овие два метода треба да ја детектираат и дефинираат. За овој параметар, и кај двата метода, пробавме 5 различни вредности (2, 3, 4, 5 и 6).

За имплементацијата на машините со носечки вектори кои се користени како основен класификатор кај *HOMER* е користена *LIBSVM*<sup>1</sup> [70] библиотеката. Конкретно, користени се машини со носечки вектори со Гаусов кернел, каде кернел параметарот *гамма* (анг. *gamma*) и казната *C* се определени преку 10-кратна вкрстена валидација на примероците од податочното множество за обучување. Вредностите  $2^{-15}$ ,  $2^{-13}$ , ...,  $2^1$ ,  $2^3$  се користени за определување на параметарот *гамма*, додека *C* е определен од вредностите  $2^{-5}$ ,  $2^{-3}$ , ...,  $2^{13}$ ,  $2^{15}$ .

За да ја процениме статистичката значајност на разликата на перформансите на предложениот метод (*RF-HHPCT*) во однос на методите кои таа ги интегрира, како што е предложено во истражувањето на Demšar [74], користен е *Вилкоксон тестот* [77].

#### 8.4.2 Резултати и дискусија

Резултатите кои се добиени од експерименталната евалуација на предложениот метод (*RF-HHPCT*), *HOMER* и ансамлот *случајни шуми* од дрвата за предиктивно кластерирање за повеќецелна класификација се претставени во четири табели (Табела 8.4, Табела 8.5, Табела 8.6 и Табела 8.7). Во Табела 8.4 се прикажани перформансите на методите во однос на мерките базирани на примерок. Во Табела 8.5 се прикажани перформансите на методите во однос на мерките базирани на ознака, додека Табела 8.6 и Табела 8.7 ги прикажуваат перформансите во однос на мерките базирани на рангирање и пресметковна ефикасност, соодветно. Најдобрите резултати во табелите по евалуациска мерка и податочно множество се прикажани задебелено. Во Табела 8.3 се прикажани вредностите на параметарот *k* од алгоритмот *балансиран k-средини* за кои се добиени најдобрите резултати за *RF-HHPCT*. Во првата колона од четирите табели кои ги прикажуваат перформансите на споредуваните методи се означени евалуациските мерки. Во втората колона се прикажани имињата на споредуваните методи, а останатите колони ги прикажуваат резултатите добиени од експериментите, по една колона за секое од податочните множества.

Во Табела 8.8 се прикажани вредностите на *p* (ниво на значајност) според *Вилкоксон тестот* за секоја евалуациска мерка добиени со статистичка евалуација на методите *RF-HHPCT*, *RF-PCT* и *HOMER* по принципот секој против секој. Во првата колона се прикажани евалуациските мерки, додека во останатите колони се прикажани вредностите на нивото на значајност *p* на статистичките резултати. Знакот > во третата и петтата колона покажува дека првиот метод е статистички подобар од вториот (прв метод vs втор метод, за секој од споредуваните парови), додека знакот < покажува дека вториот метод е статистички подобар од првиот. Вредностите на *p* помали од 0.05

<sup>1</sup> <http://www.csie.ntu.edu.tw/~cjlin/libsvm/>

покажуваат дека меѓу двата споредувани методи постои статистички значајна разлика и истите се прикажани задебелено.

**Табела 8.3** Вредностите на параметарот  $k$  (на *balanced k-means* алгоритмот) за сите податочни множества за кои се добиени просечно најдобрите перформанси за *RF-HHPCT*

	<b>emotions</b>	<b>scene</b>	<b>yeast</b>	<b>medical</b>	<b>enron</b>	<b>corel5k</b>	<b>tmc2007</b>	<b>mediamill</b>	<b>bibtex</b>	<b>delicious</b>	<b>bookmarks</b>
<b><math>k</math></b>	<b>2</b>	<b>3</b>	<b>5</b>	<b>3</b>	<b>4</b>	<b>3</b>	<b>3</b>	<b>3</b>	<b>3</b>	<b>4</b>	<b>3</b>

Од спроведените експерименти и добиените резултати, може да се донесат неколку значајни заклучоци кои се однесуваат на перформансите на методите.

Прво, гледано во целина *RF-HHPCT* ги подобрува перформансите на *RF-PCT*. Статистички значајно подобрување има за 4 евалуациски мерки, статистички незначајно подобрување за 11 евалуациски мерки а незначително влошување на перформансите има само по 3 евалуациски мерки. Во однос на мерките базирани на примерок, статистички значајно се подобрени перформансите за евалуациските мерки *odsv*, *точност* и  $F_1$ . За мерките базирани на ознака нема значајни промени. *RF-PCT* единствено е малку подобар на *макро* мерките базирани на ознака.

Второ, перформансите на *RF-HHPCT* и *HOMER* статистички значајно се разликуваат за скоро сите евалуациски мерки. Подобрени се перформансите на *RF-HHPCT* во однос на *Хаминг загуба* и евалуациските мерки кои ја мерат *прецизноста*. Притоа намалена е и разликата во однос на *odsv* и *точност* евалуациските мерки.

Трето, перформансите на рангирањето кај *RF-HHPCT* се особено подобрени во споредба со *HOMER* и со *RF-PCT*. Според *Вилкоксон тестот*, *RF-HHPCT* е статистички значајно подобар од *HOMER* за сите евалуациски мерки базирани на рангирање (*една грешка*, *опфатеност*, *загуба при рангирање* и *просечна прецизност*). Подобар е и од *RF-PCT* за евалуациската мерка *опфатеност*. Тоа значи дека дефинирањето на хиерархијата со алгоритмот *балансиран  $k$ -средина* позитивно влијае на перформансите на рангирање.

Последниот заклучок е доста интересен и се однесува на перформансите на *RF-HHPCT* на податочните множества со поголем број на ознаки. Имено, ако добро се анализираат перформансите на трите методи, може да се забележи дека за помалите податочни множества перформансите на *HOMER* се подобри. Меѓутоа, од друга страна, перформансите на дрвата за одлучување се значително подобри на податочните множества со поголем број на ознаки. Доминацијата станува уште повеќе очигледна кога на дрвата за предиктивно кластерирање како дополнителен влез ќе им биде зададена и хиерархија која претходно е дефинирана со алгоритмот *балансиран  $k$ -средина* (*RF-HHPCT*). Тоа значи дека групирањето на големиот број на ознаки во хиерархиски дефинирани кластери ја зголемува можноста за поточно конструирање на дрвата за предиктивно кластерирање и овозможува зголемена прецизност на нивната предикција.

**Табела 8.4** Перформанси на методите (*HOMER*, *RF-PCT* и *RF-HHPCT*) во однос на евалуациските мерки базирани на примерок (DNF – *Did Not Finish* кај *HOMER* методот означува дека тој не конструирал предиктивен модел во временски период од една недела на расположливите ресурси)

		emotions	scene	yeast	medical	enron	corel5k	tmc2007	mediamill	bibtex	delicious	bookmarks
<i>Натп. забуба</i>	<i>HOMER</i>	0.361	0.082	0.207	<b>0.012</b>	0.051	0.012	0.015	0.038	0.014	0.022	DNF
	<i>RF-PCT</i>	<b>0.189</b>	0.094	0.197	0.014	<b>0.046</b>	<b>0.009</b>	<b>0.011</b>	<b>0.029</b>	0.013	<b>0.018</b>	0.009
	<i>RF-HHPCT</i>	<b>0.189</b>	<b>0.080</b>	<b>0.195</b>	<b>0.012</b>	<b>0.046</b>	<b>0.009</b>	0.015	<b>0.029</b>	<b>0.012</b>	<b>0.018</b>	<b>0.008</b>
<i>точн.</i>	<i>HOMER</i>	0.471	0.717	0.559	0.713	<b>0.478</b>	<b>0.179</b>	0.888	0.413	<b>0.330</b>	<b>0.207</b>	DNF
	<i>RF-PCT</i>	0.519	0.541	<b>0.478</b>	0.591	0.416	0.009	0.914	<b>0.441</b>	0.166	0.146	0.204
	<i>RF-HHPCT</i>	<b>0.520</b>	<b>0.555</b>	0.475	<b>0.715</b>	0.426	0.023	<b>0.916</b>	0.437	0.256	0.138	<b>0.209</b>
<i>прец.</i>	<i>HOMER</i>	0.509	<b>0.746</b>	0.663	<b>0.762</b>	0.616	<b>0.317</b>	0.926	0.597	<b>0.472</b>	0.369	DNF
	<i>RF-PCT</i>	<b>0.644</b>	0.565	0.744	0.635	0.709	0.030	<b>0.977</b>	0.772	0.292	<b>0.512</b>	0.218
	<i>RF-HHPCT</i>	0.637	0.555	<b>0.747</b>	0.731	<b>0.713</b>	0.071	0.966	<b>0.774</b>	0.423	<b>0.512</b>	<b>0.225</b>
<i>одсue</i>	<i>HOMER</i>	<b>0.775</b>	<b>0.744</b>	<b>0.714</b>	<b>0.760</b>	<b>0.610</b>	<b>0.250</b>	<b>0.943</b>	<b>0.563</b>	<b>0.389</b>	<b>0.303</b>	DNF
	<i>RF-PCT</i>	0.582	0.541	0.523	0.599	0.452	0.009	0.920	0.476	0.167	0.160	0.208
	<i>RF-HHPCT</i>	0.588	0.623	0.645	0.690	0.540	0.024	0.885	0.488	0.260	0.171	<b>0.214</b>
<i>клас. точн.</i>	<i>HOMER</i>	0.163	<b>0.661</b>	<b>0.213</b>	0.610	<b>0.145</b>	<b>0.002</b>	0.765	0.053	<b>0.165</b>	0.001	DNF
	<i>RF-PCT</i>	0.307	0.518	0.152	0.538	0.131	0.000	<b>0.816</b>	0.122	0.098	<b>0.007</b>	0.189
	<i>RF-HHPCT</i>	<b>0.308</b>	0.508	0.182	<b>0.622</b>	0.136	0.000	0.748	0.128	<b>0.165</b>	<b>0.007</b>	<b>0.193</b>
<i>F<sub>1</sub></i>	<i>HOMER</i>	<b>0.614</b>	<b>0.745</b>	0.687	<b>0.761</b>	0.613	<b>0.280</b>	0.934	0.579	<b>0.426</b>	<b>0.343</b>	DNF
	<i>RF-PCT</i>	0.611	0.553	0.614	0.616	0.552	0.014	<b>0.948</b>	0.589	0.212	0.244	0.213
	<i>RF-HHPCT</i>	0.612	0.587	<b>0.692</b>	0.710	<b>0.615</b>	0.036	0.924	<b>0.599</b>	0.322	0.256	<b>0.219</b>

Табела 8.5 Перформанси на методите (*HOMER*, *RF-PCT* и *RF-HHPCT*) во однос на евалуациските мерки базирани на примерок (DNF – *Did Not Finish* кај *HOMER* методот означува дека тој не конструирал предиктивен модел во временски период од една недела на расположливите ресурси)

		emotions	scene	yeast	medical	enron	core5k	tmc2007	mediamill	bibtex	delicious	bookmarks
мик прец.	HOMER	0.471	0.804	0.647	0.807	0.597	0.308	0.922	0.569	0.547	0.396	DNF
	RF-PCT	0.783	0.930	<b>0.755</b>	0.885	0.738	0.696	<b>0.992</b>	0.798	<b>0.957</b>	0.695	<b>0.895</b>
	RF-HHPCT	<b>0.785</b>	<b>0.932</b>	0.749	<b>0.895</b>	<b>0.740</b>	<b>0.699</b>	0.987	<b>0.800</b>	0.912	<b>0.709</b>	0.885
мак. прец.	HOMER	0.464	0.807	0.471	0.287	0.241	<b>0.044</b>	0.954	0.107	<b>0.391</b>	0.154	DNF
	RF-PCT	<b>0.802</b>	<b>0.919</b>	<b>0.674</b>	0.269	0.233	0.015	<b>0.997</b>	0.441	0.127	<b>0.293</b>	<b>0.522</b>
	RF-HHPCT	0.801	0.911	0.580	<b>0.272</b>	<b>0.246</b>	0.028	0.995	<b>0.455</b>	0.324	0.252	0.496
мик одбив.	HOMER	<b>0.782</b>	<b>0.727</b>	<b>0.702</b>	<b>0.742</b>	<b>0.585</b>	<b>0.248</b>	<b>0.932</b>	<b>0.537</b>	<b>0.353</b>	<b>0.297</b>	DNF
	RF-PCT	0.589	0.523	0.521	0.569	0.422	0.009	0.902	0.435	0.131	0.151	0.136
	RF-HHPCT	0.569	0.514	0.525	0.651	0.431	0.024	0.896	0.427	0.203	0.140	<b>0.139</b>
мак. одбив.	HOMER	<b>0.775</b>	<b>0.734</b>	<b>0.466</b>	<b>0.282</b>	<b>0.163</b>	<b>0.041</b>	<b>0.897</b>	0.074	<b>0.247</b>	<b>0.103</b>	DNF
	RF-PCT	0.569	0.533	0.286	0.176	0.100	0.002	0.769	0.080	0.043	0.060	<b>0.072</b>
	RF-HHPCT	0.550	0.525	0.285	0.218	0.104	0.004	0.678	<b>0.081</b>	0.104	0.049	0.071
мик. $F_1$	HOMER	0.588	0.764	0.673	<b>0.773</b>	<b>0.591</b>	<b>0.275</b>	0.927	0.553	<b>0.429</b>	<b>0.339</b>	DNF
	RF-PCT	<b>0.672</b>	<b>0.669</b>	<b>0.617</b>	0.693	0.537	0.018	<b>0.945</b>	<b>0.563</b>	0.230	0.248	0.236
	RF-HHPCT	0.660	0.663	<b>0.617</b>	0.754	0.545	0.046	0.939	0.557	0.332	0.234	<b>0.240</b>
мак. $F_1$	HOMER	0.570	<b>0.768</b>	<b>0.447</b>	<b>0.282</b>	<b>0.167</b>	<b>0.036</b>	<b>0.924</b>	0.073	<b>0.266</b>	<b>0.103</b>	DNF
	RF-PCT	<b>0.650</b>	0.658	0.322	0.207	0.122	0.004	0.857	0.112	0.055	0.083	<b>0.101</b>
	RF-HHPCT	0.628	0.648	0.321	0.237	0.127	0.007	0.780	<b>0.115</b>	0.134	0.068	0.099

Табела 8.6 Перформанси на методите (*HOMER*, *RF-PCT* и *RF-HHPCT*) во однос на евалуациските мерки базирани на рангирање (DNF – *Did Not Finish* кај *HOMER* методот означува дека тој не конструирал предиктивен модел во временски период од една недела на расположливите ресурси)

		emotions	scene	yeast	medical	enron	core5k	tmc2007	mediamill	bibtex	delicious	bookmarks
една грешка	HOMER	0.411	0.216	0.248	0.216	0.314	0.652	0.050	0.219	0.466	0.509	DNF
	RF-PCT	<b>0.262</b>	<b>0.210</b>	0.248	0.174	0.221	<b>0.608</b>	<b>0.006</b>	<b>0.159</b>	0.433	0.332	0.541
	RF-HHPCT	0.272	0.225	<b>0.239</b>	<b>0.150</b>	<b>0.219</b>	0.686	0.011	0.162	<b>0.375</b>	<b>0.328</b>	<b>0.534</b>
опфат.	HOMER	2.634	0.739	7.285	5.324	24.190	250.800	2.369	47.046	65.626	933.956	DNF
	RF-PCT	<b>1.827</b>	<b>0.461</b>	6.179	1.619	12.074	107.412	<b>1.219</b>	16.926	25.854	504.999	34.185
	RF-HHPCT	1.866	0.495	<b>6.177</b>	<b>1.402</b>	<b>11.668</b>	<b>101.930</b>	1.235	<b>16.824</b>	<b>20.870</b>	<b>488.441</b>	<b>34.122</b>
загуба ранг.	HOMER	0.297	0.119	0.205	0.090	0.183	0.352	0.028	0.177	0.255	0.379	DNF
	RF-PCT	<b>0.151</b>	<b>0.072</b>	0.167	0.024	0.079	0.117	0.006	<b>0.047</b>	0.093	0.106	0.104
	RF-HHPCT	0.163	0.079	<b>0.165</b>	<b>0.019</b>	<b>0.077</b>	<b>0.114</b>	<b>0.001</b>	<b>0.047</b>	<b>0.072</b>	<b>0.102</b>	<b>0.103</b>
прос. прец.	HOMER	0.698	0.848	0.740	0.786	0.604	0.222	0.945	0.583	0.407	0.231	DNF
	RF-PCT	<b>0.812</b>	<b>0.874</b>	0.757	0.868	0.698	<b>0.334</b>	<b>0.996</b>	<b>0.737</b>	0.525	0.395	0.480
	RF-HHPCT	0.805	0.864	<b>0.760</b>	<b>0.887</b>	<b>0.700</b>	0.313	0.992	0.735	<b>0.579</b>	<b>0.398</b>	<b>0.481</b>

**Табела 8.7** Перформанси на методите (*HOMER*, *RF-PCT* и *RF-HHPCT*) во однос на времињата на обучување и тестирање (DNF – *Did Not Finish* кај *HOMER* методот означува дека тој не конструирал предиктивен модел во временски период од една недела на расположливите ресурси)

		emotions	scene	yeast	medical	enron	corel5k	tmc2007	mediamill	bibtex	delicious	bookmarks
време обуч.	<b>HOMER</b>	4.0	68.0	101.0	<b>16.0</b>	158.0	<b>771.0</b>	31300.0	78195.0	2896.0	21218.0	DNF
	<b>RF-PCT</b>	<b>2.9</b>	<b>23.0</b>	<b>25.0</b>	27.0	<b>47.0</b>	902.0	<b>557.0</b>	<b>8360.0</b>	<b>1550.0</b>	<b>5376.0</b>	<b>28900.0</b>
	<b>RF-HHPCT</b>	3.1	26.0	29.0	28.0	51.0	921.0	602.0	9224.0	1682.0	5628.0	37654.0
време тест.	<b>HOMER</b>	1.0	21.0	17.0	1.5	22.0	14.0	730.0	6079.0	155.0	816.0	DNF
	<b>RF-PCT</b>	<b>0.3</b>	1.0	<b>0.2</b>	<b>0.5</b>	<b>1.0</b>	2.5	2.8	<b>4.0</b>	<b>18.0</b>	48.0	58.0
	<b>RF-HHPCT</b>	<b>0.3</b>	<b>0.9</b>	<b>0.2</b>	<b>0.5</b>	1.1	<b>2.4</b>	<b>2.6</b>	4.2	19.0	<b>44.0</b>	<b>57.0</b>

**Табела 8.8** Ниво на значајност  $p$  според *Вилкоксон тестот* за секоја евалуациска мерка во однос на перформансите на методите *RF-HHPCT*, *RF-PCT* и *HOMER*. Знакот  $>$  покажува дека првиот метод е статистички подобар од вториот (прв метод vs втор метод), додека знакот  $<$  покажува дека вториот метод е статистички подобар од првиот. Вредностите на  $p$  помали од 0.05 покажуваат дека меѓу двата споредувани метода постои статистички значајна разлика

	<i>RF-HHPCT</i> vs <i>RF-PCT</i>	<i>RF-HHPCT</i> vs <i>HOMER</i>
<i>Хаминг загуба</i>	0.130 $>$	<b>0.008</b> $>$
<i>точност</i>	<b>0.007</b> $>$	0.241 $<$
<i>прецизност</i>	0.266 $>$	0.646 $>$
<i>odsuv</i>	<b>0.016</b> $>$	<b>0.005</b> $<$
$F_1$	<b>0.016</b> $>$	<b>0.006</b> $<$
<i>класификациска точност</i>	0.197 $>$	0.799 $<$
<i>микро прецизност</i>	0.964 $>$	<b>0.005</b> $>$
<i>микро odsuv</i>	0.689 $>$	<b>0.005</b> $<$
<i>микро <math>F_1</math></i>	0.593 $>$	0.060 $<$
<i>макро прецизност</i>	0.929 $<$	0.074 $>$
<i>макро odsuv</i>	0.789 $<$	<b>0.006</b> $<$
<i>макро <math>F_1</math></i>	0.929 $<$	<b>0.007</b> $<$
<i>една грешка</i>	0.929 $>$	<b>0.019</b> $>$
<i>опфатеност</i>	<b>0.033</b> $>$	<b>0.005</b> $>$
<i>загуба при рангирање</i>	0.210 $>$	<b>0.005</b> $>$
<i>просечна прецизност</i>	0.969 $>$	<b>0.005</b> $>$
<i>време на обучување</i>	<b>0.003</b> $>$	<b>0.046</b> $>$
<i>време на тестирање</i>	0.477 $>$	<b>0.005</b> $>$

Единствен исклучок од ова е *corel5k* податочното множество, кое иако има 384 ознаки, сепак се издвојува од останатите податочни множества поради малиот број на примероци по ознака. Оваа карактеристика ја лоциравме како една од причините,

зошто хиерархијата не влијае позитивно на перформансите на ансамблиите од дрва за предиктивно кластерирање во овој случај.

Дополнително интересен е фактот дека *RF-PCT* има најдобри перформанси за *tmc2007* податочното множество. Во најголем број од евалуациските мерки овој метод е значително подобар од *HOMER* методот. *HOMER* има подобри перформанси единствено за мерките кои мерат *odsv* и *макро F<sub>1</sub>*. Во споредба со *RF-HHPCT*, *RF-PCT* има споредливи или за нијанса подобри резултати. *RF-HHPCT* покажа подобри резултати само за евалуациската мерка *загуба при рангирање*.

## 8.5 Заклучок

Во оваа глава беше предложена интеграција на два метода: *HOMER* и дрвата за предиктивно кластерирање за хиерархиска повеќезначна класификација. Имено, по организирањето на ознаките (предикциите) на секој од примероците во согласност со вештачката хиерархија дефинирана со алгоритамот *балансиран к-средина*, хиерархискиот повеќезначен класификациски проблем добиен со соодветна трансформација на оригиналниот повеќезначен класификациски проблем го решивме со *PCTs* за хиерархиска повеќезначна класификација. На тој начин, ново дефинираната хиерархија над податочното множество влијае на процесот на дефинирање на кластерите во секој од јазлите на дрвото (*PCT*). Добиените предикции со соодветна трансформација се пресликуваат во предикции релевантни за оригиналниот повеќезначен проблем.

Во експериментална евалуација беше направена споредба на предложениот метод со двата метода кои тој ги интегрира и кои според обемната експериментална евалуација покажаа просечно најдобри перформанси. Грите методи ги споредивме на 11 повеќезначни класификациски проблеми од 5 апликациски области (класификација на текст, класификација на слики предикција на функција на гени, класификација на музика и класификација на видео) во однос на 18 различни евалуациски мерки (6 мерки базирани на ознака, 6 мерки базирани на примерок, 4 мерки базирани на рангирање и 2 мерки за ефикасноста на методите). За да ја процениме статистичката значајност на разликата на перформансите на споредуваните методи го користевме *Вилкоксон тестот*.

Врз основа на резултатите добиени од експериментите и статистичката евалуација донесовме три генерални заклучоци.

Разликата меѓу перформансите на различните типови на методи во однос на мерките кои го мерат *одзивот* на методите и мерките кои ја мерат нивната *прецизност* е значително намалена со интеграцијата на *HOMER* и дрвата за предиктивно кластерирање. Тоа го потврдува и *Вилкоксон тестот* која покажа дека *RF-HHPCT* е статистички значајно подобар од *RF-PCT* за *odsv* евалуациската мерка (базирана на примерок). Анализата покажа дека просечно перформансите на *RF-HHPCT* се подобри во однос на *RF-PCT* во однос на сите евалуациски мерки, освен за *макро* мерките базирани на ознака.

Перформансите на *RF-HHPCT* се исклучително подобри на податочните множества со поголем број на ознаки во споредба со перформансите на *HOMER* и *RF-PCT*. Тоа значи дека групирањето на големиот број на ознаки во хиерархиски дефинирани

кластери ја зголемува можноста за поточно конструирање на дрвата за предиктивно кластерирање и овозможува зголемена прецизност на нивната предикција.

На крајот како генерален заклучок може да се донесе дека дефинирањето на вештачка хиерархијата на целните обележја и трансформацијата на повеќезначниот класификациски проблем во хиерархиски повеќезначен класификациски проблем позитивно влијае на перформансите во однос на сите евалуациски мерки. Особено значајно е подобрувањето на перформансите во однос на евалуациските мерки базирани на рангирање и во однос на податочните множества со голем број на ознаки.

Во идните истражувања ќе бидат разгледани различни алгоритми за хиерархиско кластерирање, со цел да се покаже дали дефинирањето на било каква вештачка хиерархија над оригиналните повеќезначни класификациски проблеми би ги подобрила перформансите што ги постигнуваат на нив конвенционалните повеќезначни класификатори. Дополнително, експерименталната евалуација ќе биде проширена со класификациски проблеми чии ознаки се оригинално организирани во хиерархиска структура со цел да истражи сличноста на оригиналната хиерархија и експериментално дефинираната хиерархија на ознаките и да се потврди хипотезата која претходно ја поставивме.

## Глава 9

# Комплетна споредба на методите за повеќезначна класификација и рангирање

Во оваа глава се споредени перформансите на предложените методи и методите за повеќезначна класификација и рангирање споредени во Глава 5. Споредбата е направена во однос на 18-те евалуациски мерки вклучувајќи ги и времињата на обучување и тестирање на 11-те бенчмарк повеќезначни класификациски проблеми. Експерименталниот дизајн и условите под кои се споредени методите за повеќезначна класификација и рангирање се опишани во Глава 5 (12-те конкурентни методи), Глава 6 (дво-нивовската архитектура), Глава 7 (хибридната архитектура) и Глава 8 (интеграцијата на *HOMER* и *PCTs*).

Во Додаток А (Табела А.1 до Табела А.18) се прикажани комплетните резултати на сите методи за повеќезначна класификација и рангирање за секоја евалуациска мерка посебно. Во првата колона од овие табели се прикажани имињата на споредуваните методи, а останатите колони ги прикажуваат резултатите добиени од експериментите, по една колона за секое од податочните множества. Дополнително, за да се процени статистичката значајност на разликата на перформансите на споредуваните методи користен е *коригиран Фридман тест* [72] и *дополнителен Немењи тест* [73] како што е предложено во истражувањето на Demšar [74].

На Слика А.1 до Слика А.12 се претставени резултатите од *дополнителниот Немењи тест* преку дијаграмите на просечни рангови [74]. Критичните дијаграми содржат нумерирана оска на која се означени просечните рангови на методите. Најдобрите методи се нанесени на десната страна на дијаграмите. Ако разликата на пресечните рангови на методите е статистички незначајна (на ниво на значајност  $p = 0.05$ ), линиите кои ги означуваат ранговите се поврзани со хоризонтална линија.

Некои методи не генерираа модел за предвидување за големите податочни множества за време од една недела на ресурсите кои ги имавме на располагање. Во табелите со резултати наместо постигнатите перформанси за овие методи е запишано **DNF (Did Not Finish)**. Имајќи го ова во предвид, статистичката анализа беше направена два пати. Првиот пат во анализата се вклучени само оние податочни множества (8 податочни множества) за кои сите методи генерирале модел и дале резултати (Слика А.1 до Слика А.6). Во втората анализа (Слика А.7 до Слика А.12) вклучени се сите податочни множества, а оние методи кои не генерирале модел за конкретното податочно множество се казнети со доделување на најслабиот ранг во однос на сите евалуациски мерки.

Врз основа на добиените резултати може да се забележи дека методите предложени во дисертацијата за сите евалуациски мерки се наоѓаат во првата половина на методи според нивниот ранг. За евалуациските мерки базирани на примерок доминираат методите  $ML-SVMDT_{HL}$ ,  $RF-HHPCT$  и  $HOMER$ , а веднаш зад нив се  $TSCCM$ ,  $TSPCCM$ ,  $BR$  и верижниот метод  $CC$ .

Слично како и кај евалуациските мерки базирани на примерок, така и кај мерките базирани на ознака најдобри перформанси во однос на евалуациските мерки што мерат прецизност покажаа  $RF-HHPCT$ ,  $RF-PCT$  и хибридната архитектура, додека  $HOMER$  и дво-нивовската верижна архитектура покажаа најдобри резултати во однос на *одсвот*. Најдобар баланс на перформансите покажа предложената хибридна архитектура, особено методот кој како валидациска функција користи *Хаминг загуба*.

Во однос на рангирањето, најдобар рангирачки алгоритам е  $RF-HHPCT$ . По него следуваат методите  $RF-PCT$  и дво-нивовските верижни архитектури. Од анализата на перформансите на методите во однос на времињата на обучување и тестирање, може да се заклучи дека методите базирани на дрвата за одлучување имаат значително повисока ефикасност од  $SVM$ -базираните методи. Имено,  $PCT$  и  $ML-C4.5$  се најефикасните методи, а по нив следуваат ансамблиите од дрва за одлучување ( $RF-PCT$  и  $RFML-C4.5$ ), хибридната архитектура базирана на дрвата за одлучување и  $ML-kNN$  методот.

Од критичните дијаграми на податочните множества за кои сите методи дале резултати, лесно може да се забележи дека две групи на методи јасно се издвојуваат. Во првата група се методите  $RF-HHPCT$ ,  $ML-SVMDT_{HL}$ ,  $TSCCM$ ,  $TSPCCM$ ,  $RF-PCT$ ,  $TSVM$ ,  $HOMER$ ,  $BR$ ,  $CC$  и  $ML-SVMDT_{FI}$ , додека во втората група се сите останати методи кои ги споредуваме. Методите од првата група имаат подобри перформанси во однос на сите перформансни евалуациски мерки. Од друга страна, методите базирани на дрвата за одлучување од втората група имаат најдобри времиња на обучување и тестирање. *Коригираниот Фридман тест* за определување на статистичката значајност на разликата во перформансите на методите е доста конзервативен и не покажува статистички значајна разлика меѓу перформансите на методите кои припаѓаат на иста група. Но, најчесто најдобриот метод од првата група е статистички значајно подобар од неколку (повеќето) методи од втората група (на пример  $RF-HHPCT$  за евалуациската мерка *загуба при рангирање* или *микро прецизност*).

Во однос на перформансите на различните типови на алгоритми за машинско учење, резултатите покажуваат дека  $SVM$ -базираните методи (особено дво-нивовската архитектура) имаат најдобри перформанси за малите податочни множества и податочните множества со поголем број на обележја, додека методите базирани на дрвата за одлучување најдобри перформанси имаат за големите податочни множества. Од методите базирани на дрвата за одлучување се издвојуваат методите ансамбли ( $RF-HHPCT$ ,  $RF-PCT$  и  $RFML-C4.5$ ) и хибридната архитектура која користи локални машини со носечки вектори.

Слични заклучоци можат да се донесат ако се погледнат и резултатите добиени од статистичката евалуација на методите за сите податочни множества. Имено, овие дијаграми (Слика А.7 до Слика А.12) уште повеќе ја истакнуваат разликата која постои меѓу  $SVM$ -базираните методи и методите базирани на дрвата за одлучување. Во оваа евалуација просечно најдобри перформанси покажаа методите ансамбли  $RF-HHPCT$ ,  $RF-PCT$  и хибридната архитектура.

## Глава 10

### Заклучок

Во оваа докторска дисертација е даден преглед на областа на повеќезначна класификација и рангирање. Анализата е направена од повеќе аспекти: дефинирање и поставување на проблемот на повеќезначна класификација и рангирање, категоризација на постоечките методи во оваа област, опис на методи за градење на системи за повеќезначна класификација и рангирање и нивна експериментална евалуација, анализа на добиените резултати и статистичка евалуација на повеќезначните предикции.

Конкретно, прво е даден широк преглед на областа на повеќезначна класификација и рангирање. Во овој преглед се опишани и референцирани голем број од постоечките техники и методи за повеќезначна класификација и рангирање кои се предложени како можни пристапи за решавање на проблеми од различни апликациски области во изминатите неколку години. Анализирајќи ги карактеристиките на различните методи како и начините според кои тие ги решаваат повеќезначните класификациски проблеми, предложена е нова онтологија според која методите се групираат во три основни категории: методи кои го трансформираат проблемот, методи кои го адаптираат алгоритмот и методи кои градат ансамбли. Во оваа онтологија како ново дефинирана категорија се појавува категоријата на методите кои градат ансамбли кои сами по себе се издвојуваат според своите карактеристики, репрезентацијата на проблемот и неговото решавање. Потоа, беа избрани 12-те најчесто цитирани и најзначајни методи за повеќезначна класификација и рангирање и истите беа споредени на 11 бенчмарк повеќезначни класификациски проблеми во однос на 18 евалуациски мерки.

Откако добиените резултати беа анализирани и беа увидени предностите и слабостите на споредуваните методи, беа предложени три различни методи за повеќезначна класификација и рангирање.

Прво беше предложена дво-нивовската, ефикасна парови на ознаки архитектура (дво-нивовскиот метод со гласање, дво-нивовскиот верижен метод и дво-нивовскиот скратен верижен метод) за повеќезначна класификација и рангирање која припаѓа на категоријата на методи кои го трансформираат проблемот. Предложената архитектура значително го намали времето на тестирање на методите *парови на ознаки* (*CLR* и *QWML*) и покажа значајно подобри резултати во однос на евалуациските мерки базирани на примерок, ознака и рангирање. Со воведувањето на прагот на одлука беше постигнато намалување на пресметковната комплексност во фазата на тестирање, без притоа негативно да се влијае на останатите перформанси. Воведувањето на верижната структура кај *TSCCM* и *TSPCCM*, уште повеќе ги подобри перформансите на дво-нивовската архитектура во однос на евалуациските мерки базирани на примерок, ознака и рангирање.

Меѓутоа, главен недостаток на оваа архитектура, како и на останатите методи парови на ознаки се времињата на обучување за големите податочни множества. Со цел

да го решиме овој проблем, беше предложена нова хибридна архитектура базирана на дрва за одлучување кои користат локални машини со носечки вектори за повеќезначна класификација и рангирање. Претставени беа две методи за градење на оваа архитектура ( $ML-SVMDT_{HL}$  и  $ML-SVMDT_{F1}$ ) и две модификации ( $ML-SVMDT_{pre}$  и  $ML-SVMDT_{post}$ ). За разлика од дво-нивовската *парови на ознаки* архитектура, хибридната архитектура понуди одличен баланс на перформансите во однос на предвидувањето и пресметковната ефикасност. Нејзините перформанси на предвидување најчесто се подобри од перформансите на двата метода кои оваа архитектура ги интегрира или многу слични на перформансите на подобриот од нив. Во однос на времињата на обучување и тестирање, хибридната архитектура е нешто побавна во однос на  $ML-C4.5$ , но има ефикасност сосема задоволителна за големите податочни множества. За споредба, *Вилкоксон тестот* покажа дека таа е статистички значајно побрза од глобалниот *BR* метод, кој е најбрзиот (најефикасниот) метод кој припаѓа на категоријата на методи кои го трансформираат проблемот.

На крајот беше предложен уште еден метод кој припаѓа на третата категорија на методи (методите кои градат ансамбли). Тој ги интегрира методите *HOMER* и *Random Forest* ансамблот од дрва за предиктивно кластерирање за хиерархиска повеќезначна класификација. Резултатите покажаа дека перформансите на *RF-HHPCT* просечно се подобрени во однос на *RF-PCT* за сите евалуациски мерки, освен за *макро* мерките базирани на ознака. Потоа, перформансите на *RF-HHPCT* се исклучително подобри на податочните множества со поголем број на ознаки во споредба со перформансите на методите кои тој ги интегрира. Тоа значи дека групирањето на големиот број на ознаки во хиерархиски дефинирани кластери ја зголемува можноста за поточно конструирање на дрвата за предиктивно кластерирање и овозможува зголемена прецизност на нивната предикција. На крајот генерално може да се заклучи дека дефинирањето на вештачка хиерархијата на целните обележја и трансформацијата на повеќезначниот класификациски проблем во хиерархиски повеќезначен класификациски проблем позитивно влијае на перформансите во однос на сите евалуациски мерки. Особено значајно е подобрувањето на перформансите во однос на евалуациските мерки базирани на рангирање и во однос на податочните множества со голем број на ознаки.

Врз основа на статистичката евалуација на перформансите направена за сите методи (предложените и конкурентните методи) за повеќезначна класификација и рангирање може да се забележи дека за сите евалуациски мерки предложените методи се наоѓаат во првата половина според нивниот ранг. За евалуациските мерки базирани на примерок најдобри перформанси имаат  $ML-SVMDT_{HL}$ , *RF-HHPCT* и *HOMER*, а веднаш позади нив се дво-нивовските верижни методи (*TSCCM* и *TSPCCM*), *BR* и верижниот метод *CC*.

Најдобри перформанси во однос на *прецизност* евалуациските мерки базирани на примерок, покажаа *RF-HHPCT*, *RF-PCT* и хибридната архитектура, додека *HOMER* и дво-нивовската верижна архитектура покажаа најдобри резултати во однос на *odsub* евалуациските мерки.

Во однос на рангирањето, најдобар рангирачки алгоритам е *RF-HHPCT*. *PCT* и  $ML-C4.5$  се најефикасните методи, а по нив следуваат ансамблиите од дрва за одлучување (*RF-PCT* и *RFML-C4.5*), хибридната архитектура базирана на дрвата за одлучување и *ML-kNN* методот.

Во однос на перформансите на различните типови на алгоритми за машинско учење, *SVM*-базираните методи (особено дво-нивовската архитектура) се вистинскиот избор за мали податочни множества и податочни множества со поголем број на обележја, додека методите базирани на дрвата за одлучување (*RF-HHPC*, *RF-PC* и *RFML-C4.5* и хибридната архитектура која користи локални машини со носечки вектори) се вистинскиот избор за големи податочни множества.

## Литература

- [1] C. N. Silla, Jr. and A. A. Freitas, “A survey of hierarchical classification across different application domains,” *Data Mining and Knowledge Discovery*, vol. 22, pp. 31–72, 2011.
- [2] G. Tsoumakas, I. Katakis, and I. Vlahavas, “Mining multi-label data,” in *Data Mining and Knowledge Discovery Handbook*, pp. 667–685, Springer Berlin / Heidelberg, 2010.
- [3] Y. Yang, “An evaluation of statistical approaches to text categorization,” *Information Retrieval*, vol. 1, pp. 69–90, 1999.
- [4] R. E. Schapire and Y. Singer, “Boostexter: A boosting-based system for text categorization,” *Machine Learning*, vol. 39, pp. 135–168, 2000.
- [5] S. Godbole and S. Sarawagi, “Discriminative Methods for Multi-labeled Classification,” in *Advances in Knowledge Discovery and Data Mining*, pp. 22–30, Springer Berlin / Heidelberg, 2004.
- [6] S. Zhu, X. Ji, W. Xu, and Y. Gong, “Multi-labelled classification using maximum entropy method,” in *Proc. of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 274–281, 2005.
- [7] N. Ghamrawi and A. McCallum, “Collective multi-label classification,” in *Proc. of the 14th ACM international conference on Information and knowledge management*, pp. 195–200, 2005.
- [8] M. R. Boutell, J. Luo, X. Shen, and C. M. Brown, “Learning multi-label scene classification,” *Pattern Recognition*, vol. 37, no. 9, pp. 1757–1771, 2004.
- [9] P. Duygulu, K. Barnard, J. de Freitas, and D. Forsyth, “Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary,” in *Proc. of the 7th European Conference on Computer Vision*, pp. 349–354, 2002.
- [10] A. Elisseeff and J. Weston, “A Kernel Method for Multi-Labelled Classification,” in *Proc. of the Annual ACM Conference on Research and Development in Information Retrieval*, pp. 274–281, 2005.
- [11] B. Klimt and Y. Yang, “The Enron Corpus: A New Dataset for Email Classification Research,” in *Proc. of the 15th European conference on Machine Learning*, pp. 217–226, 2004.
- [12] J. Read, B. Pfahringer, G. Holmes, and E. Frank, “Classifier chains for multi-label classification,” in *Proc. of the 20th European Conference on Machine Learning*, pp. 254–269, 2009.
- [13] I. Katakis, G. Tsoumakas, and I. Vlahavas, “Multilabel Text Classification for Automated Tag Suggestion,” in *Proc. of the ECML/PKDD Discovery Challenge*, 2008.
- [14] G. Tsoumakas, I. Katakis, and I. Vlahavas, “Effective and Efficient Multilabel Classification in Domains with Large Number of Labels,” in *Proc. of the ECML/PKDD Workshop on Mining Multidimensional Data*, pp. 30–44, 2008.
- [15] A. Srivastava and B. Zane-Ulman, “Discovering recurring anomalies in text reports regarding complex space systems,” in *Proc. of the IEEE Aerospace Conference*, pp. 55–63, 2005.

- [16] K. Trohidis, G. Tsoumakas, G. Kalliris, and I. Vlahavas, “Multilabel Classification of Music into Emotions,” in *Proc. of the 9th International Conference on Music Information Retrieval*, pp. 320–330, 2008.
- [17] C. G. M. Snoek, M. Worring, J. C. van Gemert, J.-M. Geusebroek, and A. W. M. Smeulders, “The challenge problem for automated detection of 101 semantic concepts in multimedia,” in *Proc. of the 14th Annual ACM International Conference on Multimedia*, pp. 421–430, 2006.
- [18] G. Tsoumakas and I. Vlahavas, “Random k-labelsets: An ensemble method for multilabel classification,” in *Proc. of the 18th European conference on Machine Learning*, pp. 406–417, 2007.
- [19] G. Tsoumakas and I. Katakis, “Multi Label Classification: An Overview,” *International Journal of Data Warehouse and Mining*, vol. 3, no. 3, pp. 1–13, 2007.
- [20] E. L. Mencía, S.-H. Park, and J. Fürnkranz, “Efficient voting prediction for pairwise multilabel classification,” *Neurocomputing*, vol. 73, pp. 1164–1176, 2010.
- [21] W. Chen, J. Yan, B. Zhang, Z. Chen, and Q. Yang, “Document transformation for multi-label feature selection in text categorization,” in *Proc. of the 7-th IEEE International Conference on Data Mining*, pp. 451–456, 2007.
- [22] J. Read, “A Pruned Problem Transformation Method for Multi-label classification,” in *Proc. of the New Zealand Computer Science Research Student Conference*, pp. 143–150, 2008.
- [23] G. Tsoumakas, I. Katakis, and I. Vlahavas, “Random k-labelsets for multilabel classification,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 23, pp. 1079–1089, 2011.
- [24] G. Tsoumakas, A. Dimou, and E. Spyromitros, *Correlation-Based Pruning of Stacked Binary Relevance Models for Multi-Label Learning*, pp. 101–116. 2009.
- [25] R. Yan, J. Tesic, and J. R. Smith, “Model-shared subspace boosting for multi-label classification,” in *Proc. of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 834–843, 2007.
- [26] S. Ji, L. Tang, S. Yu, and J. Ye, “Extracting shared subspace for multi-label classification,” in *Proc of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 381–389, 2008.
- [27] A. Ráez, L. López, and R. Steinberger, “Adaptive selection of base classifiers in one-against-all learning for large multi-labeled collections,” in *Advances in Natural Language Processing* (J. Vicedo, P. Martínez-Barco, R. Muñoz, and M. Saiz Noeda, eds.), vol. 3230 of *Lecture Notes in Computer Science*, pp. 1–12, 2004.
- [28] E. Hüllermeier, J. Fürnkranz, W. Cheng, and K. Brinker, “Label ranking by learning pairwise preferences,” *Artificial Intelligence.*, vol. 172, pp. 1897–1916, 2008.
- [29] J. Fürnkranz, “Round robin classification,” *Journal of Machine Learning Research*, vol. 2, pp. 721–747, 2002.
- [30] T.-F. Wu, C.-J. Lin, and R. C. Weng, “Probability estimates for multi-class classification by pairwise coupling,” *Journal of Machine Learning Research*, vol. 5, pp. 975–1005, 2004.

- [31] K. Brinker, J. Fürnkranz, and E. Hüllermeier, “A unified model for multilabel classification and ranking,” in *Proc. of the 17th European Conference on Artificial Intelligence*, pp. 489–493, 2006.
- [32] S.-H. Park and J. Fürnkranz, “Efficient pairwise classification,” in *Proc. of the 18th European Conference on Machine Learning*, pp. 658–665, 2007.
- [33] M. L. Zhang and Z. H. Zhou, “Multi-label learning by instance differentiation,” in *Proc. of the 22nd national conference on Artificial intelligence*, pp. 669–674, 2007.
- [34] Z. H. Zhou and M. L. Zhang, “Multi-instance multilabel learning with application to scene classification,” in *Advances in Neural Information Processing Systems*, vol. 19, pp. 1609–1616, 2007.
- [35] Z. Barutcuoglu, R. E. Schapire, and O. G. Troyanskaya, “Hierarchical multi-label prediction of gene function,” *Bioinformatics*, vol. 22, pp. 830–836, 2006.
- [36] Y. Freund and R. E. Schapire, “A decision-theoretic generalization of on-line learning and an application to boosting,” *Journal of Computer and System Sciences*, vol. 55, no. 1, pp. 119 – 139, 1997.
- [37] F. De Comité, R. Gilleron, and M. Tommasi, “Learning multi-label alternating decision trees from texts and data,” in *Proc. of the 3rd international conference on Machine learning and data mining in pattern recognition*, pp. 35–49, 2003.
- [38] M. L. Zhang and Z. H. Zhou, “Multi-label neural networks with applications to functional genomics and text categorization,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 18, no. 10, pp. 1338–1351, 2006.
- [39] K. Crammer and Y. Singer, “A family of additive online algorithms for category ranking,” *Journal of Machine Learning Research*, vol. 3, pp. 1025–1058, 2003.
- [40] X. Luo and Z. A. N. Heywood, “Evaluation of Two Systems on Multi-class Multi-label Document Classification,” in *Proc. of the 15th International Symposium on Methodologies for Intelligent Systems*, pp. 161–169, 2005.
- [41] A. Wieczorkowska, P. Synak, and Z. Ras, “Multi-label classification of emotions in music,” in *Intelligent Information Processing and Web Mining*, pp. 307–315, Springer Berlin / Heidelberg, 2006.
- [42] K. Brinker and E. Hüllermeier, “Case-based multilabel ranking,” in *Proc. of the 20th international joint conference on Artificial intelligence*, pp. 702–707, 2007.
- [43] M. L. Zhang and Z. H. Zhou, “MI-knn: A lazy learning approach to multi-label learning,” *Pattern Recognition*, vol. 40, no. 7, pp. 2038–2048, 2007.
- [44] E. Spyromitros, G. Tsoumakas, and I. Vlahavas, “An empirical study of lazy multilabel classification algorithms,” in *Proc. of the 5th Hellenic conference on Artificial Intelligence: Theories, Models and Applications*, pp. 401–406, 2008.
- [45] A. Veloso, W. Meira, Jr., M. Gonçalves, and M. Zaki, “Multi-label lazy associative classification,” in *Proc. of the 11th European conference on Principles and Practice of Knowledge Discovery in Databases*, pp. 605–612, 2007.
- [46] N. Cesa-Bianchi, C. Gentile, and L. Zaniboni, “Incremental algorithms for hierarchical classification,” *Journal of Machine Learning Research*, vol. 7, pp. 31–54, 2006.

- [47] N. Cesa-Bianchi, C. Gentile, and L. Zaniboni, “Hierarchical classification: combining bayes with svm,” in *Proc. of the 23rd international conference on Machine learning*, pp. 177–184, 2006.
- [48] F. A. Thabtah, P. Cowling, and Y. Peng, “MMAC: A New Multi-class, Multi-label Associative Classification Approach,” in *Proc. of the 4th IEEE International Conference on Data Mining*, pp. 217–224, 2004.
- [49] L. Hansen and P. Salamon, “Neural network ensembles,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, pp. 993–1001, 1990.
- [50] A. Al-Ani and M. Deriche, “A new technique for combining multiple classifiers using the dempster-shafer theory of evidence,” *J. Artif. Int. Res.*, vol. 17, pp. 333–361, 2002.
- [51] L. Breiman, “Bagging predictors,” *Mach. Learn.*, vol. 24, pp. 123–140, 1996.
- [52] Y. Freund and R. E. Schapire, “Experiments with a New Boosting Algorithm,” in *Proc. of the 13th International Conference on Machine Learning*, pp. 148–156, 1996.
- [53] L. Breiman, “Using adaptive bagging to debias regressions,” tech. rep., Statistics Department, University of California, Berkeley, 1999.
- [54] T. G. Dietterich, “Ensemble methods in machine learning,” in *Proc. of the 1st International Workshop on Multiple Classifier Systems*, pp. 1–15, 2000.
- [55] J. Read, B. Pfahringer, and G. Holmes, “Multi-label Classification Using Ensembles of Pruned Sets,” in *Proc. of the 8th IEEE International Conference on Data Mining*, pp. 995–1000, 2008.
- [56] D. Kocev, C. Vens, J. Struyf, and S. Džeroski, “Ensembles of multi-objective decision trees,” in *Proc. of the 18th European conference on Machine Learning*, pp. 624–631, 2007.
- [57] D. Kocev, *Ensembles for predicting structured outputs*. PhD thesis, IPS Jožef Stefan, Ljubljana, Slovenia, 2011.
- [58] J. Fürnkranz, E. Hullermeier, E. L. Mencía, and K. Brinker, “Multi-label classification via calibrated label ranking,” *Machine Learning*, vol. 73, no. 2, pp. 133–153, 2008.
- [59] A. Clare and R. D. King, “Knowledge discovery in multi-label phenotype data,” in *Proc. of the 5th European Conference on PKDD*, pp. 42–53, 2001.
- [60] A. Clare, *Machine learning and data mining for yeast functional genomics*. PhD thesis, University of Wales Aberystwyth, Aberystwyth, Wales, UK, 2003.
- [61] J. R. Quinlan, *C4.5: Programs for Machine Learning (Morgan Kaufmann Series in Machine Learning)*. Morgan Kaufmann, 1 ed., 1992.
- [62] H. Blockeel and J. Struyf, “Efficient algorithms for decision tree cross-validation,” *Journal of Machine Learning Research*, vol. 3, pp. 621–650, 2003.
- [63] C. Vens, J. Struyf, L. Schietgat, S. Džeroski, and H. Blockeel, “Decision trees for hierarchical multi-label classification,” *Machine Learning*, vol. 73, no. 2, pp. 185–214, 2008.
- [64] I. Slavkov, V. Gjorgjioski, J. Struyf, and S. Džeroski, “Finding explained groups of time-course gene expression profiles with predictive clustering trees,” *Molecular BioSystems*, vol. 6, pp. 729–740, 2010.

- [65] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone, *Classification and Regression Trees*. Chapman & Hall, 1984.
- [66] L. Breiman, “Random forests,” *Machine Learning*, vol. 45, pp. 5–32, 2001.
- [67] E. Bauer and R. Kohavi, “An empirical comparison of voting classification algorithms: Bagging, boosting, and variants,” *Machine Learning*, vol. 36, pp. 105–139, 1999.
- [68] G. Tsoumakas, E. Spyromitros-Xioufīs, J. Vilcek, and I. Vlahavas, “Mulan: A java library for multi-label learning,” *J. Mach. Learn. Res.*, vol. 12, pp. 2411–2414, 2011.
- [69] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, “The weka data mining software: an update,” *SIGKDD Explorations*, vol. 11, pp. 10–18, 2009.
- [70] C.-C. Chang and C.-J. Lin, *LIBSVM: a library for support vector machines*, 2001. Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.
- [71] I. H. Witten and E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann Series in Data Management Systems, Morgan Kaufmann, second ed., 2005.
- [72] M. Friedman, “A comparison of alternative tests of significance for the problem of m rankings,” *Annals of Mathematical Statistics*, vol. 11, pp. 86–92, 1940.
- [73] P. B. Nemenyi, *Distribution-free multiple comparisons*. PhD thesis, Princeton University, Princeton, New Jersey, USA, 1963.
- [74] J. Demšar, “Statistical comparisons of classifiers over multiple data sets,” *Journal of Machine Learning Research*, vol. 7, pp. 1–30, 2006.
- [75] R. L. Iman and J. M. Davenport, “Approximations of the critical region of the friedman statistic,” *Communications in Statistics*, pp. 571–595, 1980.
- [76] E. S. Pearson and H. O. Hartley, *Biometrika Tables for Statisticians*, vol. I. Cambridge University Press, 1966.
- [77] F. Wilcoxon, “Individual Comparisons by Ranking Methods,” *Biometrics Bulletin*, vol. 1, no. 6, pp. 80–83, 1945.
- [78] A. M. Kumar and M. Gopal, “A hybrid svm based decision tree,” *Pattern Recognition*, vol. 43, pp. 3977–3987, 2010.
- [79] G. M. Dong and J. Chen, “Study on support vector machine based decision tree and application,” in *Proc. of the 5th International Conference on Fuzzy Systems and Knowledge Discovery*, pp. 318–322, 2008.
- [80] K. P. Bennett and J. A. Blue, “A Support Vector Machine Approach to Decision Trees,” in *Proc. of the International Joint Conference on Neural Networks*, (Anchorage, Alaska), pp. 2396–2401, 1997.
- [81] G. Madzarov, D. Gjorgjevikj, and I. Chorbev, “A multi-class svm classifier utilizing binary decision tree,” *Informatica*, vol. 33, no. 2, pp. 233–241, 2009.
- [82] K. Ting and L. Zhu, “Boosting support vector machines successfully,” in *Multiple Classifier Systems*, pp. 509–518, 2009.
- [83] J. Gama, “Functional trees,” *Machine Learning*, vol. 55, pp. 219–250, 2004.

- [84]R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification (2nd Edition)*. Wiley-Interscience, 2000.
- [85]C. J. C. Burges, “A tutorial on support vector machines for pattern recognition,” *Data Mining and Knowledge Discovery*, vol. 2, pp. 121–167, 1998.
- [86]T. Kanungo, D. M. Mount, N. S. Netanyahu, C. D. Piatko, R. Silverman, and A. Y. Wu, “An efficient k-means clustering algorithm: Analysis and implementation,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, pp. 881–892, 2002.

## Додаток А

**Табела А.1** Перформанси на методите за повеќезначна класификација и рангирање во однос на *Хаминг загуба* евалуациската мерка (DNF – *Did Not Finish* кај методите означува дека тие не конструирале предиктивен модел во временски период од една недела на расположливите ресурси)

	emotions	scene	yeast	medical	enron	corel5k	tmc2007	mediamill	bibtex	delicious	bookmarks
<b>BR</b>	0.257	0.079	<b>0.190</b>	0.077	<b>0.045</b>	0.017	0.013	0.032	0.012	0.018	DNF
<b>CC</b>	0.256	0.082	0.193	0.077	0.064	0.017	0.013	0.032	0.012	0.018	DNF
<b>CLR</b>	0.257	0.080	<b>0.190</b>	0.017	0.048	0.012	0.014	0.043	0.012	DNF	DNF
<b>TSVM</b>	0.259	0.080	<b>0.190</b>	<b>0.011</b>	0.046	0.013	0.013	0.032	0.012	DNF	DNF
<b>TSCCM</b>	0.259	0.080	0.192	<b>0.011</b>	0.046	0.011	0.013	0.032	0.012	DNF	DNF
<b>TSPCCM</b>	0.260	0.079	0.192	<b>0.011</b>	<b>0.045</b>	0.011	0.013	0.032	0.012	DNF	DNF
<b>QWML</b>	0.254	0.081	0.191	0.012	0.048	0.012	0.014	0.043	0.012	DNF	DNF
<b>HOMER</b>	0.361	0.082	0.207	0.012	0.051	0.012	0.015	0.038	0.014	0.022	DNF
<b>ML-C4.5</b>	0.247	0.141	0.234	0.013	0.053	0.010	0.093	0.044	0.016	0.019	0.009
<b>ML-SVMDT<sub>HL</sub></b>	0.247	<b>0.077</b>	<b>0.190</b>	<b>0.011</b>	0.046	<b>0.009</b>	0.013	0.030	<b>0.011</b>	<b>0.017</b>	<b>0.008</b>
<b>ML-SVMDT<sub>F1</sub></b>	0.258	0.099	0.212	0.012	0.049	<b>0.009</b>	<b>0.009</b>	0.031	0.012	0.018	0.009
<b>ML-SVMDT<sub>pre</sub></b>	0.254	0.097	0.205	<b>0.011</b>	0.049	<b>0.009</b>	0.011	0.032	0.012	0.018	0.009
<b>ML-SVMDT<sub>post</sub></b>	0.257	0.099	0.204	<b>0.011</b>	0.050	<b>0.009</b>	0.011	0.032	0.012	0.018	0.009
<b>PCT</b>	0.267	0.129	0.219	0.023	0.058	<b>0.009</b>	0.075	0.034	0.014	0.019	0.009
<b>ML-kNN</b>	0.294	0.099	0.198	0.017	0.051	<b>0.009</b>	0.058	0.031	0.014	0.018	0.009
<b>RAkEL</b>	0.282	<b>0.077</b>	0.192	0.012	<b>0.045</b>	<b>0.009</b>	0.021	0.035	DNF	DNF	DNF
<b>ECC</b>	0.281	0.085	0.207	0.014	0.049	<b>0.009</b>	0.026	0.035	0.013	DNF	DNF
<b>RFML-C4.5</b>	0.198	0.116	0.205	0.022	0.047	<b>0.009</b>	0.037	0.030	0.014	0.018	0.009
<b>RF-HHPCT</b>	<b>0.189</b>	0.080	0.195	0.012	0.046	<b>0.009</b>	0.015	<b>0.029</b>	0.012	0.018	<b>0.008</b>
<b>RF-PCT</b>	<b>0.189</b>	0.094	0.197	0.014	0.046	<b>0.009</b>	0.011	<b>0.029</b>	0.013	0.018	0.009

**Табела А.2** Перформанси на методите за повеќезначна класификација и рангирање во однос на *прецизност* евалуациската мерка (DNF – *Did Not Finish* кај методите означува дека тие не конструирале предиктивен модел во временски период од една недела на расположливите ресурси)

	emotions	scene	yeast	medical	enron	corel5k	tmc2007	mediamill	bibtex	delicious	bookmarks
<b>BR</b>	0.550	0.718	0.722	0.211	0.703	0.042	0.941	0.731	<b>0.515</b>	0.443	DNF
<b>CC</b>	0.551	0.758	0.727	0.217	0.464	0.042	0.944	0.741	0.508	0.399	DNF
<b>CLR</b>	0.538	0.714	0.719	0.695	0.650	<b>0.329</b>	0.937	0.201	0.488	DNF	DNF
<b>TSVM</b>	0.550	0.709	0.721	0.792	0.672	0.302	0.939	0.712	0.510	DNF	DNF
<b>TSCCM</b>	0.540	0.714	0.711	0.784	0.676	<b>0.329</b>	0.943	0.721	0.512	DNF	DNF
<b>TSPCCM</b>	0.533	0.710	0.711	0.780	0.675	0.289	0.944	0.722	0.512	DNF	DNF
<b>QWML</b>	0.548	0.711	0.718	0.697	0.624	0.326	0.937	0.203	0.496	DNF	DNF
<b>HOMER</b>	0.509	0.746	0.663	0.762	0.616	0.317	0.926	0.597	0.472	0.369	DNF
<b>ML-C4.5</b>	0.606	0.592	0.620	0.797	0.623	0.005	0.146	0.056	0.123	0.001	0.271
<b>ML-SVMDT<sub>HL</sub></b>	0.548	0.718	0.692	<b>0.806</b>	0.670	0.142	0.937	0.731	0.495	0.506	0.277
<b>ML-SVMDT<sub>F1</sub></b>	0.549	0.628	0.705	0.777	0.662	0.139	0.962	0.731	0.482	0.497	0.278
<b>ML-SVMDT<sub>pre</sub></b>	0.583	0.668	0.698	0.790	0.669	0.127	0.946	0.727	0.484	0.486	<b>0.281</b>
<b>ML-SVMDT<sub>post</sub></b>	0.556	0.662	0.700	0.798	0.660	0.135	0.949	0.729	0.490	0.477	<b>0.281</b>
<b>PCT</b>	0.577	0.565	0.705	0.285	0.415	0.000	0.659	0.694	0.140	0.001	0.133
<b>ML-kNN</b>	0.502	0.661	0.732	0.575	0.587	0.035	0.738	0.724	0.254	0.424	0.218
<b>RAkEL</b>	0.564	0.768	0.715	0.730	0.708	0.000	0.928	0.705	DNF	DNF	DNF
<b>ECC</b>	0.580	<b>0.770</b>	0.667	0.662	0.652	0.002	0.872	0.690	0.324	DNF	DNF
<b>RFML-C4.5</b>	0.625	0.403	0.738	0.284	0.690	0.018	0.874	0.765	0.159	0.472	0.182
<b>RF-HHPCT</b>	0.637	0.555	<b>0.747</b>	0.731	<b>0.713</b>	0.071	0.966	<b>0.774</b>	0.423	<b>0.512</b>	0.225
<b>RF-PCT</b>	<b>0.644</b>	0.565	0.744	0.635	0.709	0.030	<b>0.977</b>	0.772	0.292	<b>0.512</b>	0.218

**Табела А.3** Перформанси на методите за повеќезначна класификација и рангирање во однос на *odsv* евалуациската мерка (DNF – *Did Not Finish* кај методите означува дека тие не конструирале предиктивен модел во временски период од една недела на расположливите ресурси)

	emotions	scene	yeast	medical	enron	corel5k	tmc2007	mediamill	bibtex	delicious	bookmarks
<b>BR</b>	0.409	0.711	0.591	0.735	0.497	0.055	0.928	0.450	0.373	0.155	DNF
<b>CC</b>	0.397	0.726	0.600	0.754	0.507	0.056	0.934	0.424	0.378	0.157	DNF
<b>CLR</b>	0.410	0.712	0.601	0.795	0.557	0.264	0.929	0.101	0.364	DNF	DNF
<b>TSVM</b>	0.412	0.704	0.602	<b>0.842</b>	0.568	<b>0.296</b>	0.930	0.464	0.383	DNF	DNF
<b>TSCCM</b>	0.415	0.712	0.616	0.801	0.565	0.272	0.934	0.468	0.382	DNF	DNF
<b>TSPCCM</b>	0.400	0.705	0.616	0.798	0.572	0.197	0.936	0.464	0.382	DNF	DNF
<b>QWML</b>	0.429	0.709	0.600	0.801	0.453	0.264	0.929	0.101	0.366	DNF	DNF
<b>HOMER</b>	<b>0.775</b>	0.744	<b>0.714</b>	0.760	<b>0.610</b>	0.250	0.943	<b>0.563</b>	0.389	<b>0.303</b>	DNF
<b>ML-C4.5</b>	0.703	0.582	0.608	0.740	0.487	0.002	0.111	0.052	0.111	0.001	0.244
<b>ML-SVMDT<sub>HL</sub></b>	0.697	0.711	0.566	0.754	0.547	0.149	0.936	0.457	<b>0.392</b>	0.236	0.248
<b>ML-SVMDT<sub>F1</sub></b>	0.706	0.641	0.574	0.718	0.521	0.147	0.948	0.481	0.388	0.235	<b>0.256</b>
<b>ML-SVMDT<sub>pre</sub></b>	0.545	0.655	0.581	0.760	0.486	0.054	<b>0.952</b>	0.452	0.327	0.159	0.246
<b>ML-SVMDT<sub>post</sub></b>	0.535	0.648	0.586	0.765	0.480	0.056	0.948	0.449	0.338	0.161	0.255
<b>PCT</b>	0.534	0.539	0.490	0.227	0.229	0.000	0.478	0.379	0.046	0.001	0.137
<b>ML-kNN</b>	0.377	0.655	0.549	0.547	0.358	0.014	0.664	0.470	0.132	0.112	0.207
<b>RAkEL</b>	0.491	0.740	0.615	0.679	0.469	0.000	0.880	0.353	DNF	DNF	DNF
<b>ECC</b>	0.533	<b>0.771</b>	0.673	0.642	0.560	0.001	0.903	0.372	0.187	DNF	DNF
<b>RFML-C4.5</b>	0.545	0.388	0.491	0.251	0.398	0.005	0.677	0.456	0.060	0.176	0.181
<b>RF-HHPCT</b>	0.588	0.623	0.645	0.690	0.540	0.024	0.885	0.488	0.260	0.171	0.214
<b>RF-PCT</b>	0.582	0.541	0.523	0.599	0.452	0.009	0.920	0.476	0.167	0.160	0.208

**Табела А.4** Перформанси на методите за повеќезначна класификација и рангирање во однос на *точност* евалуациската мерка (DNF – *Did Not Finish* кај методите означува дека тие не конструирале предиктивен модел во временски период од една недела на расположливите ресурси)

	emotions	scene	yeast	medical	enron	corel5k	tmc2007	mediamill	bibtex	delicious	bookmarks
<b>BR</b>	0.361	0.689	0.520	0.206	0.446	0.030	0.891	0.403	0.348	0.136	DNF
<b>CC</b>	0.356	0.723	0.527	0.211	0.334	0.030	0.899	0.390	<b>0.352</b>	0.137	DNF
<b>CLR</b>	0.361	0.686	0.524	0.656	0.459	0.195	0.889	0.095	0.334	DNF	DNF
<b>TSVM</b>	0.369	0.680	0.526	<b>0.758</b>	0.478	0.200	0.892	0.407	<b>0.352</b>	DNF	DNF
<b>TSCCM</b>	0.366	0.686	0.530	0.744	0.478	<b>0.202</b>	0.895	0.411	<b>0.352</b>	DNF	DNF
<b>TSPCCM</b>	0.357	0.681	0.529	0.737	<b>0.481</b>	0.150	0.897	0.410	<b>0.352</b>	DNF	DNF
<b>QWML</b>	0.373	0.683	0.523	0.658	0.388	0.195	0.889	0.095	0.338	DNF	DNF
<b>HOMER</b>	0.471	0.717	<b>0.559</b>	0.713	0.478	0.179	0.888	0.413	0.330	<b>0.207</b>	DNF
<b>ML-C4.5</b>	<b>0.536</b>	0.569	0.480	0.730	0.418	0.002	0.110	0.052	0.108	0.001	0.237
<b>ML-SVMDT<sub>HL</sub></b>	0.377	0.689	0.493	0.739	0.429	0.058	0.893	0.406	0.320	0.148	0.228
<b>ML-SVMDT<sub>F1</sub></b>	0.386	0.656	0.494	0.737	0.423	0.052	0.901	0.407	0.321	0.147	0.226
<b>ML-SVMDT<sub>pre</sub></b>	0.443	0.638	0.498	0.732	0.428	0.052	0.910	0.404	0.311	0.143	0.238
<b>ML-SVMDT<sub>post</sub></b>	0.435	0.632	0.501	0.740	0.421	0.055	0.912	0.404	0.319	0.144	<b>0.246</b>
<b>PCT</b>	0.448	0.538	0.440	0.228	0.196	0.000	0.436	0.354	0.046	0.001	0.133
<b>ML-kNN</b>	0.319	0.629	0.492	0.528	0.319	0.014	0.574	0.421	0.129	0.102	0.202
<b>RAkEL</b>	0.419	0.734	0.531	0.673	0.428	0.000	0.852	0.337	DNF	DNF	DNF
<b>ECC</b>	0.432	<b>0.735</b>	0.546	0.611	0.462	0.001	0.808	0.349	0.186	DNF	DNF
<b>RFML-C4.5</b>	0.488	0.388	0.453	0.250	0.374	0.005	0.663	0.423	0.060	0.151	0.176
<b>RF-HHPCT</b>	0.520	0.555	0.475	0.715	0.426	0.023	<b>0.916</b>	0.437	0.256	0.138	0.209
<b>RF-PCT</b>	0.519	0.541	0.478	0.591	0.416	0.009	0.914	<b>0.441</b>	0.166	0.146	0.204

**Табела А.5** Перформанси на методите за повеќезначна класификација и рангирање во однос на  $F_1$  евалуациската мерка (DNF – *Did Not Finish* кај методите означува дека тие не конструирале предиктивен модел во временски период од една недела на расположливите ресурси)

	emotions	scene	yeast	medical	enron	core5k	tmc2007	mediamill	bibtex	delicious	bookmarks
<b>BR</b>	0.469	0.714	0.650	0.328	0.582	0.047	0.934	0.557	0.433	0.230	DNF
<b>CC</b>	0.461	0.742	0.657	0.337	0.484	0.048	0.939	0.539	0.434	0.225	DNF
<b>CLR</b>	0.465	0.713	0.655	0.742	0.600	0.293	0.933	0.134	0.417	DNF	DNF
<b>TSVM</b>	0.471	0.706	0.656	<b>0.816</b>	0.616	<b>0.299</b>	0.935	0.562	0.437	DNF	DNF
<b>TSCCM</b>	0.469	0.713	0.660	0.792	0.616	0.298	0.938	0.567	0.437	DNF	DNF
<b>TSPCCM</b>	0.457	0.708	0.660	0.789	<b>0.620</b>	0.234	0.940	0.565	0.437	DNF	DNF
<b>QWML</b>	0.481	0.710	0.654	0.745	0.525	0.292	0.933	0.135	0.421	DNF	DNF
<b>HOMER</b>	0.614	0.745	0.687	0.761	0.613	0.280	0.934	0.579	0.426	<b>0.343</b>	DNF
<b>ML-C4.5</b>	<b>0.651</b>	0.587	0.614	0.768	0.546	0.003	0.126	0.054	0.117	0.001	0.257
<b>ML-SVMDT<sub>HL</sub></b>	0.614	0.714	0.623	0.779	0.602	0.145	0.936	0.562	<b>0.438</b>	0.322	0.262
<b>ML-SVMDT<sub>F1</sub></b>	0.618	0.634	0.633	0.746	0.583	0.143	<b>0.955</b>	0.580	0.430	0.319	<b>0.267</b>
<b>ML-SVMDT<sub>pre</sub></b>	0.563	0.661	0.634	0.775	0.563	0.076	0.949	0.557	0.390	0.240	0.262
<b>ML-SVMDT<sub>post</sub></b>	0.545	0.655	0.638	0.781	0.556	0.079	0.949	0.556	0.400	0.240	<b>0.267</b>
<b>PCT</b>	0.554	0.551	0.578	0.253	0.295	0.000	0.554	0.490	0.069	0.001	0.135
<b>ML-kNN</b>	0.431	0.658	0.628	0.560	0.445	0.021	0.699	0.570	0.174	0.017	0.213
<b>RAkEL</b>	0.525	0.754	0.661	0.704	0.564	0.000	0.904	0.471	DNF	DNF	DNF
<b>ECC</b>	0.556	<b>0.771</b>	0.670	0.652	0.602	0.001	0.887	0.483	0.237	DNF	DNF
<b>RFML-C4.5</b>	0.583	0.395	0.589	0.267	0.505	0.008	0.763	0.572	0.087	0.256	0.181
<b>RF-HHPCT</b>	0.612	0.587	<b>0.692</b>	0.710	0.615	0.036	0.924	<b>0.599</b>	0.322	0.256	0.219
<b>RF-PCT</b>	0.611	0.553	0.614	0.616	0.552	0.014	0.948	0.589	0.212	0.244	0.213

**Табела А.6** Перформанси на методите за повеќезначна класификација и рангирање во однос на *класификациска точност* евалуациската мерка (DNF – *Did Not Finish* кај методите означува дека тие не конструирале предиктивен модел во временски период од една недела на расположливите ресурси)

	emotions	scene	yeast	medical	enron	core5k	tmc2007	mediamill	bibtex	delicious	bookmarks
<b>BR</b>	0.129	0.639	0.190	0.000	<b>0.149</b>	0.000	0.772	0.080	0.194	0.004	DNF
<b>CC</b>	0.124	0.685	<b>0.239</b>	0.000	0.000	0.000	0.787	0.080	<b>0.202</b>	0.006	DNF
<b>CLR</b>	0.144	0.633	0.195	0.486	0.117	0.010	0.767	0.044	0.183	DNF	DNF
<b>TSVM</b>	0.149	0.629	0.195	0.640	0.140	0.008	0.773	0.081	0.198	DNF	DNF
<b>TSCCM</b>	0.149	0.632	0.193	0.649	0.143	0.011	0.775	0.084	0.197	DNF	DNF
<b>TSPCCM</b>	0.144	0.629	0.192	0.634	0.138	0.006	0.775	0.083	0.197	DNF	DNF
<b>QWML</b>	0.149	0.630	0.192	0.480	0.097	<b>0.012</b>	0.768	0.044	0.186	DNF	DNF
<b>HOMER</b>	0.163	0.661	0.213	0.610	0.145	0.002	0.765	0.053	0.165	0.001	DNF
<b>ML-C4.5</b>	0.277	0.533	0.158	0.646	0.140	0.000	0.078	0.049	0.095	0.001	0.209
<b>ML-SVMDT<sub>HL</sub></b>	0.183	0.639	0.168	<b>0.655</b>	0.145	0.008	0.776	0.092	0.183	0.002	0.205
<b>ML-SVMDT<sub>F1</sub></b>	0.180	0.612	0.166	0.649	0.142	0.008	0.789	0.092	0.181	0.002	0.204
<b>ML-SVMDT<sub>pre</sub></b>	0.178	0.590	0.165	0.643	0.140	0.008	0.796	0.091	0.179	0.004	0.210
<b>ML-SVMDT<sub>post</sub></b>	0.198	0.585	0.166	<b>0.655</b>	0.138	0.008	0.802	0.091	0.179	0.005	<b>0.215</b>
<b>PCT</b>	0.223	0.509	0.152	0.177	0.002	0.000	0.215	0.065	0.004	0.001	0.129
<b>ML-kNN</b>	0.084	0.573	0.159	0.462	0.062	0.000	0.305	0.110	0.056	0.003	0.187
<b>RAkEL</b>	0.208	<b>0.694</b>	0.201	0.607	0.136	0.000	0.734	0.060	DNF	DNF	DNF
<b>ECC</b>	0.168	0.665	0.215	0.526	0.131	0.001	0.608	0.065	0.109	DNF	DNF
<b>RFML-C4.5</b>	0.272	0.372	0.129	0.216	0.124	0.008	0.421	0.104	0.011	<b>0.018</b>	0.167
<b>RF-HHPCT</b>	<b>0.308</b>	0.508	0.182	0.622	0.136	0.000	0.748	<b>0.128</b>	0.165	0.007	0.193
<b>RF-PCT</b>	0.307	0.518	0.152	0.538	0.131	0.000	<b>0.816</b>	0.122	0.098	0.007	0.189

**Табела А.7** Перформанси на методите за повеќезначна класификација и рангирање во однос на *микро прецизност* евалуациската мерка (DNF – *Did Not Finish* кај методите означува дека тие не конструирале предиктивен модел во временски период од една недела на расположливите ресурси)

	emotions	scene	yeast	medical	enron	core5k	tmc2007	mediamill	bibtex	delicious	bookmarks
<b>BR</b>	0.684	0.843	0.733	0.225	0.721	0.061	0.947	0.742	0.753	0.658	DNF
<b>CC</b>	0.698	0.814	0.726	0.229	0.492	0.061	0.948	0.753	0.744	0.660	DNF
<b>CLR</b>	0.685	0.835	0.729	0.669	0.652	0.338	0.940	0.582	0.734	DNF	DNF
<b>TSVM</b>	0.673	0.839	0.729	0.778	0.672	0.312	0.944	0.717	0.736	DNF	DNF
<b>TSCCM</b>	0.673	0.838	0.717	0.807	0.677	0.320	0.946	0.725	0.733	DNF	DNF
<b>TSPCCM</b>	0.681	0.844	0.717	0.798	0.677	0.323	0.946	0.727	0.733	DNF	DNF
<b>QWML</b>	0.680	0.832	0.727	0.667	0.687	0.339	0.941	0.580	0.736	DNF	DNF
<b>HOMER</b>	0.471	0.804	0.647	0.807	0.597	0.308	0.922	0.569	0.547	0.396	DNF
<b>ML-C4.5</b>	0.607	0.619	0.618	0.796	0.613	0.160	0.940	0.597	0.359	0.000	0.632
<b>ML-SVMDT<sub>HL</sub></b>	0.683	0.843	0.738	0.843	0.669	0.698	0.942	0.755	0.787	0.692	0.798
<b>ML-SVMDT<sub>F1</sub></b>	0.643	0.798	0.730	0.839	0.648	0.657	0.977	0.760	0.788	0.679	0.793
<b>ML-SVMDT<sub>pre</sub></b>	0.634	0.787	0.695	0.834	0.668	0.664	0.947	0.749	0.789	0.662	0.855
<b>ML-SVMDT<sub>post</sub></b>	0.628	0.782	0.698	0.830	0.657	0.662	0.951	0.745	0.764	0.641	0.755
<b>PCT</b>	0.607	0.512	0.698	0.826	0.601	0.000	0.689	0.743	<b>1.000</b>	0.000	<b>0.947</b>
<b>ML-kNN</b>	0.584	0.691	0.736	0.807	0.684	0.730	0.757	0.739	0.819	0.651	0.850
<b>RAkEL</b>	0.586	0.831	0.720	0.881	0.743	0.000	0.938	0.725	DNF	DNF	DNF
<b>ECC</b>	0.579	0.773	0.662	0.834	0.642	0.333	0.869	0.708	0.948	DNF	DNF
<b>RFML-C4.5</b>	0.783	<b>0.960</b>	0.747	0.884	<b>0.768</b>	<b>0.750</b>	0.963	0.788	0.940	0.589	0.878
<b>RF-HHPCT</b>	<b>0.785</b>	0.932	0.749	<b>0.895</b>	0.740	0.699	0.987	<b>0.800</b>	0.912	<b>0.709</b>	0.885
<b>RF-PCT</b>	0.783	0.930	<b>0.755</b>	0.885	0.738	0.696	<b>0.992</b>	0.798	0.957	0.695	0.895

**Табела А.8** Перформанси на методите за повеќезначна класификација и рангирање во однос на *микро одзив* евалуациската мерка (DNF – *Did Not Finish* кај методите означува дека тие не конструирале предиктивен модел во временски период од една недела на расположливите ресурси)

	emotions	scene	yeast	medical	enron	corel5k	tmc2007	mediamill	bibtex	delicious	bookmarks
<b>BR</b>	0.406	0.694	0.587	0.725	0.464	0.057	0.917	0.415	0.328	0.143	DNF
<b>CC</b>	0.393	0.708	0.588	0.739	0.472	0.057	0.924	0.385	0.335	0.144	DNF
<b>CLR</b>	0.409	0.695	0.595	0.782	0.532	0.258	0.920	0.066	0.322	DNF	DNF
<b>TSVM</b>	0.414	0.687	0.595	<b>0.830</b>	0.541	<b>0.290</b>	0.921	0.429	0.341	DNF	DNF
<b>TSCCM</b>	0.414	0.695	0.610	0.789	0.537	0.282	0.923	0.433	0.338	DNF	DNF
<b>TSPCCM</b>	0.396	0.689	0.609	0.785	0.544	0.197	0.925	0.429	0.338	DNF	DNF
<b>QWML</b>	0.431	0.692	0.595	0.787	0.438	0.258	0.920	0.066	0.328	DNF	DNF
<b>HOMER</b>	<b>0.782</b>	0.727	<b>0.702</b>	0.742	<b>0.585</b>	0.248	0.932	<b>0.537</b>	<b>0.353</b>	<b>0.297</b>	DNF
<b>ML-C4.5</b>	0.712	0.570	0.603	0.720	0.440	0.002	0.073	0.004	0.053	0.000	0.170
<b>ML-SVMDT<sub>HL</sub></b>	0.562	0.698	0.587	0.730	0.529	0.109	0.927	0.428	0.296	0.182	0.176
<b>ML-SVMDT<sub>F1</sub></b>	0.549	0.683	0.582	0.719	0.474	0.067	0.932	0.437	0.302	0.173	0.174
<b>ML-SVMDT<sub>pre</sub></b>	0.539	0.638	0.577	0.735	0.457	0.055	<b>0.943</b>	0.416	0.281	0.148	0.170
<b>ML-SVMDT<sub>post</sub></b>	0.534	0.631	0.579	0.738	0.457	0.058	0.940	0.411	0.289	0.149	<b>0.178</b>
<b>PCT</b>	0.539	0.521	0.492	0.227	0.246	0.000	0.454	0.351	0.057	0.000	0.135
<b>ML-kNN</b>	0.376	0.634	0.543	0.522	0.353	0.015	0.621	0.432	0.118	0.101	0.135
<b>RAkEL</b>	0.489	0.721	0.602	0.600	0.435	0.000	0.847	0.315	DNF	DNF	DNF
<b>ECC</b>	0.531	<b>0.751</b>	0.655	0.624	0.532	0.001	0.869	0.333	0.142	DNF	DNF
<b>RFML-C4.5</b>	0.551	0.572	0.491	0.237	0.366	0.005	0.651	0.418	0.066	0.174	0.112
<b>RF-HHPCT</b>	0.569	0.514	0.525	0.651	0.431	0.024	0.896	0.427	0.203	0.140	0.139
<b>RF-PCT</b>	0.589	0.523	0.521	0.569	0.422	0.009	0.902	0.435	0.131	0.151	0.136

**Табела А.9** Перформанси на методите за повеќезначна класификација и рангирање во однос на *микро*  $F_1$  евалуациската мерка (DNF – *Did Not Finish* кај методите означува дека тие не конструирале предиктивен модел во временски период од една недела на расположливите ресурси)

	emotions	scene	yeast	medical	enron	corel5k	tmc2007	mediamill	bibtex	delicious	bookmarks
<b>BR</b>	0.509	0.761	0.652	0.343	0.564	0.059	0.932	0.533	0.457	0.234	DNF
<b>CC</b>	0.503	0.757	0.650	0.350	0.482	0.059	0.936	0.509	0.462	0.236	DNF
<b>CLR</b>	0.512	0.758	0.655	0.721	0.585	0.293	0.930	0.118	0.448	DNF	DNF
<b>TSVM</b>	0.512	0.756	0.655	<b>0.803</b>	0.599	<b>0.301</b>	0.933	0.537	<b>0.466</b>	DNF	DNF
<b>TSCCM</b>	0.512	0.760	0.659	0.798	0.599	0.299	0.935	0.542	0.463	DNF	DNF
<b>TSPCCM</b>	0.501	0.758	0.659	0.791	<b>0.603</b>	0.245	0.936	0.540	0.463	DNF	DNF
<b>QWML</b>	0.528	0.756	0.654	0.722	0.535	0.293	0.930	0.119	0.454	DNF	DNF
<b>HOMER</b>	0.588	0.764	<b>0.673</b>	0.773	0.591	0.275	0.927	0.553	0.429	<b>0.339</b>	DNF
<b>ML-C4.5</b>	0.655	0.593	0.610	0.756	0.512	0.004	0.135	0.007	0.093	0.000	0.268
<b>ML-SVMDT<sub>HL</sub></b>	0.617	0.764	0.654	0.782	0.591	0.188	0.934	0.546	0.430	0.288	<b>0.288</b>
<b>ML-SVMDT<sub>F1</sub></b>	0.592	0.736	0.648	0.774	0.547	0.121	<b>0.954</b>	0.555	0.437	0.276	0.285
<b>ML-SVMDT<sub>pre</sub></b>	0.583	0.705	0.630	0.781	0.543	0.102	0.945	0.535	0.414	0.242	0.283
<b>ML-SVMDT<sub>post</sub></b>	0.577	0.698	0.633	0.781	0.539	0.106	0.946	0.530	0.419	0.242	<b>0.288</b>
<b>PCT</b>	0.571	0.516	0.577	0.356	0.349	0.000	0.547	0.477	0.108	0.000	0.236
<b>ML-kNN</b>	0.457	0.661	0.625	0.634	0.466	0.030	0.682	0.545	0.206	0.175	0.232
<b>RAkEL</b>	0.533	<b>0.772</b>	0.656	0.714	0.548	0.000	0.890	0.440	DNF	DNF	DNF
<b>ECC</b>	0.554	0.762	0.658	0.714	0.582	0.002	0.869	0.453	0.247	DNF	DNF
<b>RFML-C4.5</b>	0.647	0.717	0.593	0.374	0.496	0.010	0.777	0.546	0.123	0.269	0.199
<b>RF-HHPCT</b>	0.660	0.663	0.617	0.754	0.545	0.046	0.939	0.557	0.332	0.234	0.240
<b>RF-PCT</b>	<b>0.672</b>	0.669	0.617	0.693	0.537	0.018	0.945	<b>0.563</b>	0.230	0.248	0.236

**Табела А.10** Перформанси на методите за повеќезначна класификација и рангирање во однос на *макро прецизност* евалуациската мерка (DNF – *Did Not Finish* кај методите означува дека тие не конструирале предиктивен модел во временски период од една недела на расположливите ресурси)

	emotions	scene	yeast	medical	enron	core5k	tmc2007	mediamill	bibtex	delicious	bookmarks
<b>BR</b>	0.721	0.844	0.628	<b>0.399</b>	0.258	0.052	0.972	0.112	0.528	0.299	DNF
<b>CC</b>	0.581	0.817	0.602	0.391	0.260	0.053	0.972	0.144	0.539	0.303	DNF
<b>CLR</b>	0.677	0.835	0.614	0.288	0.205	0.059	0.964	0.140	0.503	DNF	DNF
<b>TSVM</b>	0.602	0.839	0.630	0.387	0.282	<b>0.063</b>	0.970	0.115	0.530	DNF	DNF
<b>TSCCM</b>	0.614	0.838	0.582	0.379	0.267	<b>0.063</b>	0.971	0.131	<b>0.542</b>	DNF	DNF
<b>TSPCCM</b>	0.634	0.843	0.593	0.377	0.278	0.056	0.971	0.123	<b>0.542</b>	DNF	DNF
<b>QWML</b>	0.660	0.832	0.614	0.285	0.242	0.059	0.965	0.133	0.490	DNF	DNF
<b>HOMER</b>	0.464	0.807	0.471	0.287	0.241	0.044	0.954	0.107	0.391	0.154	DNF
<b>ML-C4.5</b>	0.602	0.635	0.377	0.263	0.142	0.004	0.925	0.046	0.128	0.000	0.292
<b>ML-SVMDT<sub>HL</sub></b>	0.654	0.844	0.559	0.364	0.246	0.055	0.980	0.261	0.513	0.319	<b>0.526</b>
<b>ML-SVMDT<sub>F1</sub></b>	0.632	0.805	0.551	0.377	0.250	0.055	0.986	0.270	0.480	0.318	0.495
<b>ML-SVMDT<sub>pre</sub></b>	0.625	0.785	0.515	0.379	0.266	0.055	0.984	0.258	0.495	0.312	0.485
<b>ML-SVMDT<sub>post</sub></b>	0.596	0.777	0.545	0.320	<b>0.307</b>	0.055	0.983	0.201	0.520	0.290	0.485
<b>PCT</b>	0.628	0.682	0.479	0.018	0.023	0.000	0.386	0.401	0.006	0.000	0.018
<b>ML-kNN</b>	0.518	0.784	0.600	0.267	0.170	0.031	0.780	0.308	0.192	0.134	0.414
<b>RAkEL</b>	0.547	0.835	0.480	0.269	0.222	0.000	0.973	0.025	DNF	DNF	DNF
<b>ECC</b>	0.531	0.785	0.391	0.266	0.249	0.001	0.938	0.037	0.121	DNF	DNF
<b>RFML-C4.5</b>	<b>0.828</b>	<b>0.963</b>	0.533	0.190	0.245	0.007	0.994	0.397	0.080	<b>0.422</b>	0.388
<b>RF-HHPCT</b>	0.801	0.911	0.580	0.272	0.246	0.028	0.995	<b>0.455</b>	0.324	0.252	0.496
<b>RF-PCT</b>	0.802	0.919	<b>0.674</b>	0.269	0.233	0.015	<b>0.997</b>	0.441	0.127	0.293	0.522

**Табела А.11** Перформанси на методите за повеќезначна класификација и рангирање во однос на *Macro Recall* евалуациската мерка (DNF – *Did Not Finish* кај методите означува дека тие не конструирале предиктивен модел во временски период од една недела на расположливите ресурси)

	emotions	scene	yeast	medical	enron	core5k	tmc2007	mediamill	bibtex	delicious	bookmarks
<b>BR</b>	0.378	0.703	0.355	0.423	0.120	0.023	0.915	0.049	0.250	0.072	DNF
<b>CC</b>	0.364	0.716	0.357	<b>0.428</b>	0.146	0.023	<b>0.924</b>	0.044	0.257	0.075	DNF
<b>CLR</b>	0.381	0.704	0.361	0.307	0.139	0.039	0.914	0.028	0.236	DNF	DNF
<b>TSVM</b>	0.383	0.697	0.359	0.399	0.148	<b>0.048</b>	0.917	0.058	<b>0.262</b>	DNF	DNF
<b>TSCCM</b>	0.380	0.704	0.379	0.387	0.154	<b>0.048</b>	0.922	0.061	0.261	DNF	DNF
<b>TSPCCM</b>	0.363	0.698	0.378	0.388	0.146	0.033	<b>0.924</b>	0.056	0.261	DNF	DNF
<b>QWML</b>	0.398	0.701	0.361	0.324	0.120	0.039	0.914	0.028	0.238	DNF	DNF
<b>HOMER</b>	<b>0.775</b>	0.734	<b>0.466</b>	0.282	<b>0.163</b>	0.041	0.897	0.074	0.247	<b>0.103</b>	DNF
<b>ML-C4.5</b>	0.702	0.573	0.375	0.249	0.107	0.005	0.085	0.002	0.034	0.000	0.098
<b>ML-SVMDT<sub>HL</sub></b>	0.556	0.703	0.343	0.323	0.128	0.020	0.906	0.062	0.223	0.072	0.095
<b>ML-SVMDT<sub>F1</sub></b>	0.538	0.635	0.331	0.328	0.114	0.019	0.912	0.071	0.217	0.069	0.094
<b>ML-SVMDT<sub>pre</sub></b>	0.511	0.646	0.344	0.313	0.119	0.014	0.903	0.060	0.198	0.066	0.103
<b>ML-SVMDT<sub>post</sub></b>	0.503	0.639	0.348	0.339	0.130	0.014	0.902	0.061	0.203	0.064	<b>0.111</b>
<b>PCT</b>	0.533	0.529	0.269	0.022	0.030	0.000	0.235	0.029	0.006	0.000	0.016
<b>ML-kNN</b>	0.334	0.647	0.308	0.163	0.075	0.006	0.418	<b>0.088</b>	0.049	0.039	0.070
<b>RAkEL</b>	0.462	0.727	0.352	0.183	0.097	0.000	0.739	0.020	DNF	DNF	DNF
<b>ECC</b>	0.508	<b>0.757</b>	0.388	0.179	0.129	0.001	0.772	0.023	0.044	DNF	DNF
<b>RFML-C4.5</b>	0.532	0.381	0.257	0.040	0.082	0.001	0.297	0.065	0.013	0.092	0.048
<b>RF-HHPCT</b>	0.550	0.525	0.285	0.218	0.104	0.004	0.678	0.081	0.104	0.049	0.071
<b>RF-PCT</b>	0.569	0.533	0.286	0.176	0.100	0.002	0.769	0.080	0.043	0.060	0.072

**Табела А.12** Перформанси на методите за повеќезначна класификација и рангирање во однос на *Macro F<sub>1</sub>* евалуациската мерка (DNF – *Did Not Finish* кај методите означува дека тие не конструирале предиктивен модел во временски период од една недела на расположливите ресурси)

	emotions	scene	yeast	medical	enron	core5k	tmc2007	mediamill	bibtex	delicious	bookmarks
<b>BR</b>	0.440	0.765	0.392	0.361	0.143	0.021	0.942	0.056	0.307	0.096	DNF
<b>CC</b>	0.420	0.762	0.390	0.371	0.153	0.021	<b>0.947</b>	0.052	0.316	0.100	DNF
<b>CLR</b>	0.443	0.762	0.392	0.281	0.149	0.042	0.938	0.037	0.291	DNF	DNF
<b>TSVM</b>	0.443	0.759	0.388	<b>0.383</b>	0.169	<b>0.048</b>	0.943	0.064	<b>0.318</b>	DNF	DNF
<b>TSCCM</b>	0.442	0.763	0.411	0.370	<b>0.176</b>	<b>0.048</b>	0.945	0.067	<b>0.318</b>	DNF	DNF
<b>TSPCCM</b>	0.426	0.761	0.409	0.369	0.166	0.037	<b>0.947</b>	0.063	<b>0.318</b>	DNF	DNF
<b>QWML</b>	0.458	0.759	0.394	0.286	0.143	0.042	0.938	0.037	0.292	DNF	DNF
<b>HOMER</b>	0.570	0.768	<b>0.447</b>	0.282	0.167	0.036	0.924	0.073	0.266	0.103	DNF
<b>ML-C4.5</b>	0.630	0.596	0.370	0.250	0.115	0.008	0.124	0.003	0.045	0.000	0.119
<b>ML-SVMDT<sub>HL</sub></b>	0.601	0.765	0.379	0.337	0.155	0.036	0.940	0.081	0.311	0.105	<b>0.161</b>
<b>ML-SVMDT<sub>F1</sub></b>	0.535	0.705	0.373	0.343	0.148	0.032	0.935	<b>0.132</b>	0.258	0.102	0.158
<b>ML-SVMDT<sub>pre</sub></b>	0.525	0.704	0.372	0.331	0.141	0.020	0.939	0.075	0.249	0.089	0.144
<b>ML-SVMDT<sub>post</sub></b>	0.510	0.698	0.377	0.339	0.159	0.020	0.939	0.076	0.255	0.087	0.156
<b>PCT</b>	0.568	0.593	0.293	0.020	0.026	0.000	0.263	0.031	0.006	0.000	0.017
<b>ML-kNN</b>	0.385	0.692	0.336	0.192	0.087	0.010	0.493	0.113	0.065	0.051	0.096
<b>RAkEL</b>	0.488	<b>0.777</b>	0.359	0.210	0.115	0.000	0.826	0.019	DNF	DNF	DNF
<b>ECC</b>	0.500	0.770	0.350	0.203	0.140	0.001	0.834	0.022	0.052	DNF	DNF
<b>RFML-C4.5</b>	0.620	0.514	0.283	0.058	0.102	0.001	0.371	0.088	0.016	<b>0.142</b>	0.065
<b>RF-HHPCT</b>	0.628	0.648	0.321	0.237	0.127	0.007	0.780	0.115	0.134	0.068	0.099
<b>RF-PCT</b>	<b>0.650</b>	0.658	0.322	0.207	0.122	0.004	0.857	0.112	0.055	0.083	0.101

**Табела А.13** Перформанси на методите за повеќезначна класификација и рангирање во однос на една грешка евалуациската мерка (DNF – *Did Not Finish* кај методите означува дека тие не конструирале предиктивен модел во временски период од една недела на расположливите ресурси)

	emotions	scene	yeast	medical	enron	corel5k	tmc2007	mediamill	bibtex	delicious	bookmarks
<b>BR</b>	0.386	<b>0.180</b>	0.236	0.135	0.237	0.660	0.029	0.188	0.346	0.354	DNF
<b>CC</b>	0.376	0.204	0.268	0.123	0.238	0.674	0.026	0.193	0.342	0.367	DNF
<b>CLR</b>	0.391	0.190	<b>0.229</b>	0.168	0.231	<b>0.588</b>	0.033	0.586	0.388	DNF	DNF
<b>TSVM</b>	0.342	0.181	0.231	0.129	<b>0.212</b>	0.592	0.032	0.262	0.345	DNF	DNF
<b>TSCCM</b>	0.337	0.183	0.240	0.120	0.216	0.590	0.019	0.240	<b>0.341</b>	DNF	DNF
<b>TSPCCM</b>	0.337	0.181	0.244	<b>0.117</b>	0.218	0.656	0.018	0.251	<b>0.341</b>	DNF	DNF
<b>QWML</b>	0.391	0.193	0.233	0.165	0.269	0.592	0.033	0.560	0.380	DNF	DNF
<b>HOMER</b>	0.411	0.216	0.248	0.216	0.314	0.652	0.050	0.219	0.466	0.509	DNF
<b>ML-C4.5</b>	0.347	0.394	0.312	0.198	0.309	0.762	0.145	0.194	0.529	0.411	0.643
<b>ML-SVMDT<sub>HL</sub></b>	0.376	<b>0.180</b>	0.267	0.153	0.237	0.655	0.021	0.180	0.384	0.375	0.599
<b>ML-SVMDT<sub>F1</sub></b>	0.370	0.192	0.266	0.155	0.256	0.660	0.014	0.175	0.387	0.375	0.590
<b>ML-SVMDT<sub>pre</sub></b>	0.371	0.259	0.253	0.156	0.264	0.652	0.013	0.183	0.407	0.372	0.594
<b>ML-SVMDT<sub>post</sub></b>	0.396	0.259	0.246	0.162	0.273	0.666	0.016	0.184	0.382	0.376	0.577
<b>PCT</b>	0.386	0.389	0.264	0.612	0.392	0.776	0.306	0.220	0.783	0.592	0.817
<b>ML-kNN</b>	0.406	0.242	0.234	0.279	0.280	0.706	0.190	0.182	0.576	0.416	0.639
<b>RAkEL</b>	0.396	0.197	0.254	0.312	0.290	0.758	0.047	0.234	DNF	DNF	DNF
<b>ECC</b>	0.426	0.213	0.249	0.315	0.247	0.992	0.052	0.242	0.666	DNF	DNF
<b>RFML-C4.5</b>	0.277	0.232	0.250	0.243	0.219	0.644	0.071	0.171	0.544	0.368	0.607
<b>RF-HHPCT</b>	0.272	0.225	0.239	0.150	0.219	0.686	0.011	0.162	0.375	<b>0.328</b>	<b>0.534</b>
<b>RF-PCT</b>	<b>0.262</b>	0.210	0.248	0.174	0.221	0.608	<b>0.006</b>	<b>0.159</b>	0.433	0.332	0.541

**Табела А.14** Перформанси на методите за повеќезначна класификација и рангирање во однос на *Coverage* евалуациската мерка (DNF – *Did Not Finish* кај методите означува дека тие не конструирале предиктивен модел во временски период од една недела на расположливите ресурси)

	emotions	scene	yeast	medical	enron	corel5k	tmc2007	mediamill	bibtex	delicious	bookmarks
<b>BR</b>	2.307	<b>0.399</b>	6.330	1.610	12.530	104.800	1.311	20.481	20.926	530.126	DNF
<b>CC</b>	2.317	0.417	6.439	1.471	12.437	105.428	1.302	20.333	21.078	537.388	DNF
<b>CLR</b>	2.386	0.423	6.286	2.036	11.763	<b>91.506</b>	1.363	24.247	<b>18.540</b>	DNF	DNF
<b>TSVM</b>	2.356	0.401	6.335	1.544	12.563	105.376	1.317	20.532	20.963	DNF	DNF
<b>TSCCM</b>	2.351	0.405	6.347	1.586	12.501	100.234	1.296	20.527	20.965	DNF	DNF
<b>TSPCCM</b>	2.361	0.401	6.358	1.532	12.440	104.958	1.291	20.470	20.965	DNF	DNF
<b>QWML</b>	2.807	0.631	8.659	1.832	22.746	206.880	2.796	28.982	57.343	DNF	DNF
<b>HOMER</b>	2.634	0.739	7.285	5.324	24.190	250.800	2.369	47.046	65.626	933.956	DNF
<b>ML-C4.5</b>	2.069	0.945	7.105	3.033	17.010	279.900	2.671	22.096	58.016	620.155	58.353
<b>ML-SVMDT<sub>HL</sub></b>	2.331	<b>0.399</b>	6.482	1.886	12.926	102.345	1.437	19.188	26.993	535.346	45.266
<b>ML-SVMDT<sub>F1</sub></b>	2.232	0.408	6.453	1.923	12.520	100.342	1.437	19.102	26.567	538.789	40.323
<b>ML-SVMDT<sub>pre</sub></b>	2.163	0.578	6.626	2.114	13.420	107.776	1.438	19.292	29.094	546.584	46.867
<b>ML-SVMDT<sub>post</sub></b>	2.144	0.599	6.549	2.108	13.575	108.250	1.453	19.212	25.648	541.022	42.855
<b>PCT</b>	2.356	0.964	6.705	5.813	14.920	115.676	4.572	20.456	58.599	691.622	73.780
<b>ML-kNN</b>	2.490	0.569	6.414	2.844	13.181	113.046	2.155	18.719	56.266	589.898	54.528
<b>RAkEL</b>	2.465	0.635	7.983	8.520	30.509	340.398	2.498	56.617	DNF	DNF	DNF
<b>ECC</b>	2.619	0.625	7.153	7.994	27.760	348.160	2.494	58.865	87.841	DNF	DNF
<b>RFML-C4.5</b>	<b>1.801</b>	0.495	6.276	1.889	12.485	110.356	1.416	16.868	32.580	624.572	40.903
<b>RF-HHPCT</b>	1.866	0.495	<b>6.177</b>	<b>1.402</b>	<b>11.668</b>	101.930	1.235	<b>16.824</b>	20.870	<b>488.441</b>	<b>34.122</b>
<b>RF-PCT</b>	1.827	0.461	6.179	1.619	12.074	107.412	<b>1.219</b>	16.926	25.854	504.999	34.185

**Табела А.15** Перформанси на методите за повеќезначна класификација и рангирање во однос на *загуба при рангирање* евалуациската мерка (DNF – *Did Not Finish* кај методите означува дека тие не конструирале предиктивен модел во временски период од една недела на расположливите ресурси)

	emotions	scene	yeast	medical	enron	corel5k	tmc2007	mediamill	bibtex	delicious	bookmarks
<b>BR</b>	0.246	<b>0.060</b>	0.164	0.021	0.084	0.117	0.003	0.061	0.068	0.114	DNF
<b>CC</b>	0.245	0.064	0.170	<b>0.019</b>	0.083	0.118	0.003	0.062	0.067	0.117	DNF
<b>CLR</b>	0.264	0.065	<b>0.163</b>	0.028	0.078	<b>0.100</b>	0.005	0.092	<b>0.065</b>	DNF	DNF
<b>TSVM</b>	0.252	<b>0.060</b>	<b>0.163</b>	0.020	0.083	0.117	0.004	0.061	0.067	DNF	DNF
<b>TSCCM</b>	0.250	0.061	0.166	0.021	0.083	0.116	0.003	0.061	0.068	DNF	DNF
<b>TSPCCM</b>	0.253	<b>0.060</b>	0.166	0.020	0.083	0.116	0.003	0.061	0.068	DNF	DNF
<b>QWML</b>	0.331	0.103	0.296	0.027	0.177	0.245	0.039	0.101	0.207	DNF	DNF
<b>HOMER</b>	0.297	0.119	0.205	0.090	0.183	0.352	0.028	0.177	0.255	0.379	DNF
<b>ML-C4.5</b>	0.210	0.169	0.225	0.048	0.120	0.479	0.043	0.073	0.260	0.174	0.194
<b>ML-SVMDT<sub>HL</sub></b>	0.269	<b>0.060</b>	0.182	0.027	0.089	0.117	0.006	0.058	0.098	0.116	0.146
<b>ML-SVMDT<sub>F1</sub></b>	0.270	0.075	0.183	0.026	0.090	0.117	0.004	0.056	0.092	0.117	0.136
<b>ML-SVMDT<sub>pre</sub></b>	0.226	0.096	0.184	0.031	0.092	0.120	0.005	0.058	0.107	0.119	0.152
<b>ML-SVMDT<sub>post</sub></b>	0.230	0.100	0.179	0.031	0.096	0.119	0.006	0.058	0.091	0.117	0.137
<b>PCT</b>	0.270	0.174	0.199	0.104	0.114	0.140	0.100	0.063	0.255	0.172	0.258
<b>ML-kNN</b>	0.283	0.093	0.172	0.045	0.093	0.130	0.031	0.055	0.217	0.129	0.181
<b>RAkEL</b>	0.281	0.104	0.259	0.159	0.283	0.673	0.031	0.236	DNF	DNF	DNF
<b>ECC</b>	0.310	0.103	0.224	0.152	0.238	0.749	0.032	0.258	0.394	DNF	DNF
<b>RFML-C4.5</b>	0.153	0.079	0.173	0.028	0.083	0.122	0.007	<b>0.047</b>	0.126	0.140	0.129
<b>RF-HHPCT</b>	0.163	0.079	0.165	<b>0.019</b>	<b>0.077</b>	0.114	<b>0.001</b>	<b>0.047</b>	0.072	<b>0.102</b>	<b>0.103</b>
<b>RF-PCT</b>	<b>0.151</b>	0.072	0.167	0.024	0.079	0.117	0.006	<b>0.047</b>	0.093	0.106	0.104

**Табела А.16** Перформанси на методите за повеќезначна класификација и рангирање во однос на *просечна прецизност* евалуациската мерка (DNF – *Did Not Finish* кај методите означува дека тие не конструирале предиктивен модел во временски период од една недела на расположливите ресурси)

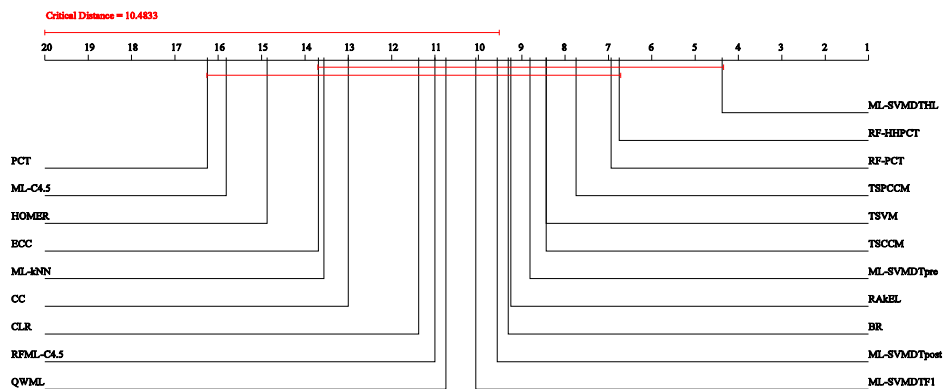
	emotions	scene	yeast	medical	enron	core5k	tmc2007	mediamill	bibtex	delicious	bookmarks
<b>BR</b>	0.721	<b>0.893</b>	0.768	0.896	0.693	0.303	0.978	0.686	0.597	0.351	DNF
<b>CC</b>	0.724	0.881	0.755	0.901	0.695	0.293	0.981	0.672	<b>0.599</b>	0.343	DNF
<b>CLR</b>	0.718	0.886	0.768	0.864	0.699	0.352	0.972	0.450	0.579	DNF	DNF
<b>TSVM</b>	0.728	0.891	<b>0.769</b>	0.899	0.695	0.331	0.976	0.674	0.598	DNF	DNF
<b>TSCCM</b>	0.732	0.891	0.765	<b>0.903</b>	0.698	<b>0.358</b>	0.984	0.679	0.598	DNF	DNF
<b>TSPCCM</b>	0.731	0.892	0.764	<b>0.903</b>	<b>0.700</b>	0.307	0.985	0.677	0.598	DNF	DNF
<b>QWML</b>	0.679	0.864	0.698	0.862	0.604	0.311	0.938	0.492	0.498	DNF	DNF
<b>HOMER</b>	0.698	0.848	0.740	0.786	0.604	0.222	0.945	0.583	0.407	0.231	DNF
<b>ML-C4.5</b>	0.759	0.751	0.706	0.823	0.629	0.196	0.842	0.669	0.392	0.321	0.378
<b>ML-SVMDT<sub>HL</sub></b>	0.721	<b>0.893</b>	0.760	0.872	0.681	0.312	0.979	0.703	0.551	0.364	0.420
<b>ML-SVMDT<sub>F1</sub></b>	0.720	0.788	0.750	0.856	0.643	0.307	0.987	0.715	0.555	0.358	0.426
<b>ML-SVMDT<sub>pre</sub></b>	0.747	0.843	0.746	0.872	0.668	0.306	0.985	0.698	0.536	0.362	0.421
<b>ML-SVMDT<sub>post</sub></b>	0.742	0.840	0.749	0.871	0.658	0.304	0.983	0.697	0.557	0.362	0.435
<b>PCT</b>	0.713	0.745	0.724	0.522	0.546	0.208	0.700	0.654	0.212	0.206	0.213
<b>ML-kNN</b>	0.694	0.851	0.758	0.784	0.635	0.266	0.844	0.703	0.349	0.326	0.381
<b>RAkEL</b>	0.713	0.862	0.715	0.676	0.522	0.088	0.939	0.492	DNF	DNF	DNF
<b>ECC</b>	0.687	0.856	0.734	0.684	0.576	0.014	0.935	0.453	0.228	DNF	DNF
<b>RFML-C4.5</b>	<b>0.812</b>	0.862	0.749	0.817	0.680	0.314	0.945	0.728	0.418	0.359	0.423
<b>RF-HHPCT</b>	0.805	0.864	0.760	0.887	<b>0.700</b>	0.313	0.992	0.735	0.579	<b>0.398</b>	<b>0.481</b>
<b>RF-PCT</b>	<b>0.812</b>	0.874	0.757	0.868	0.698	0.334	<b>0.996</b>	<b>0.737</b>	0.525	0.395	0.480

**Табела А.17** Перформанси на методите за повеќезначна класификација и рангирање во однос на *времињата на тренирање* (DNF – *Did Not Finish* кај методите означува дека тие не конструирале предиктивен модел во временски период од една недела на расположливите ресурси)

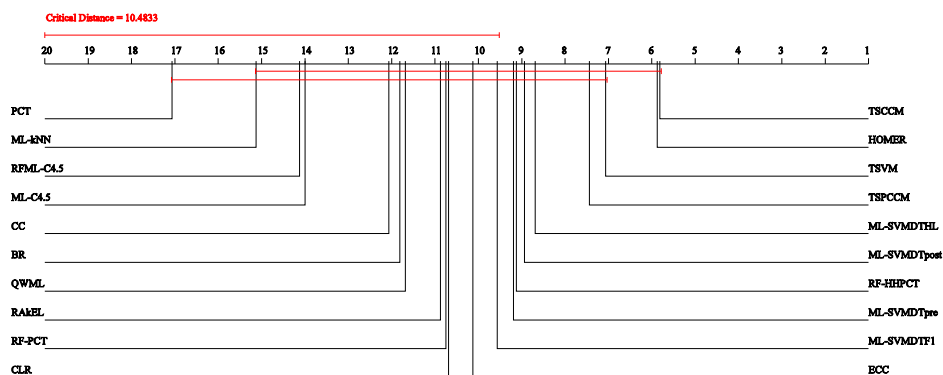
	emotions	scene	yeast	medical	enron	core5k	tmc2007	mediamill	bibtex	delicious	bookmarks
<b>BR</b>	4.0	71.0	145.0	18.0	318.0	926.0	42645.0	85468.0	11013.0	57053.0	DNF
<b>CC</b>	6.0	99.0	206.0	28.0	440.0	1225.0	46704.0	100435.0	12434.0	84903.0	DNF
<b>CLR</b>	10.0	195.0	672.0	40.0	971.0	2388.0	52427.0	260156.0	13424.0	DNF	DNF
<b>TSVM</b>	10.0	195.0	672.0	40.0	971.0	2388.0	52427.0	260156.0	13424.0	DNF	DNF
<b>TSCCM</b>	11.0	215.0	698.0	48.0	1038.0	2450.0	52688.0	261801.0	13982.0	DNF	DNF
<b>TSPCCM</b>	11.0	208.0	685.0	44.0	1020.0	2430.0	52567.0	261132.0	13642.0	DNF	DNF
<b>QWML</b>	10.0	195.0	672.0	40.0	971.0	2388.0	52427.0	260156.0	13424.0	DNF	DNF
<b>HOMER</b>	4.0	68.0	101.0	16.0	158.0	771.0	31300.0	78195.0	2896.0	21218.0	DNF
<b>ML-C4.5</b>	0.3	8.0	14.0	3.0	15.0	369.0	469.0	2030.0	566.0	2738.0	4039.0
<b>ML-SVMDT<sub>HL</sub></b>	4.0	29.0	86.0	14.0	97.0	732.0	920.0	10345.0	1304.0	1945.0	160981.0
<b>ML-SVMDT<sub>F1</sub></b>	2.3	28.0	52.0	10.0	83.0	793.0	1469.0	14035.0	2156.0	2314.0	17194.0
<b>ML-SVMDT<sub>pre</sub></b>	2.0	25.0	42.0	8.0	77.0	274.0	888.0	9015.0	767.0	1168.0	53737.0
<b>ML-SVMDT<sub>post</sub></b>	2.0	25.0	49.0	8.0	70.0	645.0	862.0	10789.0	769.0	1358.0	137660.0
<b>PCT</b>	<b>0.1</b>	<b>2.0</b>	<b>1.5</b>	<b>0.6</b>	<b>1.1</b>	<b>30.0</b>	<b>11.5</b>	<b>440.0</b>	<b>16.4</b>	<b>70.0</b>	<b>965.0</b>
<b>ML-kNN</b>	0.4	14.0	8.2	1.0	6.0	389.0	737.0	1094.0	124.0	236.0	15990.0
<b>RAkEL</b>	5.0	79.0	157.0	82.0	493.0	3380.0	102394.0	33554.0	DNF	DNF	DNF
<b>ECC</b>	4.9	319.0	497.0	103.0	1467.0	20073.0	92169.0	188957.0	29578.0	DNF	DNF
<b>RFML-C4.5</b>	1.2	10.0	19.0	7.0	25.0	385.0	460.0	4056.0	645.0	21776.0	5602.0
<b>RF-HHPCT</b>	3.1	26.0	29.0	28.0	51.0	921.0	602.0	9224.0	1682.0	5628.0	37654.0
<b>RF-PCT</b>	2.9	23.0	25.0	27.0	47.0	902.0	557.0	8360.0	1550.0	5376.0	28900.0

**Табела А.18** Перформанси на методите за повеќезначна класификација и рангирање во однос на *времињата на тестирање* (DNF – *Did Not Finish* кај методите означува дека тие не конструирале предиктивен модел во временски период од една недела на расположливите ресурси)

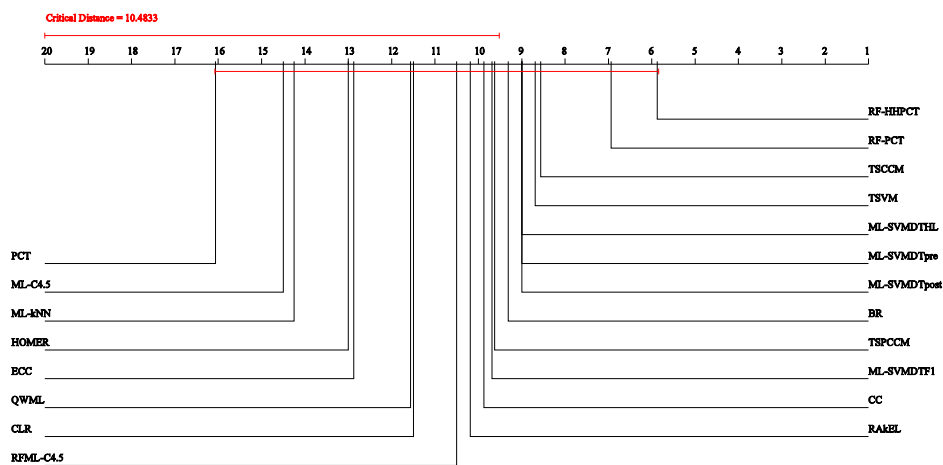
	emoions	scene	yeast	medical	enron	core5k	tmc2007	mediamill	bibtex	delicious	bookmarks
<b>BR</b>	1.0	25.0	23.0	4.0	50.0	25.0	927.0	6152.0	654.0	2045.0	DNF
<b>CC</b>	1.0	25.0	25.0	6.0	53.0	31.0	891.0	6125.0	661.0	1872.0	DNF
<b>CLR</b>	3.0	87.0	153.0	90.0	634.0	2161.0	3282.0	76385.0	16733.0	DNF	DNF
<b>TSVM</b>	1.6	29.0	42.0	5.0	55.0	38.0	1078.0	7125.0	823.0	DNF	DNF
<b>TSCCM</b>	1.6	34.0	43.0	6.9	61.0	44.0	1101.0	7669.0	866.0	DNF	DNF
<b>TSPCCM</b>	1.5	32.0	40.0	6.0	59.0	43.0	1132.0	7601.0	838.0	DNF	DNF
<b>QWML</b>	2.0	40.0	64.0	25.0	174.0	119.0	1543.0	20317.0	4710.0	DNF	DNF
<b>HOMER</b>	1.0	21.0	17.0	1.5	22.0	14.0	730.0	6079.0	155.0	816.0	DNF
<b>ML-C4.5</b>	<b>0.0</b>	1.0	0.1	0.1	0.2	<b>1.0</b>	1.7	<b>1.0</b>	6.5	19.0	21.0
<b>ML-SVMDT<sub>HL</sub></b>	0.4	8.0	4.0	0.7	7.0	8.0	18.0	353.0	48.0	102.0	1480.0
<b>ML-SVMDT<sub>F1</sub></b>	0.2	1.3	4.3	1.4	7.5	9.0	28.0	425.0	69.0	169.0	3074.0
<b>ML-SVMDT<sub>pre</sub></b>	0.6	12.0	10.0	1.8	18.0	9.0	74.0	398.0	84.0	189.0	4189.0
<b>ML-SVMDT<sub>post</sub></b>	0.6	13.0	11.0	1.7	16.0	14.0	65.0	470.0	84.0	182.0	8022.0
<b>PCT</b>	<b>0.0</b>	<b>0.0</b>	<b>0.0</b>	<b>0.0</b>	<b>0.0</b>	<b>1.0</b>	<b>0.0</b>	<b>1.0</b>	<b>0.0</b>	<b>10.0</b>	<b>15.0</b>
<b>ML-kNN</b>	0.4	14.0	5.0	0.2	3.0	45.0	230.0	477.0	64.0	55.0	4084.0
<b>RAkEL</b>	2.0	72.0	70.0	24.0	153.0	3613.0	10985.0	39001.0	DNF	DNF	DNF
<b>ECC</b>	6.6	168.0	158.0	46.0	696.0	2077.0	10865.0	50183.0	10756.0	DNF	DNF
<b>RFML-C4.5</b>	0.1	2.0	0.5	0.5	1.0	1.8	3.4	8.0	12.0	32.0	28.0
<b>RF-HHPCT</b>	0.3	0.9	0.2	0.5	1.1	2.4	2.6	4.2	19.0	44.0	57.0
<b>RF-PCT</b>	0.3	1.0	0.2	0.5	1.0	2.5	2.8	4.0	18.0	48.0	58.0



(а) Хаминг загуба

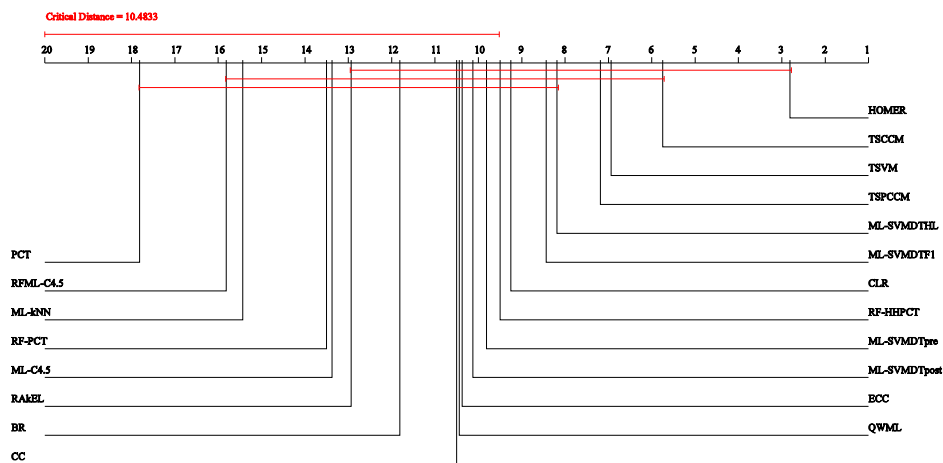


(б) точност

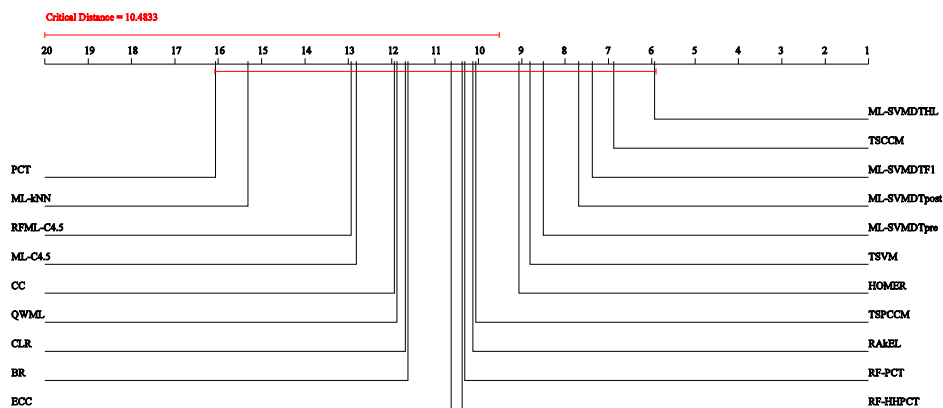


(в) прецизност

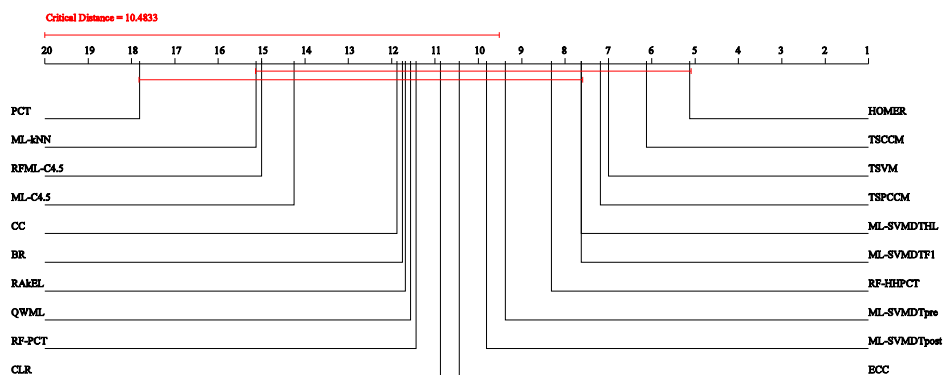
Слика А.1 Критични дијаграми за мерките базирани на примерок (Хаминг загуба, точност и прецизност). Резултати добиени од дополнителниот Немеџи тест со ниво на значајност од 0.5 на податочните множества за кои сите методи дале резултати



(a) *odsub*

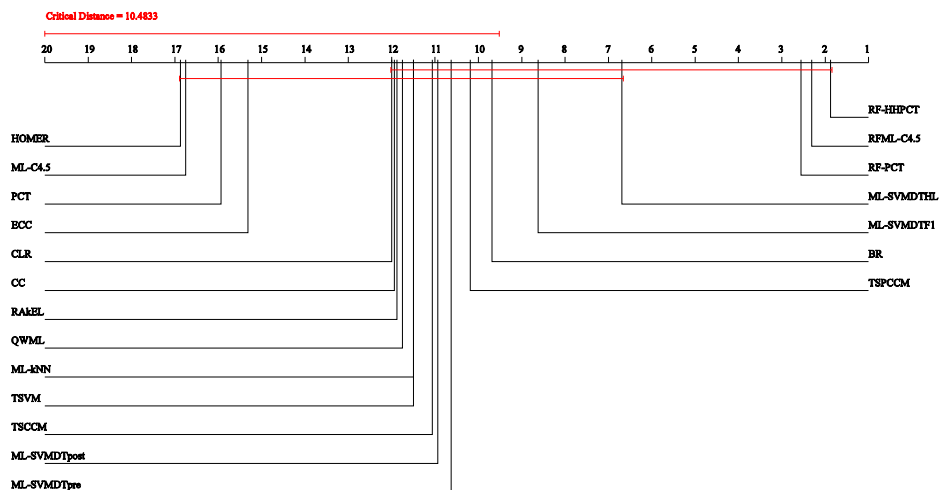


(б) класификациска точност

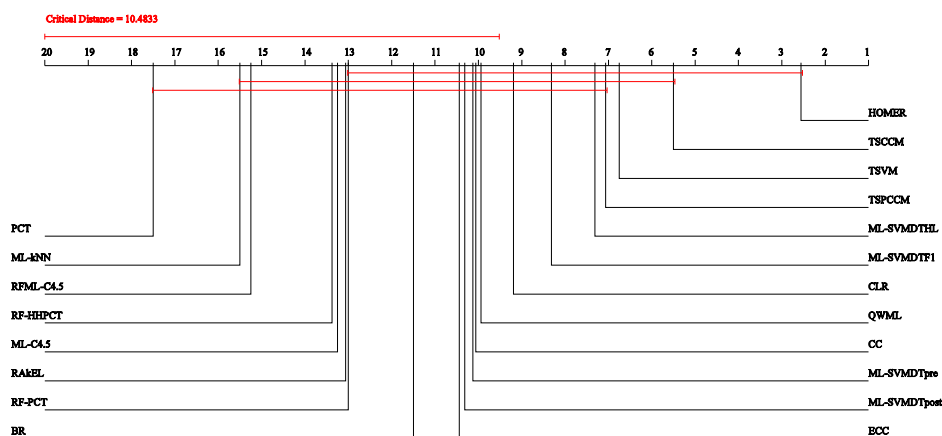


(в)  $F_1$

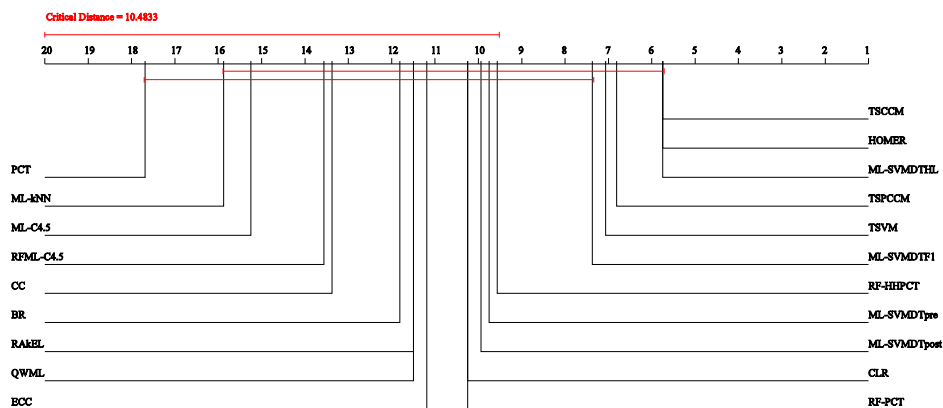
Слика А.2 Критични дијаграми за мерките базирани на примерок (*odsub*, класификациска точност и  $F_1$ ). Резултати добиени од дополнителниот Немењи тест со ниво на значајност од 0.5 на податочните множества за кои сите методи дале резултати



(а) микро прецизност

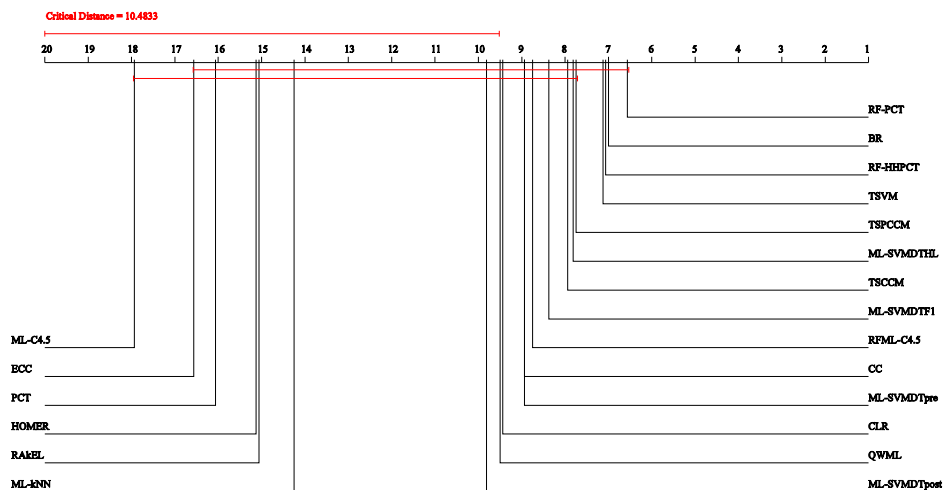


(б) микро одзив

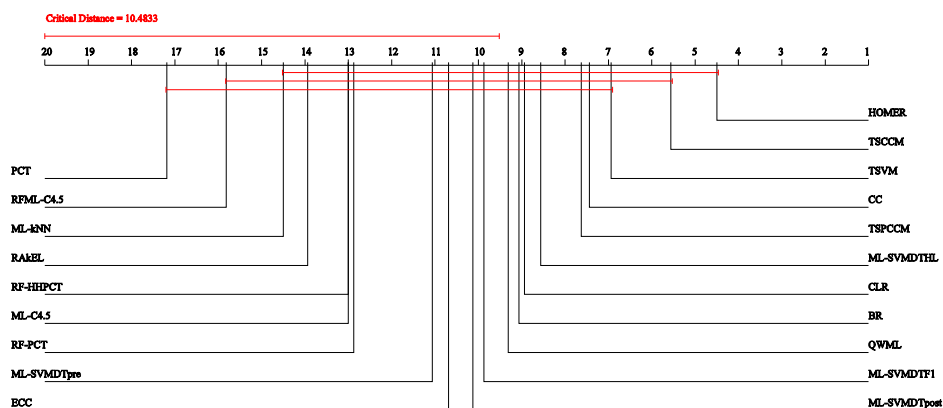


(в) микро  $F_1$

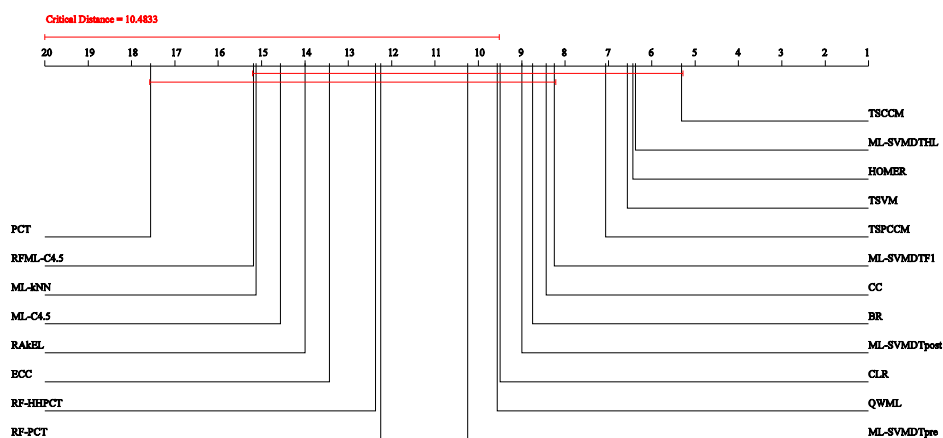
Слика А.3 Критични дијаграми за мерките базирани на ознака (микро прецизност, микро одзив и микро  $F_1$ ). Резултати добиени од дополнителниот Немењи тест со ниво на значајност од 0.5 на податочните множества за кои сите методи дале резултати



(а) макро прецизност

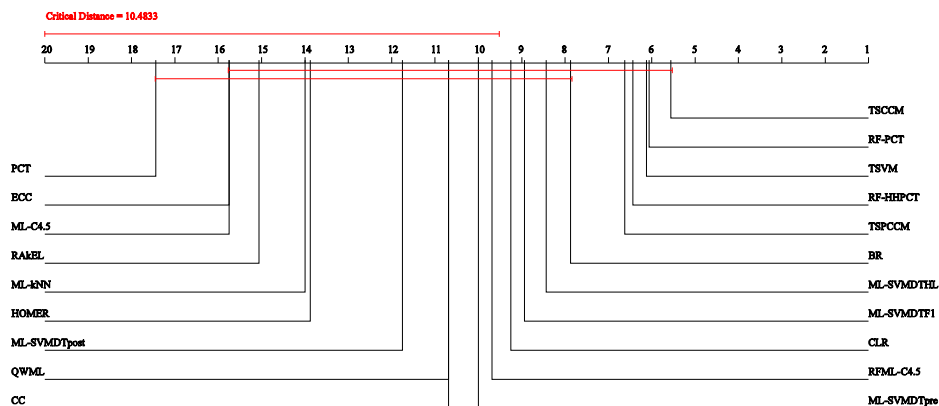


(б) макро одзив

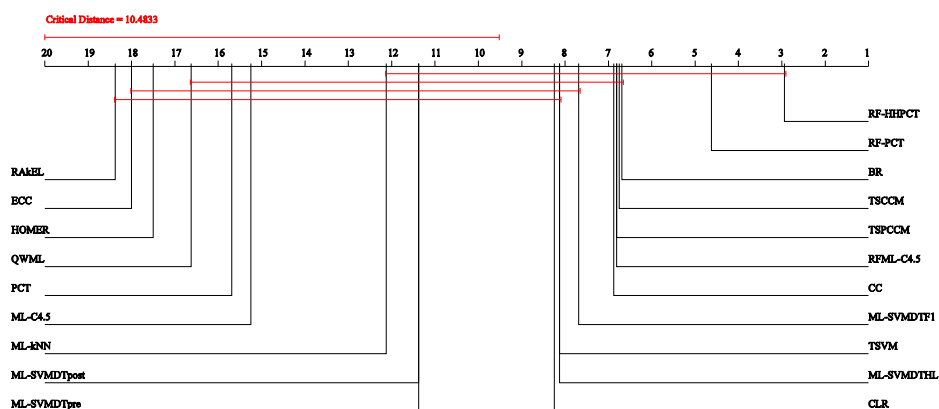


(в) макро  $F_1$

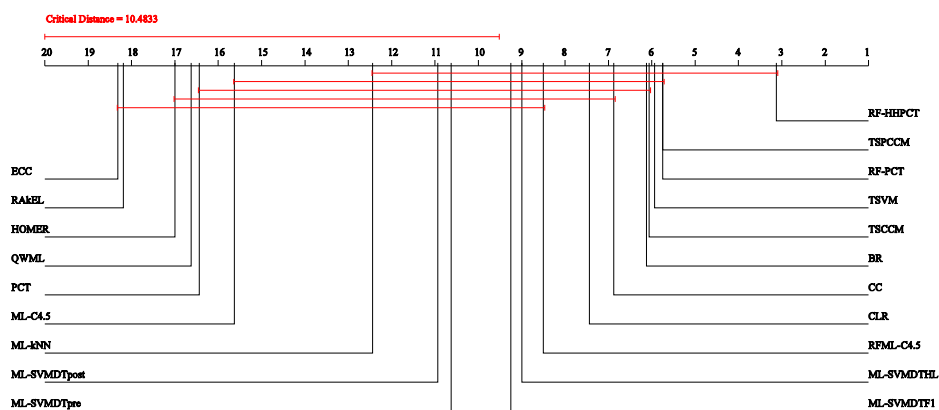
Слика А.4 Критични дијаграми за мерките базирани на ознака (макро прецизност, макро одзив и макро  $F_1$ ). Резултати добиени од дополнителниот Немењи тест со ниво на значајност од 0.5 на податочните множества за кои сите методи дале резултати



(а) една грешка

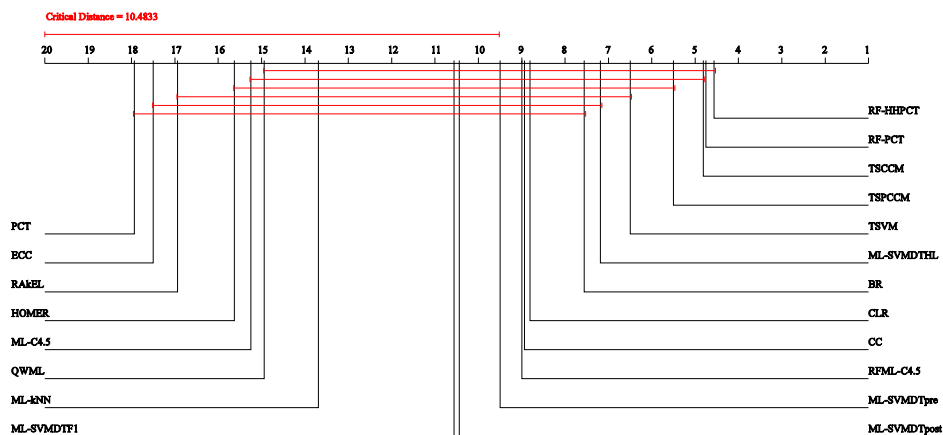


(б) опфатеност

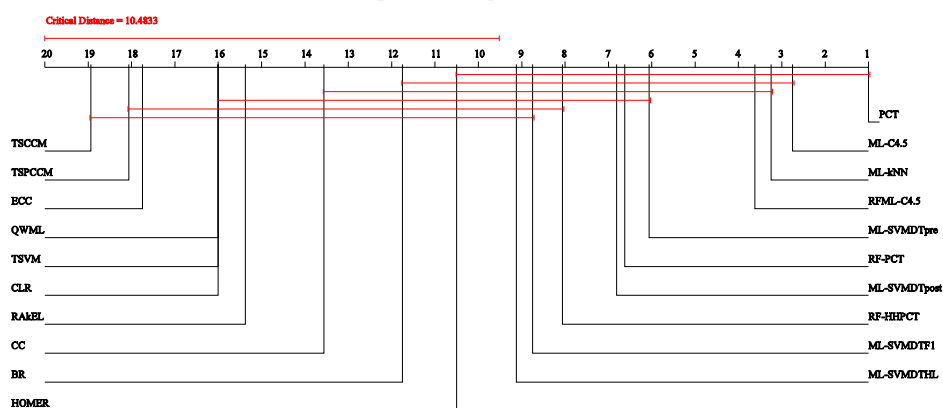


(в) загуба при рангирање

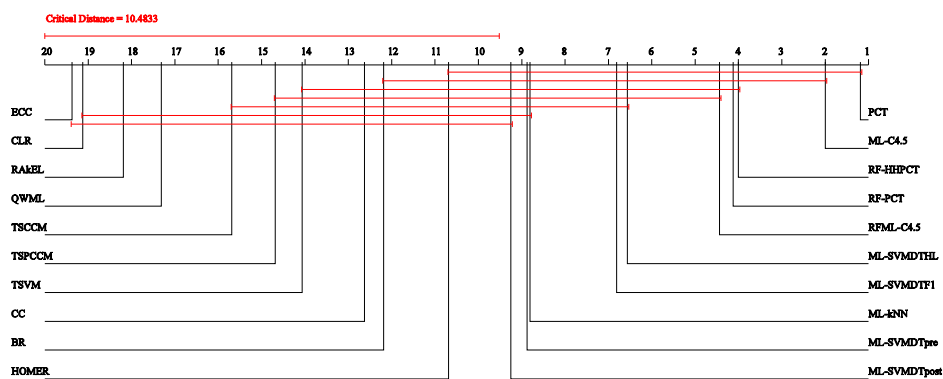
Слика А.5 Критични дијаграми за мерките базирани на рангирање (една грешка, опфатеност и загуба при рангирање). Резултати добиени од дополнителниот Немеџи тест со ниво на значајност од 0.5 на податочните множества за кои сите методи дале резултати



(а) просечна прецизност

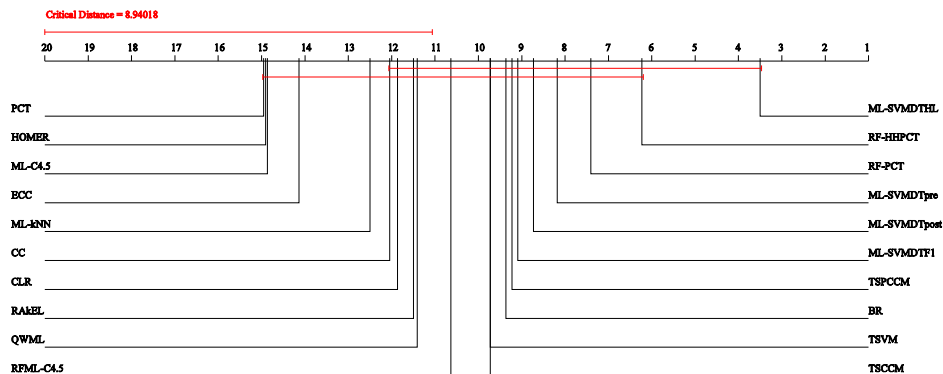


(б) времиња на обување

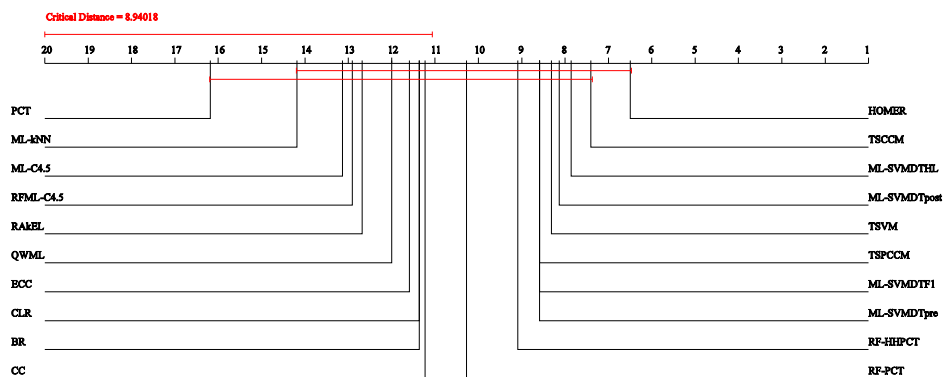


(в) времиња на тестирање

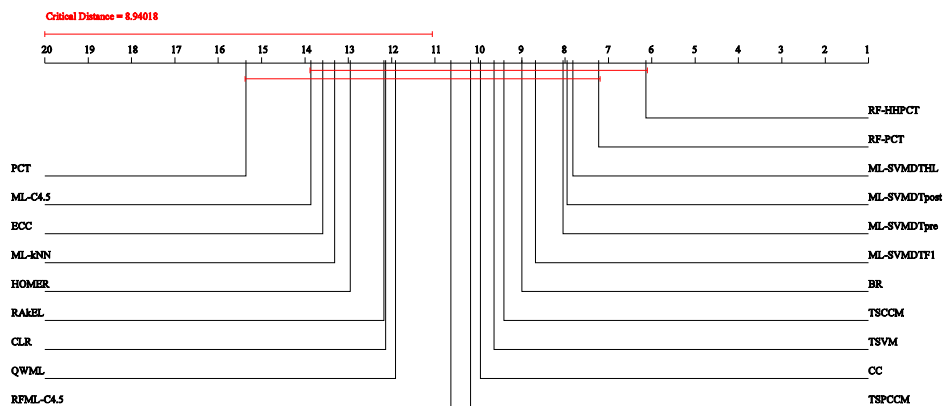
Слика А.6 Критични дијаграми за мерката базирана на рангирање (просечна прецизност) и времињата на обување и тестирање. Резултати добиени од дополнителниот Немењи тест со ниво на значајност од 0.5 на податочните множества за кои сите методи дале резултати



(а) Хаминг загуба

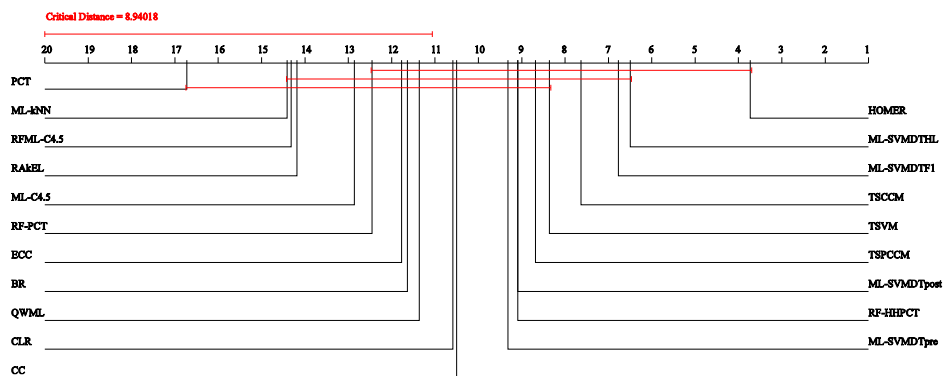


(б) точност

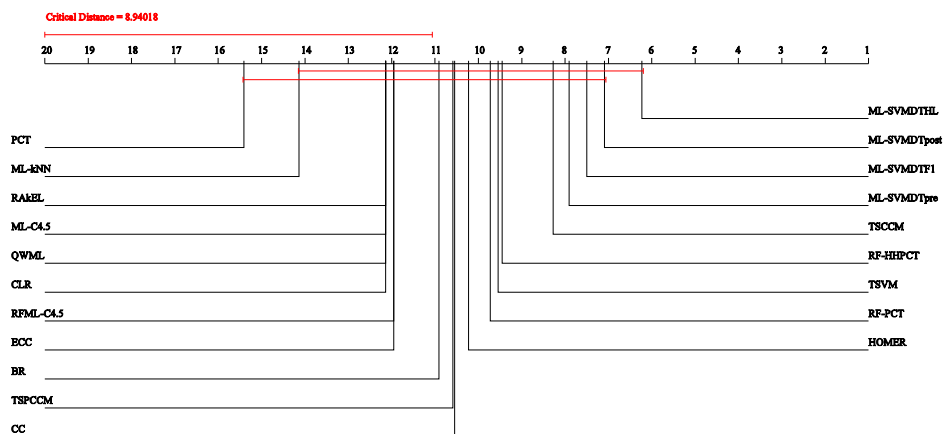


(в) прецизност

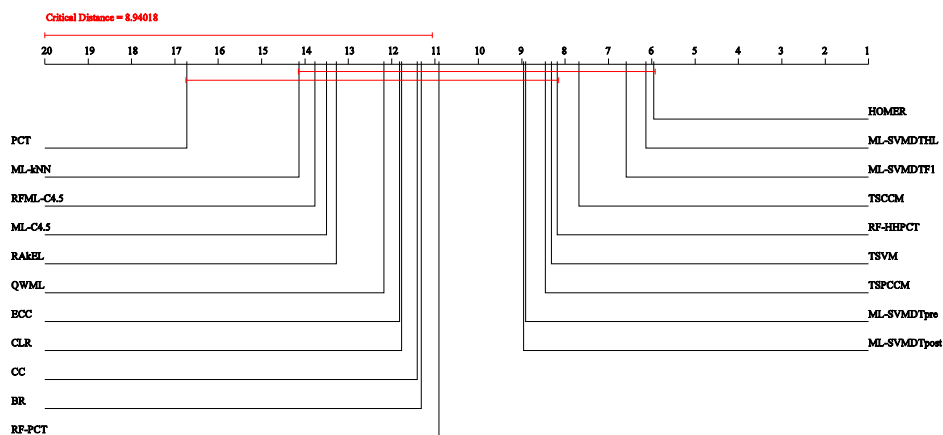
Слика А.7 Критични дијаграми за мерките базирани на примерок (Хаминг загуба, точност и прецизност). Резултати добиени од дополнителниот Немењи тест со ниво на значајност од 0.5 на сите податочни множества



(a) *odsv*

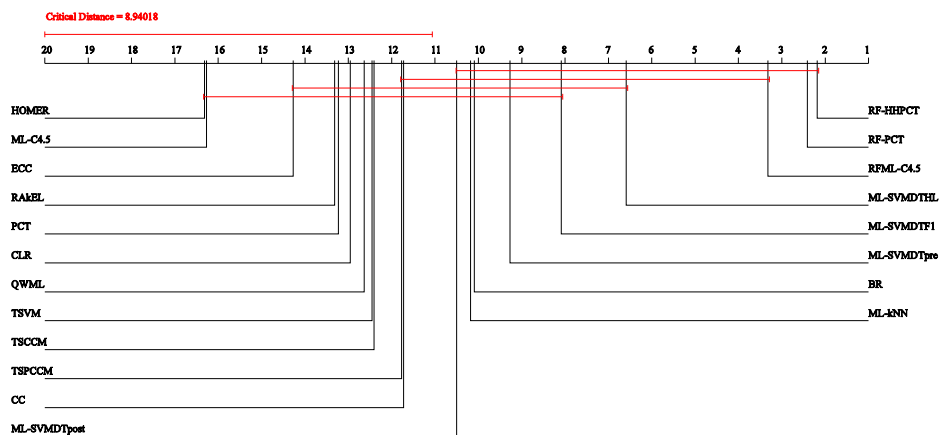


(б) *класификациска точност*

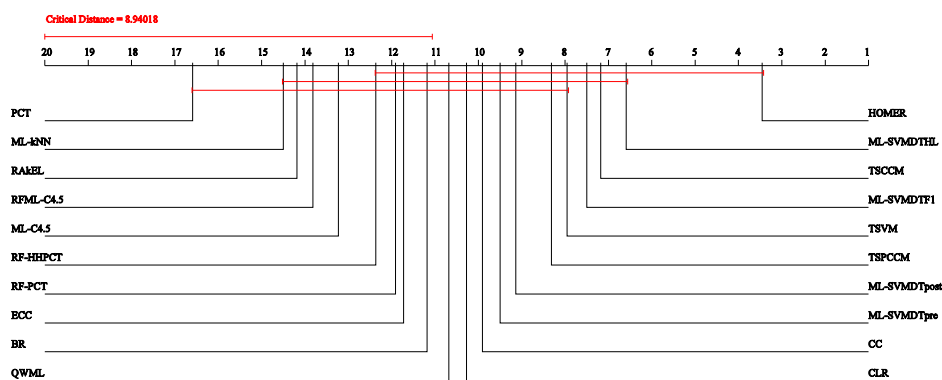


(в)  $F_1$

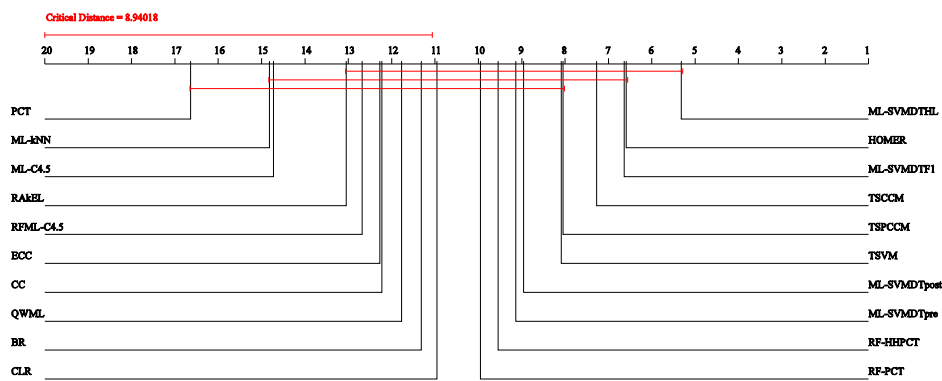
Слика А.8 Критични дијаграми за мерките базирани на примерок (*odsv*, *класификациска точност* и  $F_1$ ). Резултати добиени од дополнителниот Немеџи тест со ниво на значајност од 0.5 на сите податочни множества



(а) микро прецизност

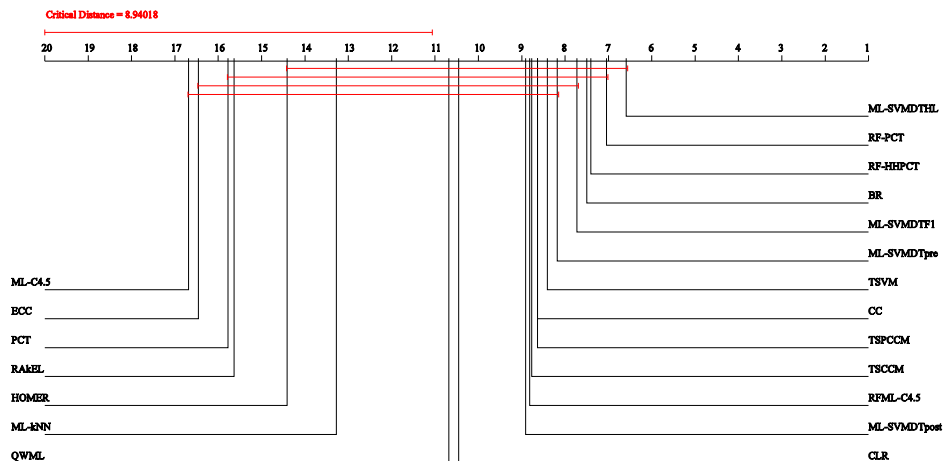


(б) микро одзив

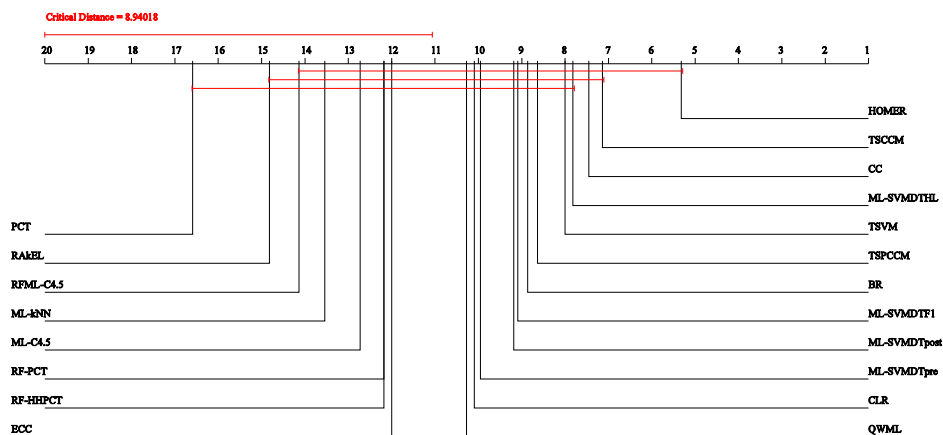


(в) микро  $F_1$

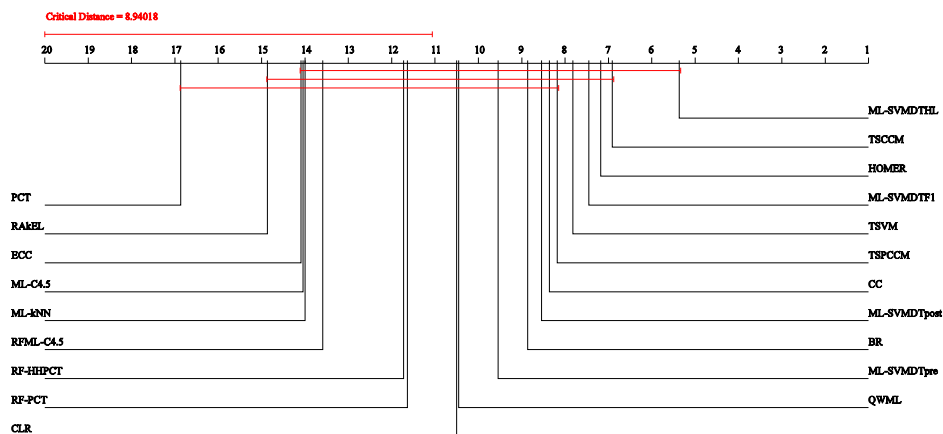
Слика А.9 Критични дијаграми за мерките базирани на ознака (микро прецизност, микро одзив и микро  $F_1$ ). Резултати добиени од дополнителниот Немењи тест со ниво на значајност од 0.5 на сите податочни множества



(а) макро прецизност

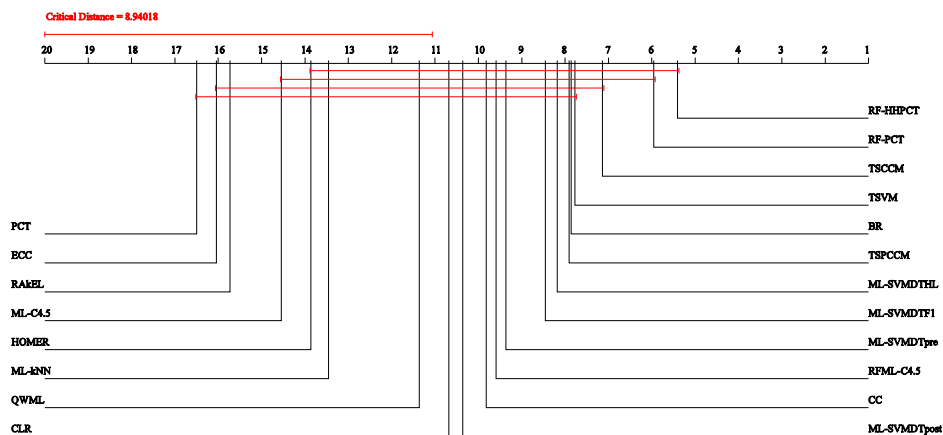


(б) макро одсиг

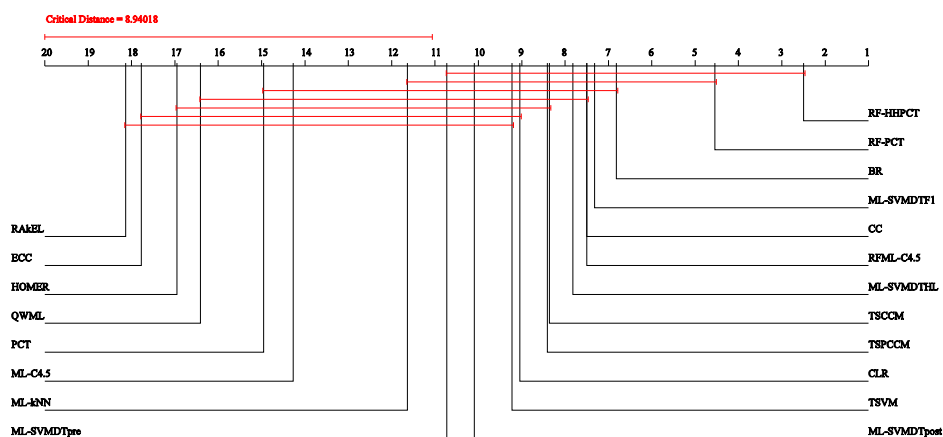


(в) макро  $F_1$

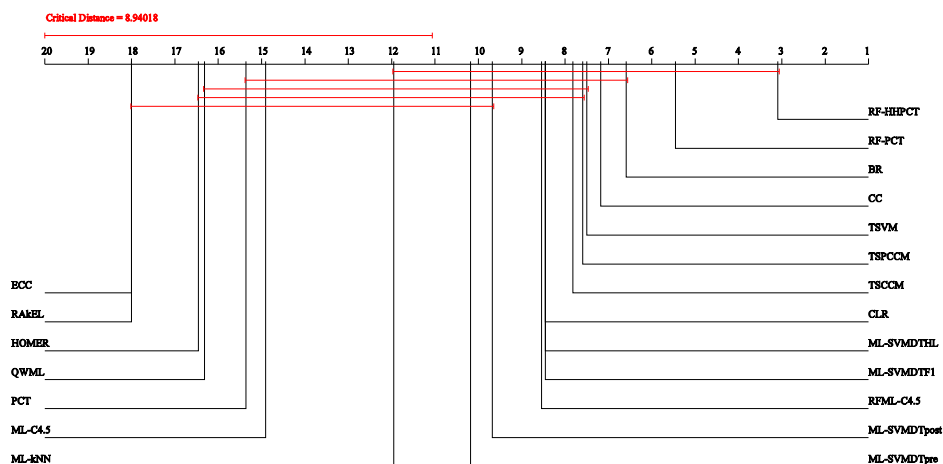
Слика А.10 Критични дијаграми за мерките базирани на ознака (макро прецизност, макро одсиг и макро  $F_1$ ). Резултати добиени од дополнителниот Немеџи тест со ниво на значајност од 0.5 на сите податочни множества



(а) една грешка

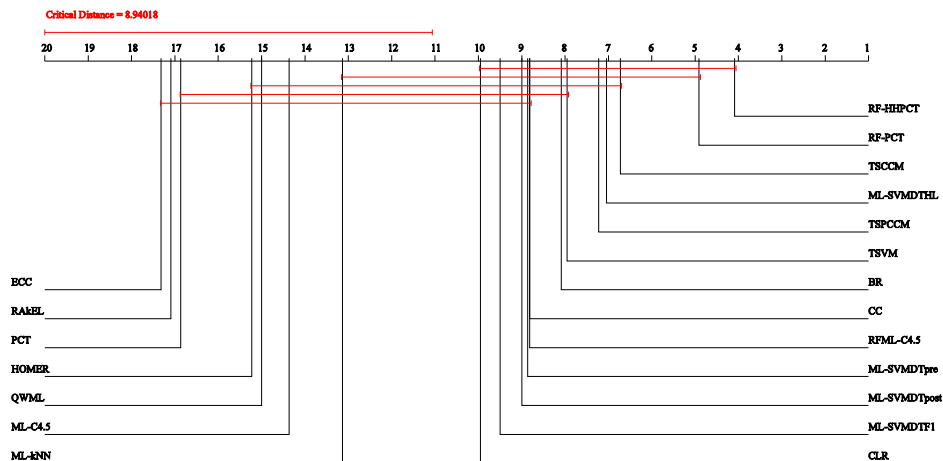


(б) опфатеност

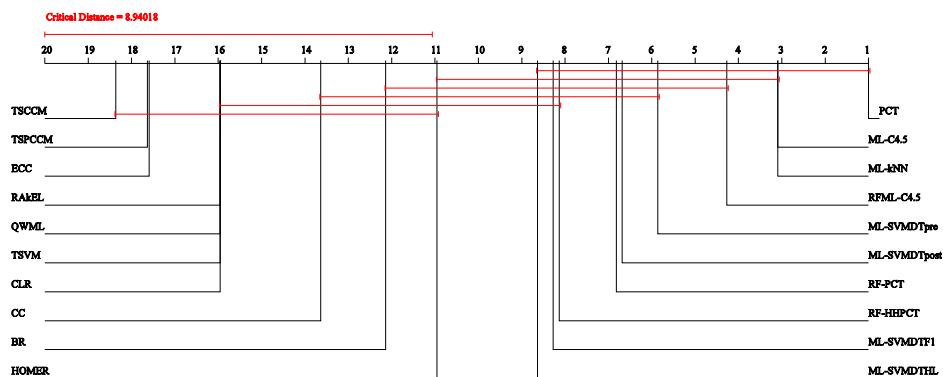


(в) загуба при рангирање

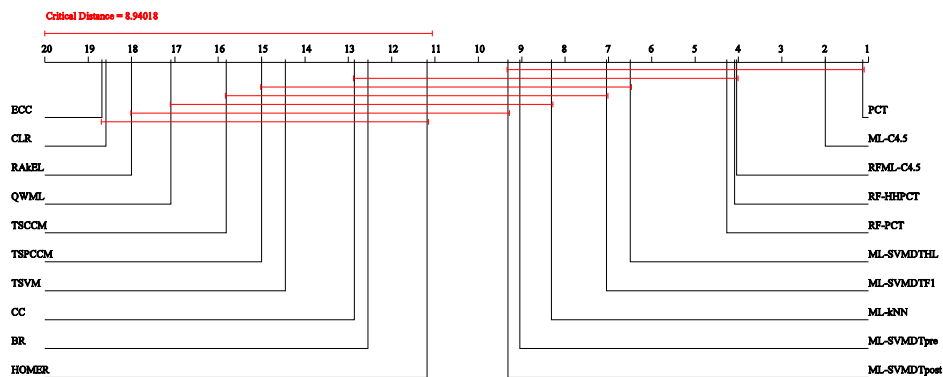
Слика А.11 Критични дијаграми за мерките базирани на рангирање (една грешка, опфатеност и загуба при рангирање). Резултати добиени од дополнителниот Немеџи тест со ниво на значајност од 0.5 на сите податочни множества



(а) просечна прецизност



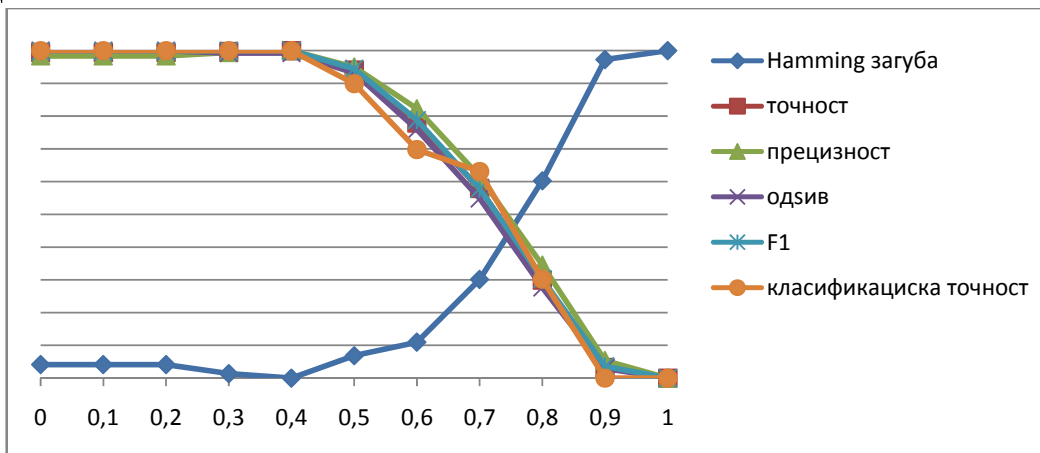
(б) времиња на обукување



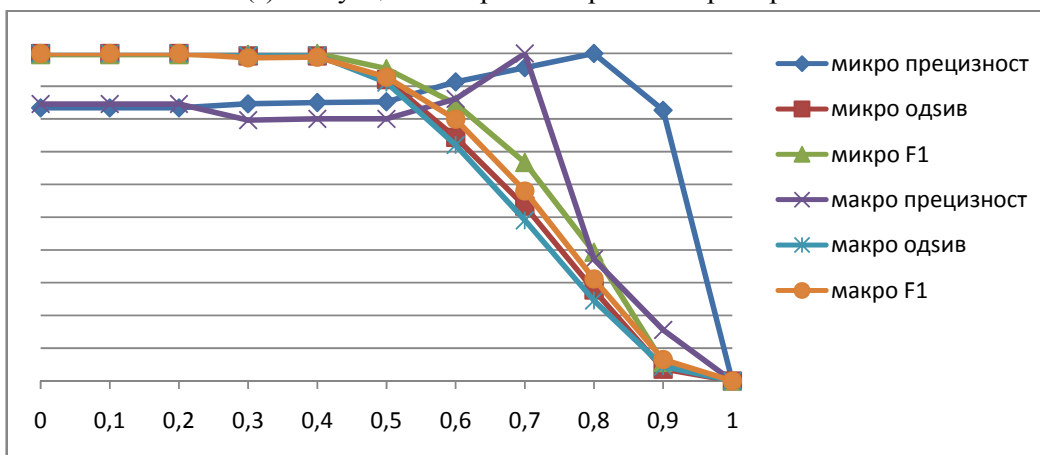
(в) времиња на тестирање

Слика А.12 Критични дијаграми за мерката базирана на рангирање (*просечна прецизност*) и времињата на обукување и тестирање. Резултати добиени од *дополнителниот Немењи тест* со ниво на значајност од 0.5 на сите податочни множества

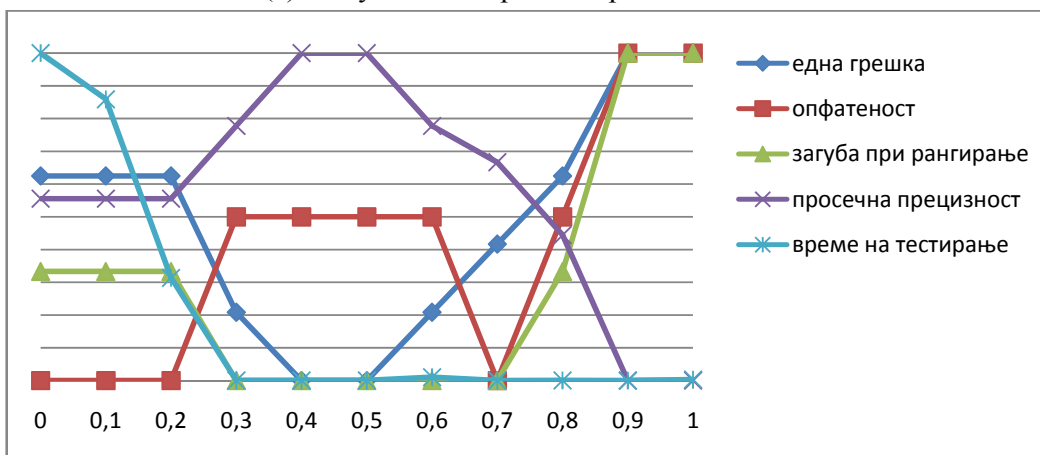
## Додаток Б



(а) евалуациски мерки базирани на примерок

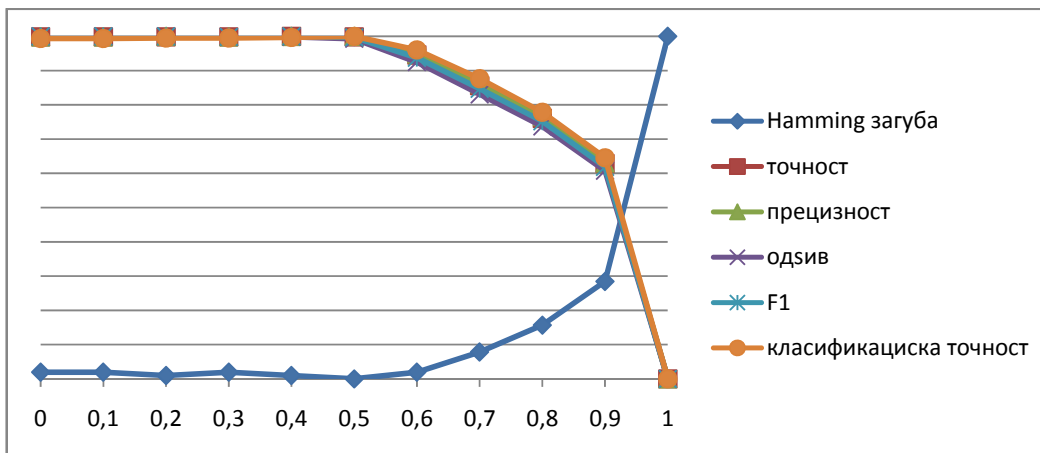


(б) евалуациски мерки базирани на ознака

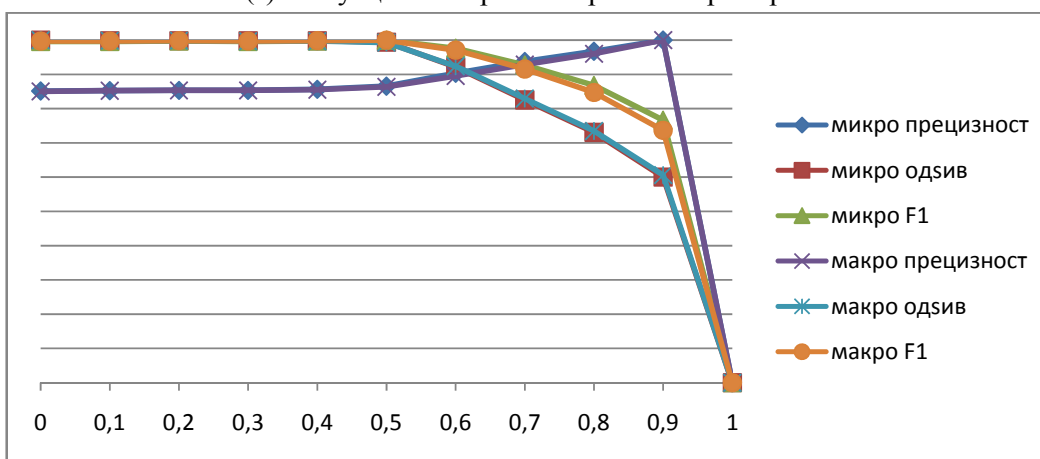


(в) евалуациски мерки базирани на рангирање и време на тестирање

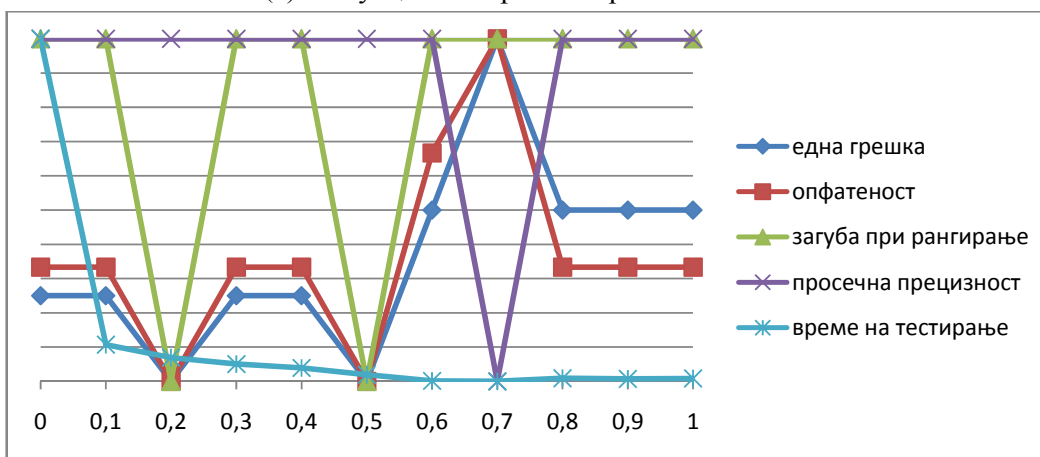
**Слика Б.1** Зависност на перформансите на дво-нивовскиот метод со гласање во однос на прагот  $t$  ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за *emotions* податочното множество



(а) евалуациски мерки базирани на примерок

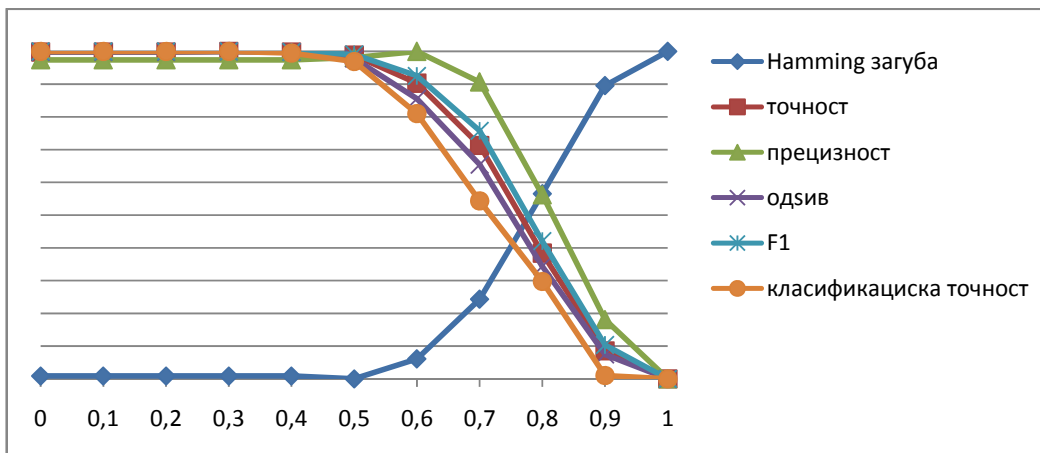


(б) евалуациски мерки базирани на ознака

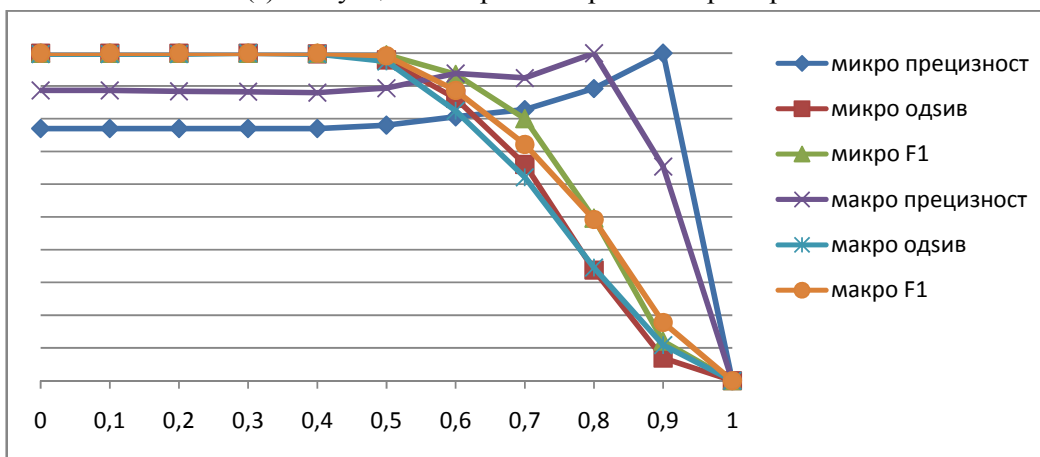


(в) евалуациски мерки базирани на рангирање и време на тестирање

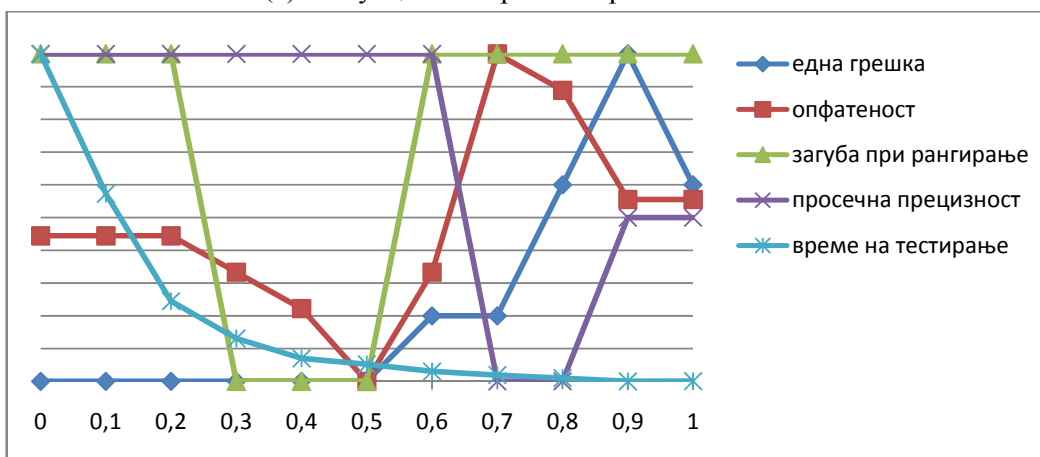
**Слика Б.2** Зависност на перформансите на дво-нивовскиот метод со гласање во однос на прагот  $t$  ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за *scene* податочното множество



(а) евалуациски мерки базирани на примерок

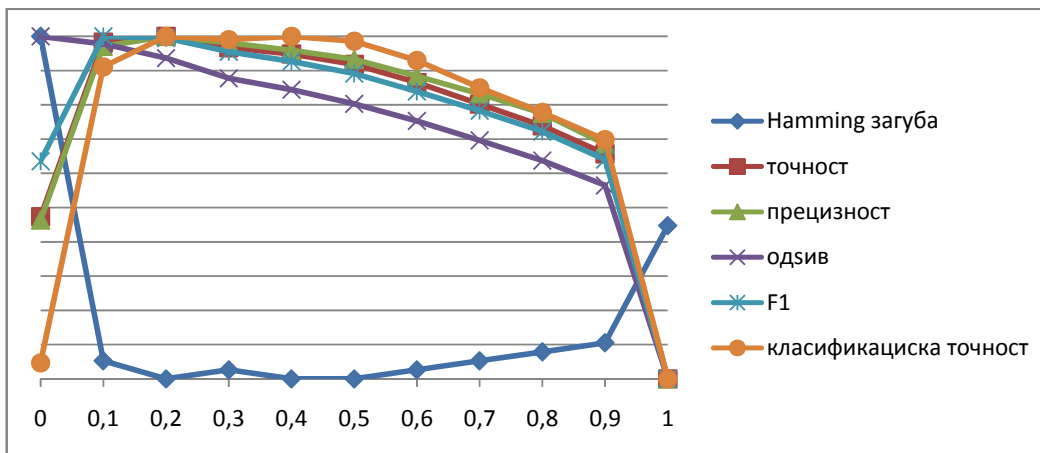


(б) евалуациски мерки базирани на ознака

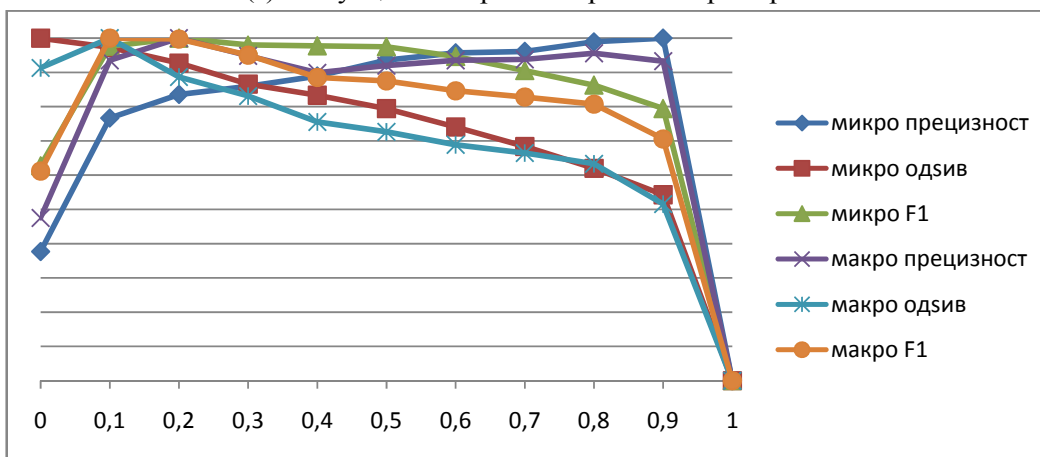


(в) евалуациски мерки базирани на рангирање и време на тестирање

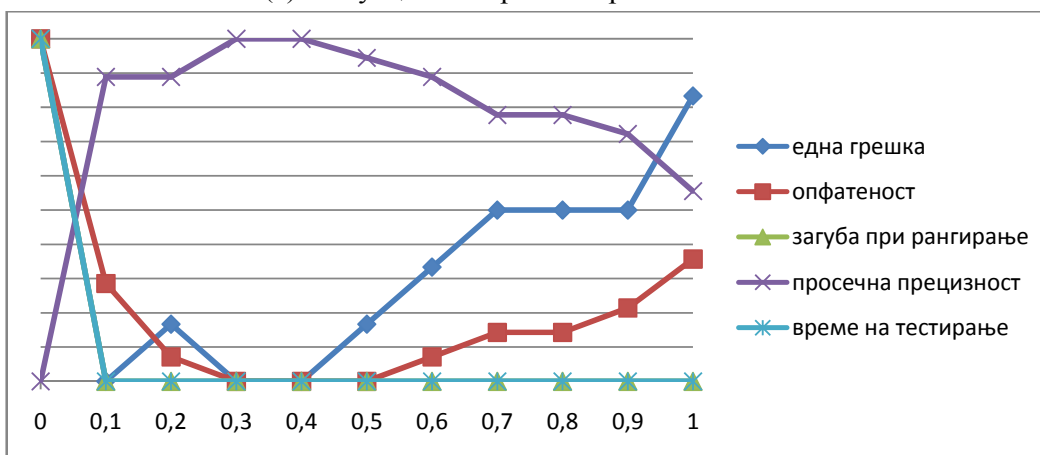
**Слика Б.3** Зависност на перформансите на дво-нивовскиот метод со гласање во однос на прагот  $t$  ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за *yeast* податочното множество



(а) евалуациски мерки базирани на примерок

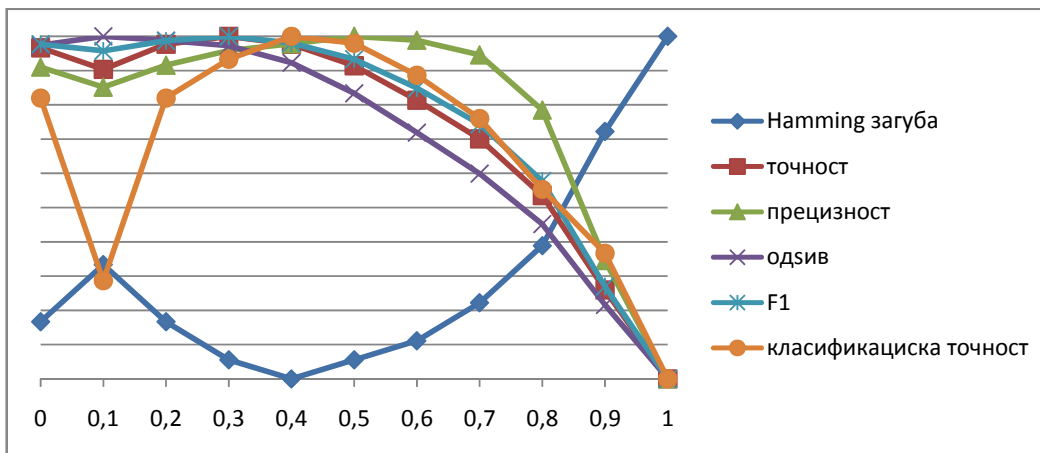


(б) евалуациски мерки базирани на ознака

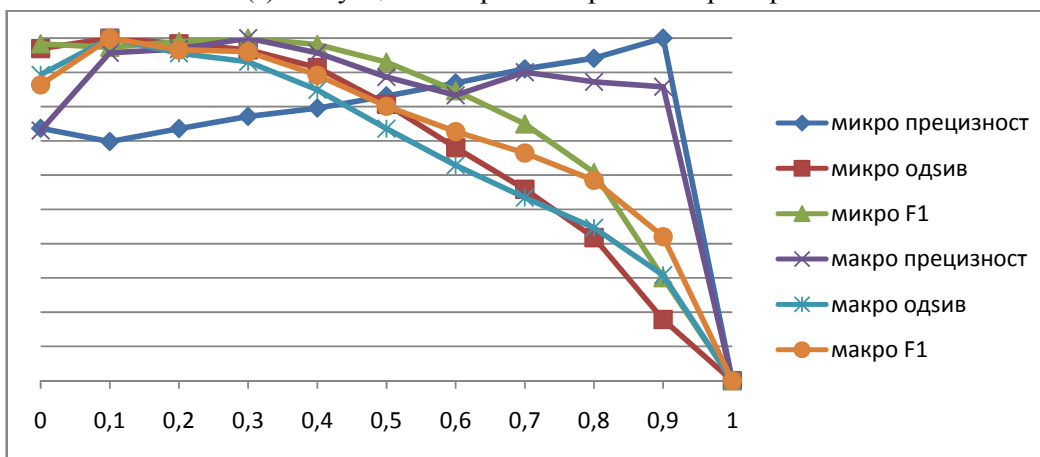


(в) евалуациски мерки базирани на рангирање и време на тестирање

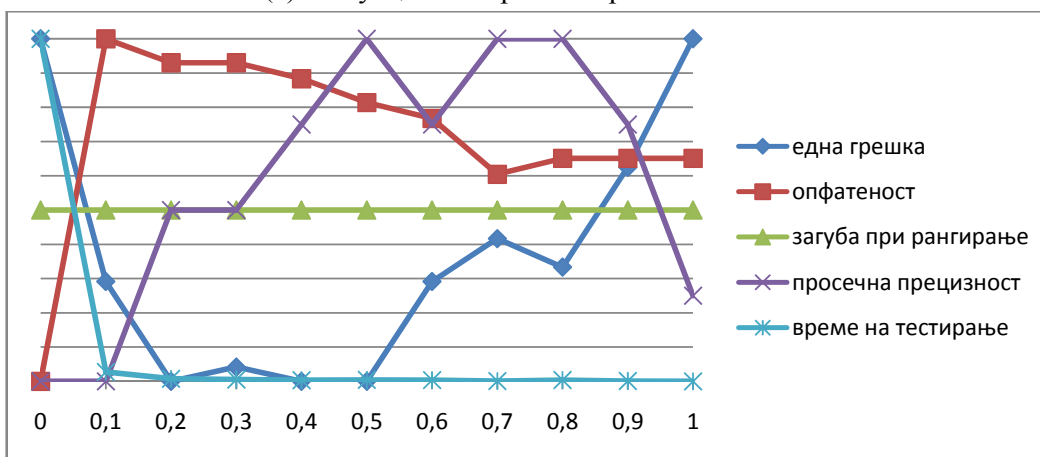
**Слика Б.4** Зависност на перформансите на дво-нивовскиот метод со гласање во однос на прагот  $t$  ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за *medical* податочното множество



(а) евалуациски мерки базирани на примерок

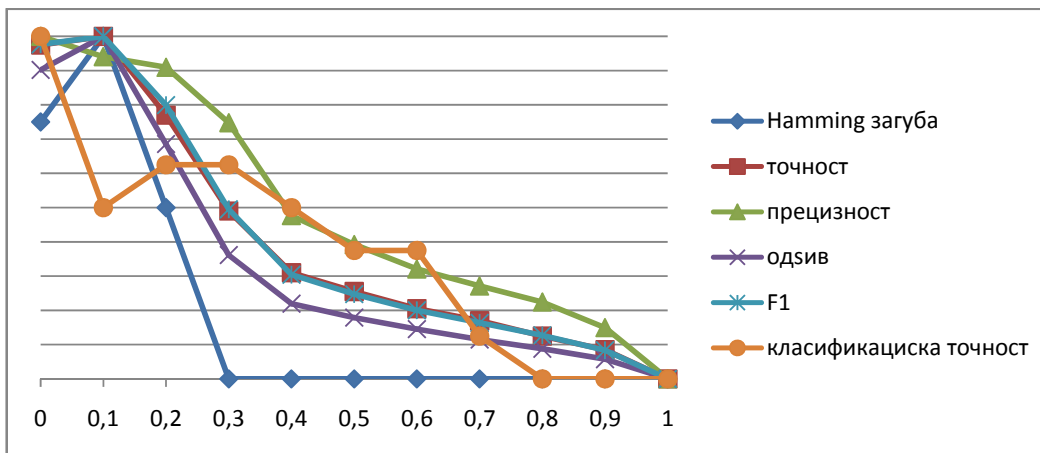


(б) евалуациски мерки базирани на ознака

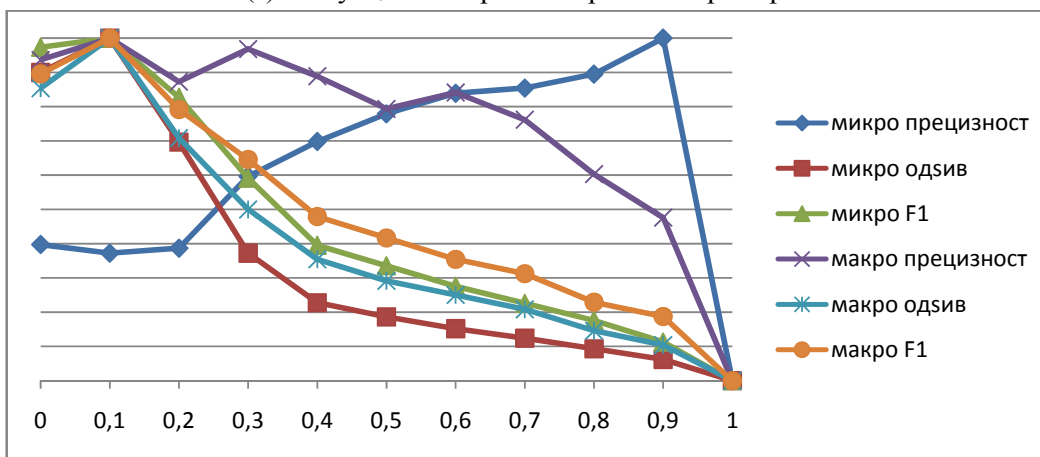


(в) евалуациски мерки базирани на рангирање и време на тестирање

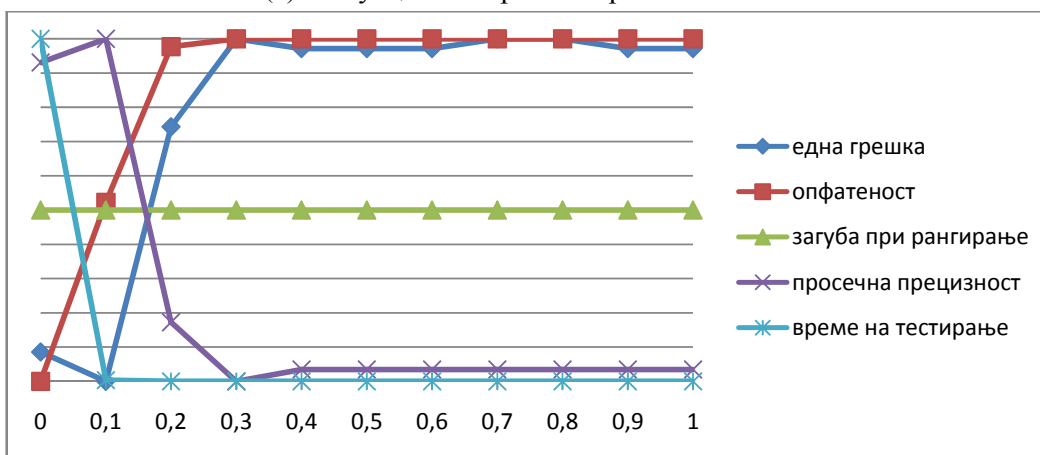
**Слика Б.5** Зависност на перформансите на дво-нивовскиот метод со гласање во однос на прагот  $t$  ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за *enron* податочното множество



(а) евалуациски мерки базирани на примерок

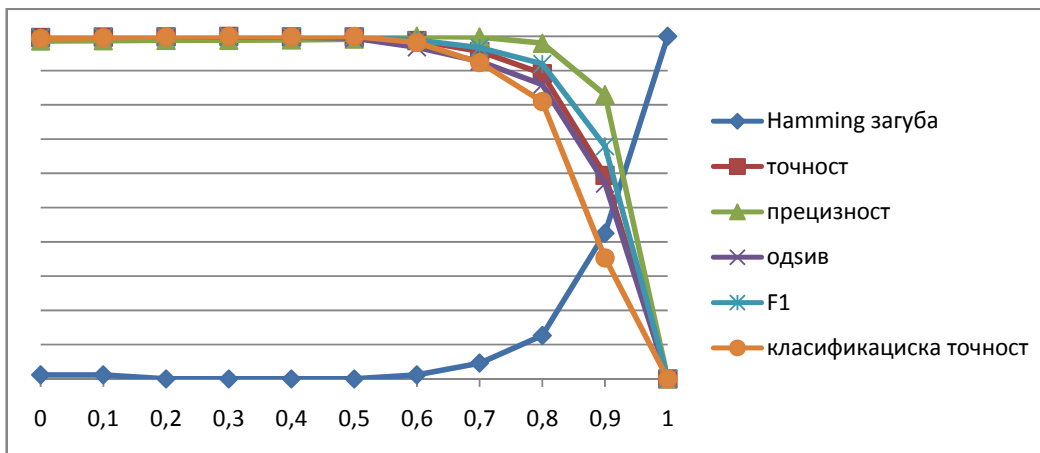


(б) евалуациски мерки базирани на ознака

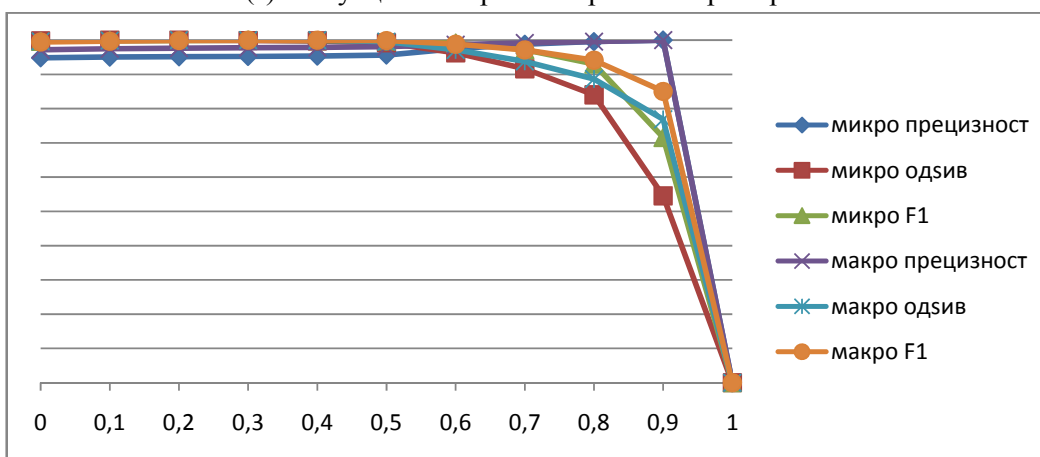


(в) евалуациски мерки базирани на рангирање и време на тестирање

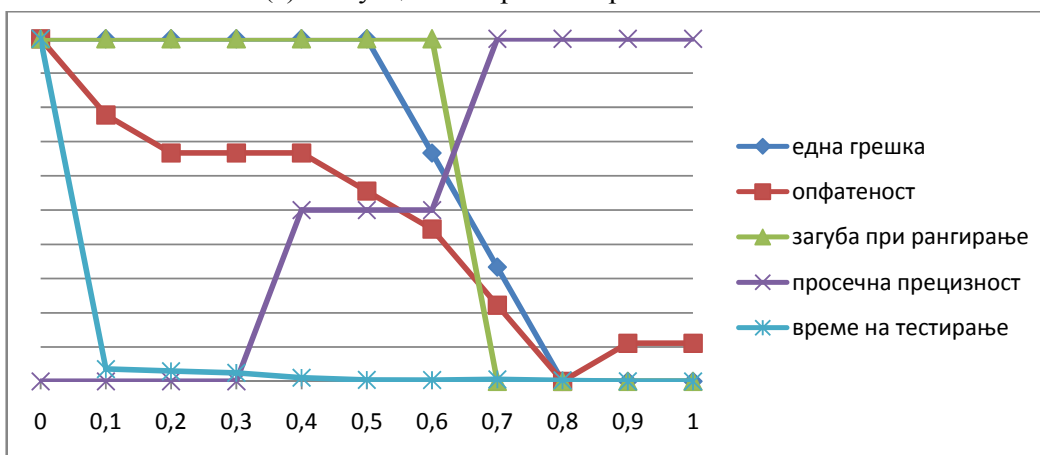
**Слика Б.6** Зависност на перформансите на дво-нивовскиот метод со гласање во однос на прагот  $t$  ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за *corel5k* податочното множество



(а) евалуациски мерки базирани на примерок

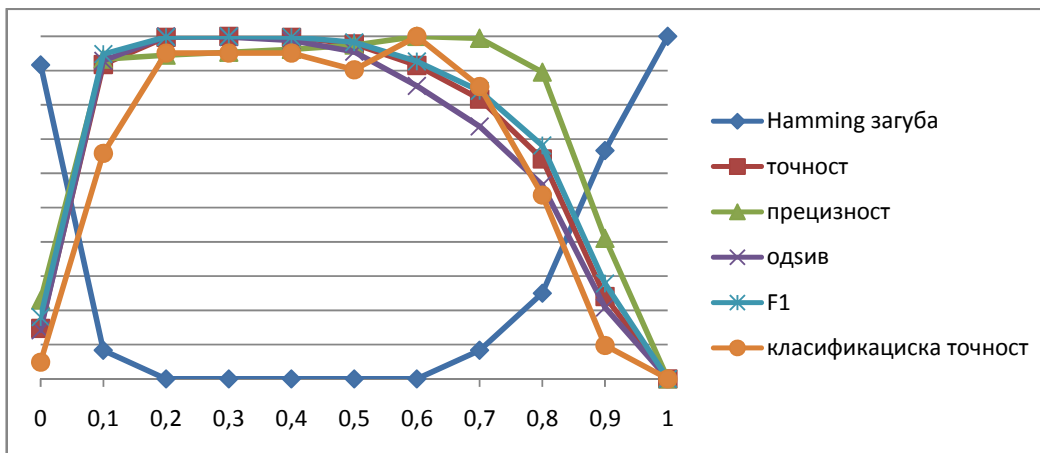


(б) евалуациски мерки базирани на ознака

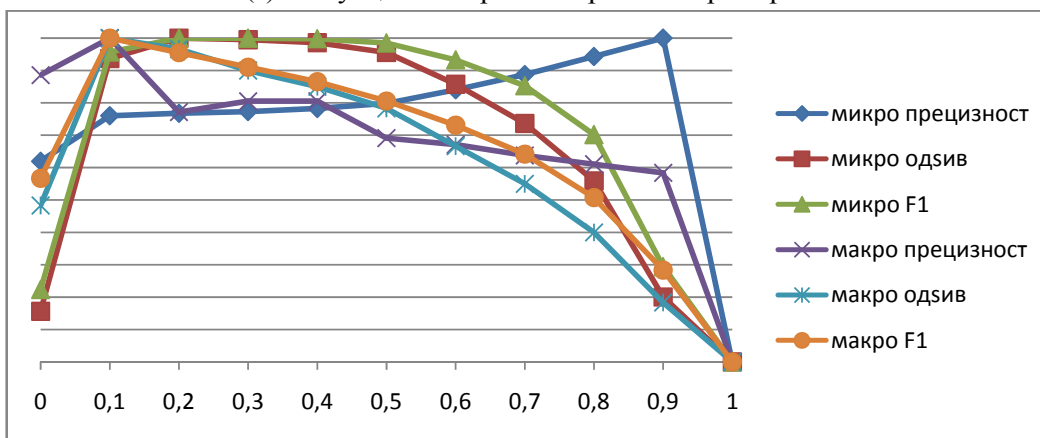


(в) евалуациски мерки базирани на рангирање и време на тестирање

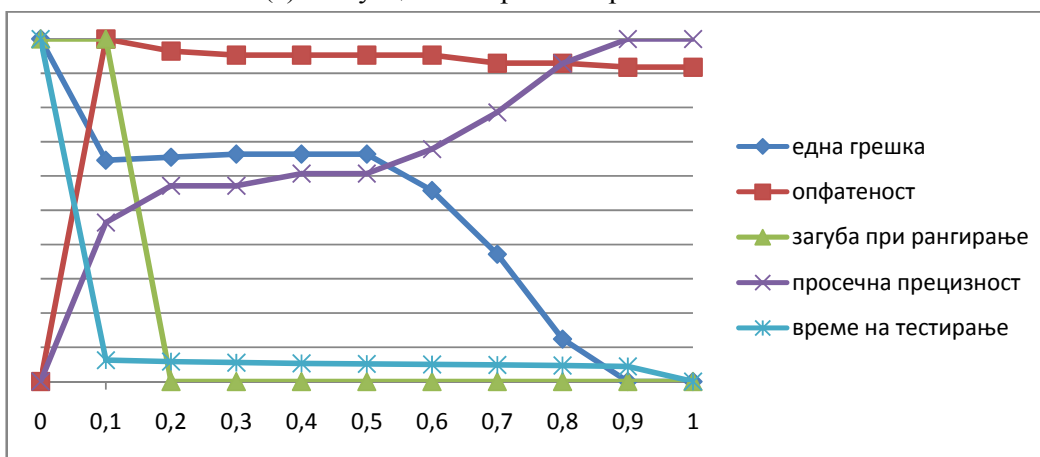
**Слика Б.7** Зависност на перформансите на дво-нивовскиот метод со гласање во однос на прагот  $t$  ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за *tmc2007* податочното множество



(а) евалуациски мерки базирани на примерок

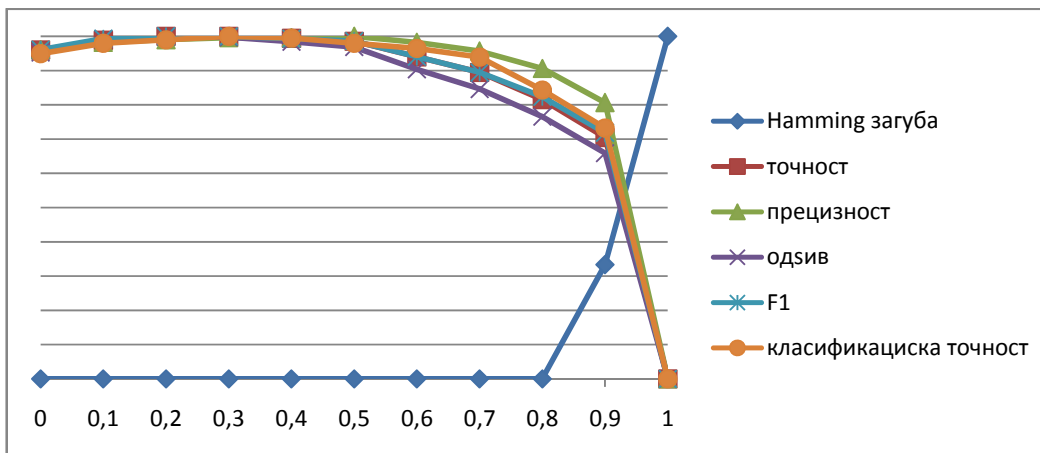


(б) евалуациски мерки базирани на ознака

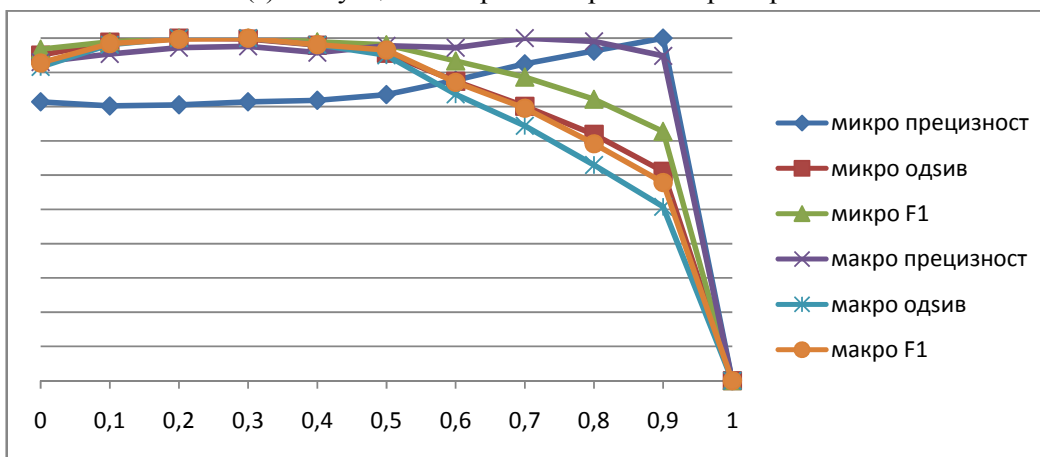


(в) евалуациски мерки базирани на рангирање и време на тестирање

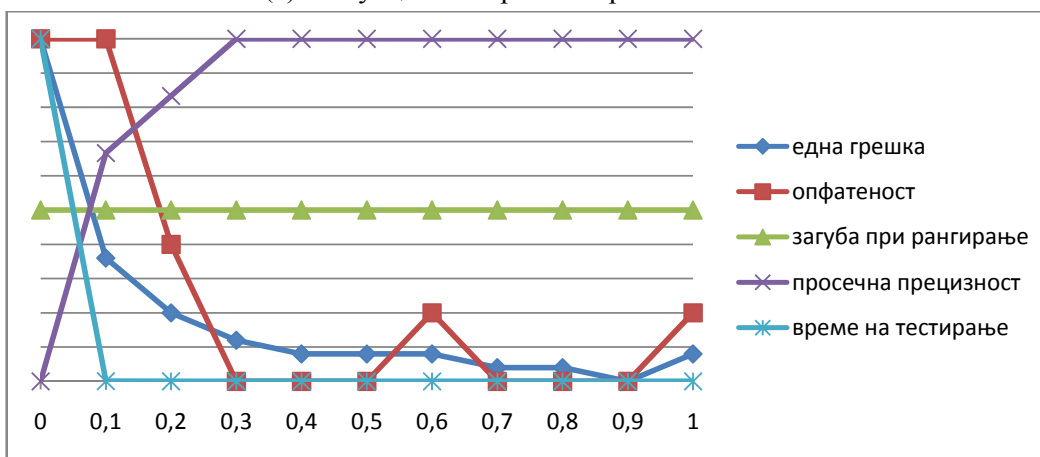
**Слика Б.8** Зависност на перформансите на дво-нивовскиот метод со гласање во однос на прагот  $t$  ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за *mediamill* податочното множество



(а) евалуациски мерки базирани на примерок

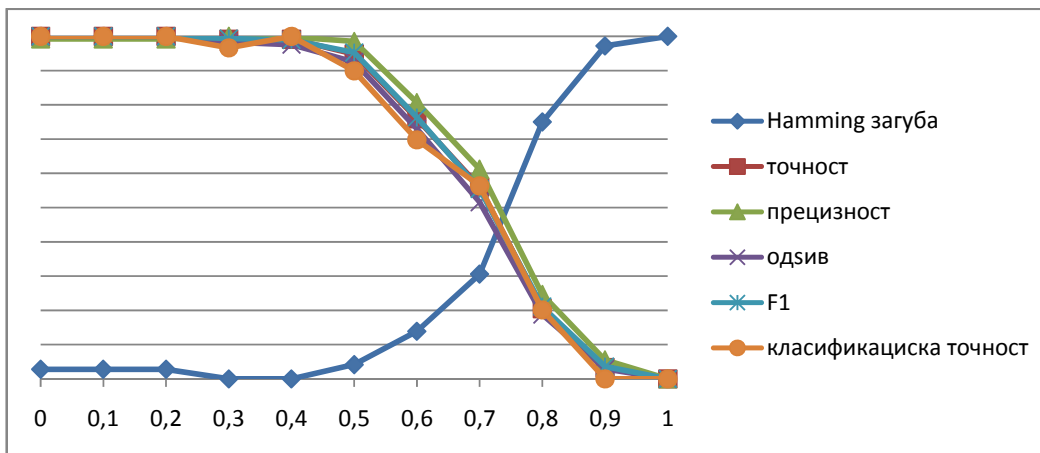


(б) евалуациски мерки базирани на ознака

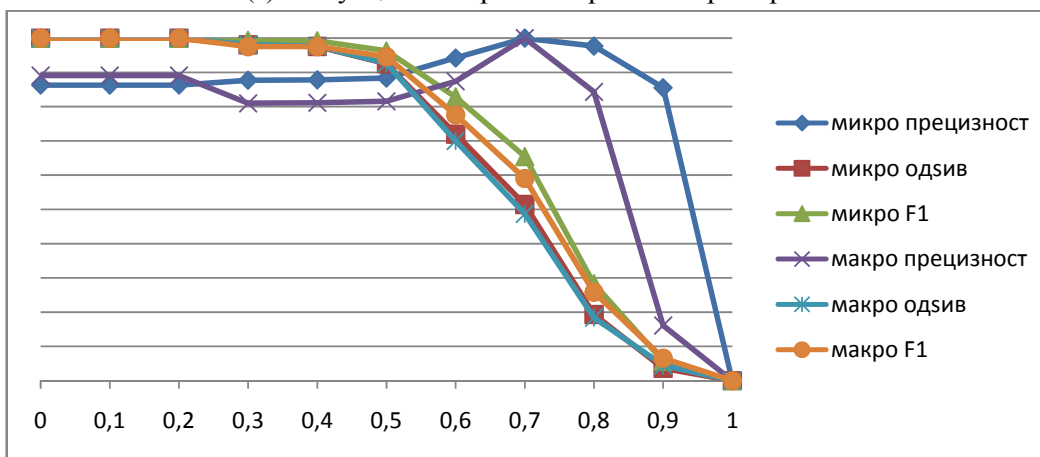


(в) евалуациски мерки базирани на рангирање и време на тестирање

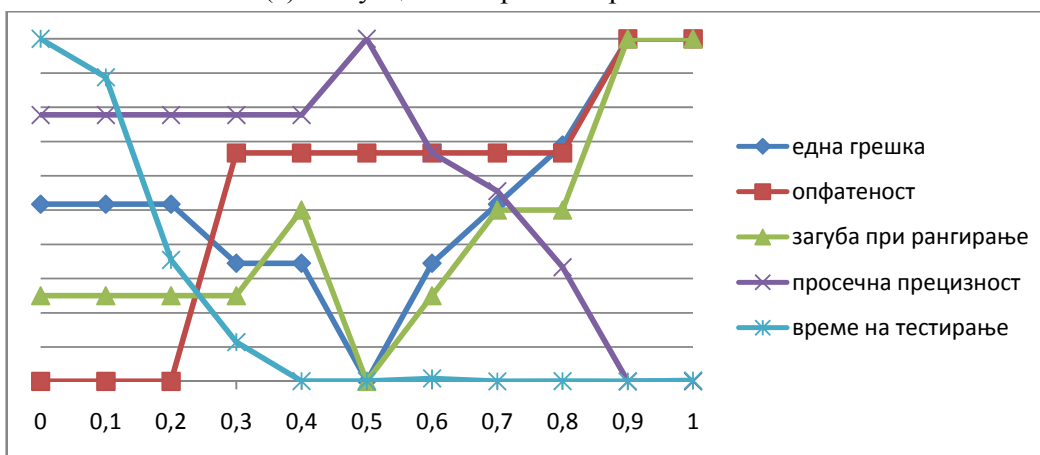
**Слика Б.9** Зависност на перформансите на дво-нивовскиот метод со гласање во однос на прагот  $t$  ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за *bibtex* податочното множество



(а) евалуациски мерки базирани на примерок

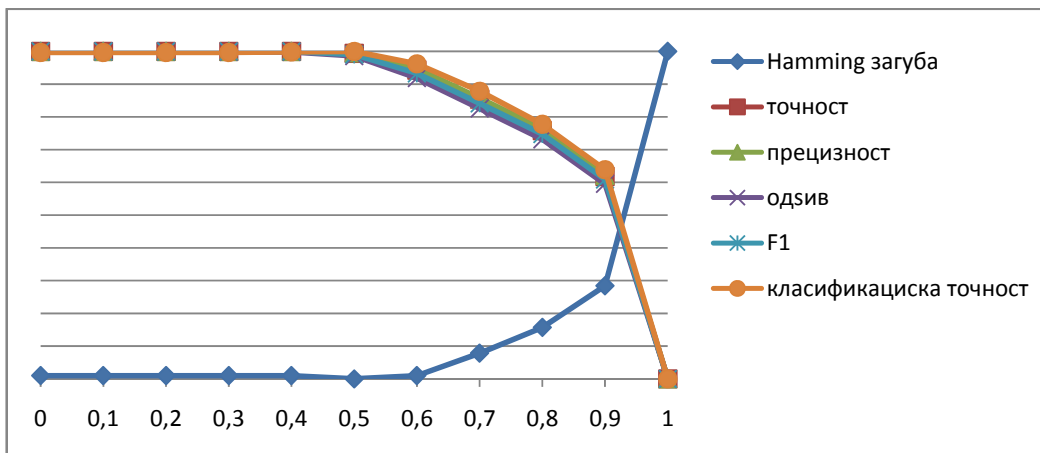


(б) евалуациски мерки базирани на ознака

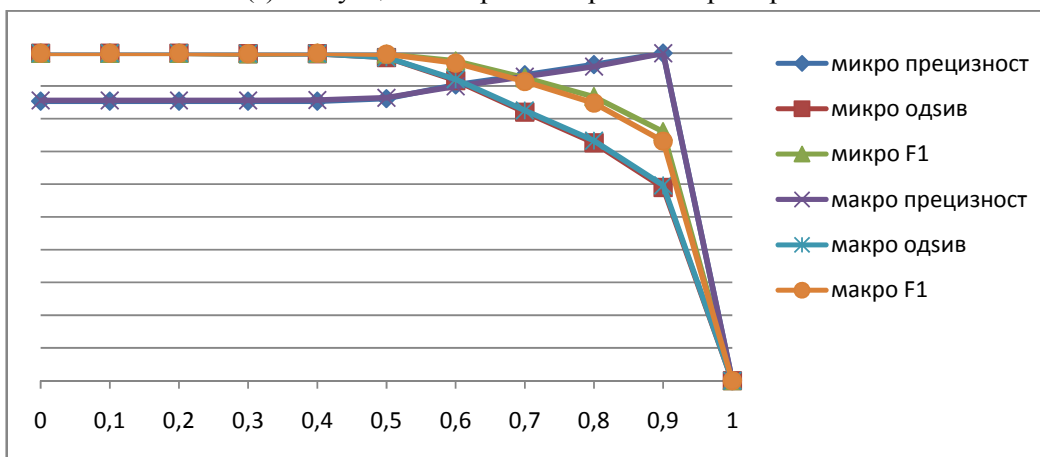


(в) евалуациски мерки базирани на рангирање и време на тестирање

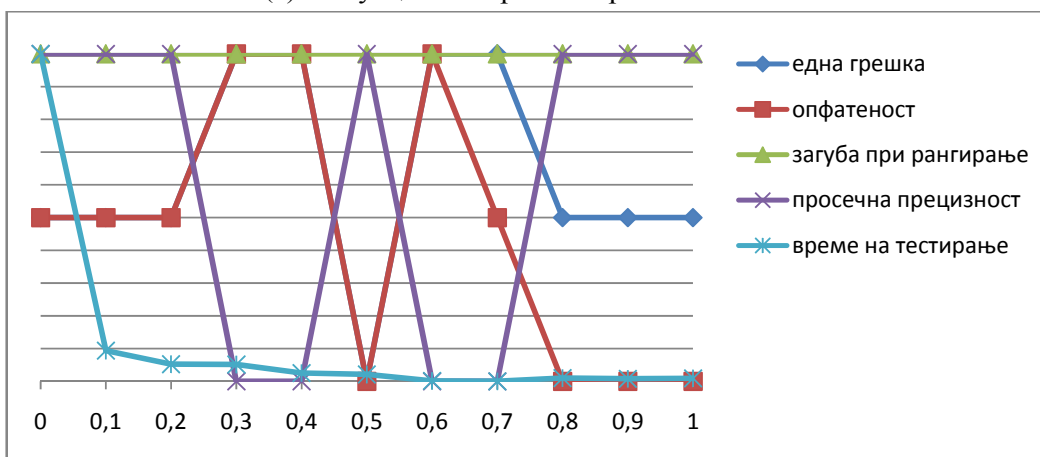
**Слика Б.10** Зависност на перформансите на дво-нивоовскиот верижан метод во однос на прагот  $t$  ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за *emotions* податочното множество



(а) евалуациски мерки базирани на примерок

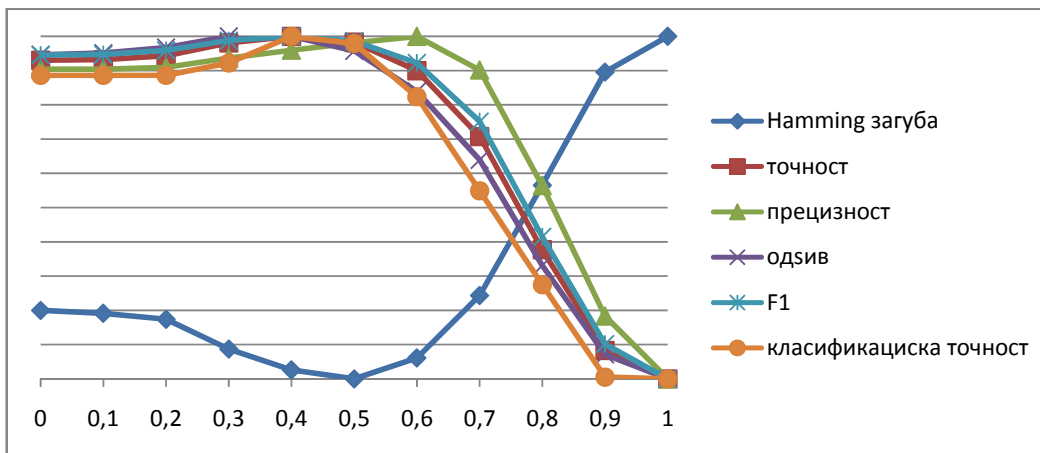


(б) евалуациски мерки базирани на ознака

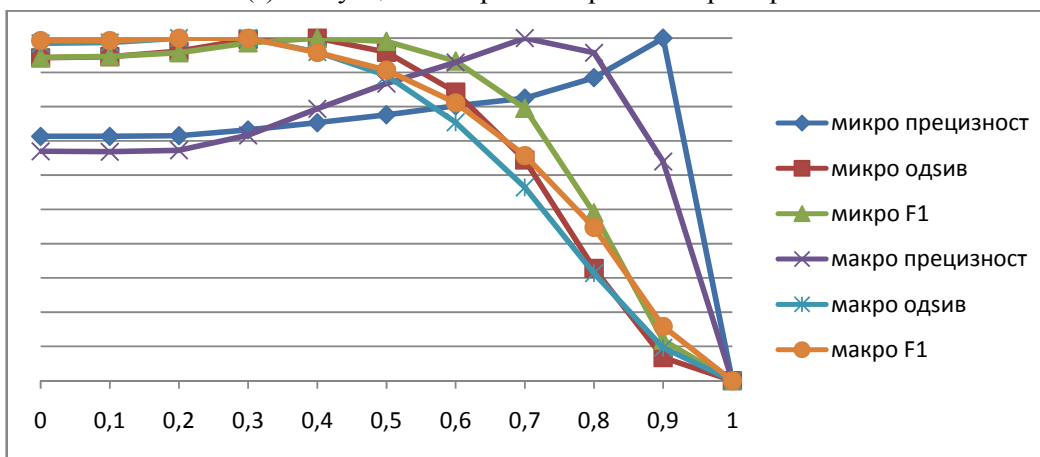


(в) евалуациски мерки базирани на рангирање и време на тестирање

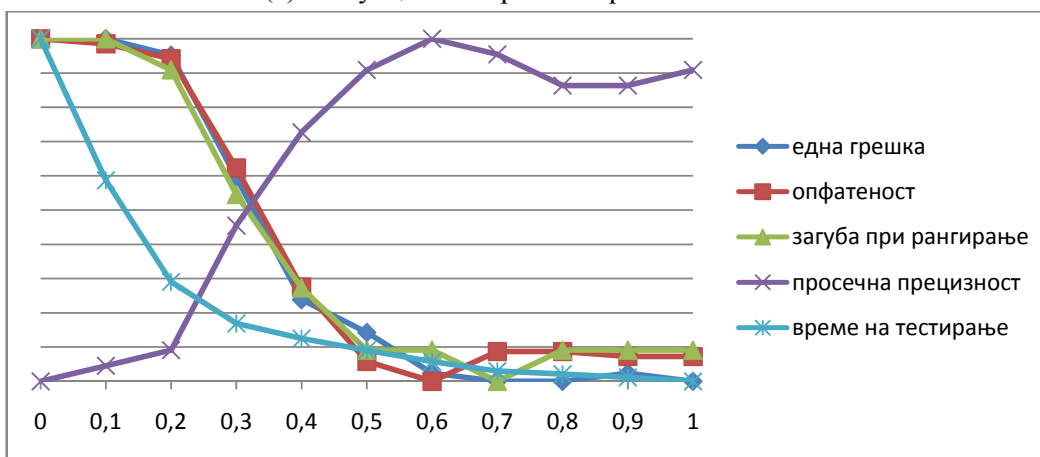
**Слика Б.11** Зависност на перформансите на дво-нивовскиот верижан метод во однос на прагот  $t$  ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за *scene* податочното множество



(а) евалуациски мерки базирани на примерок

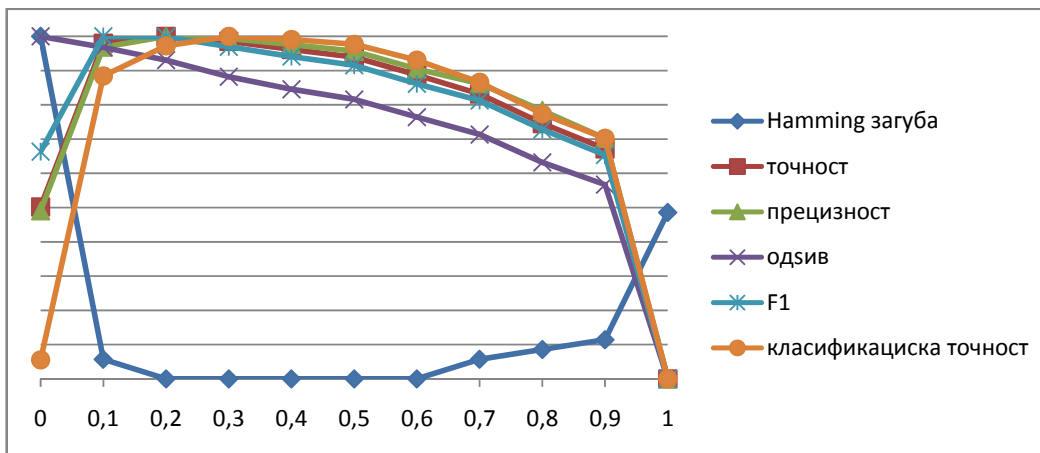


(б) евалуациски мерки базирани на ознака

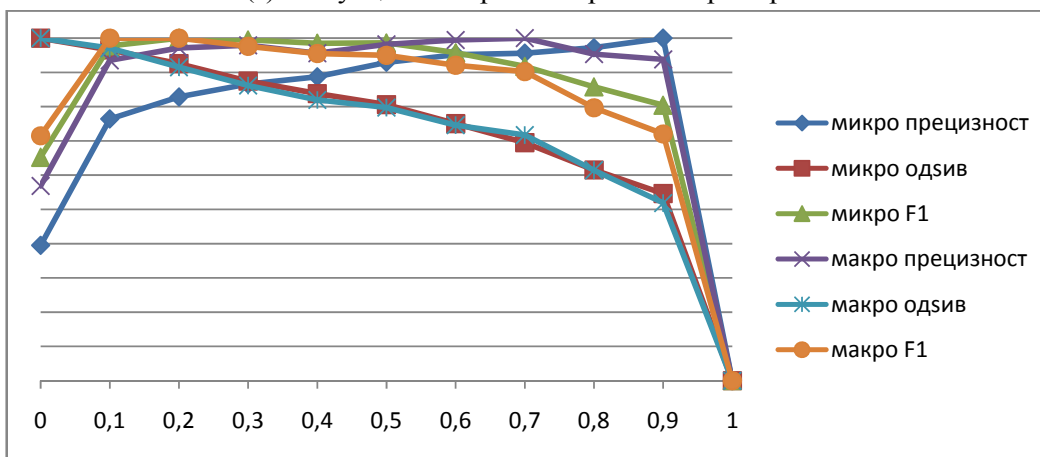


(в) евалуациски мерки базирани на рангирање и време на тестирање

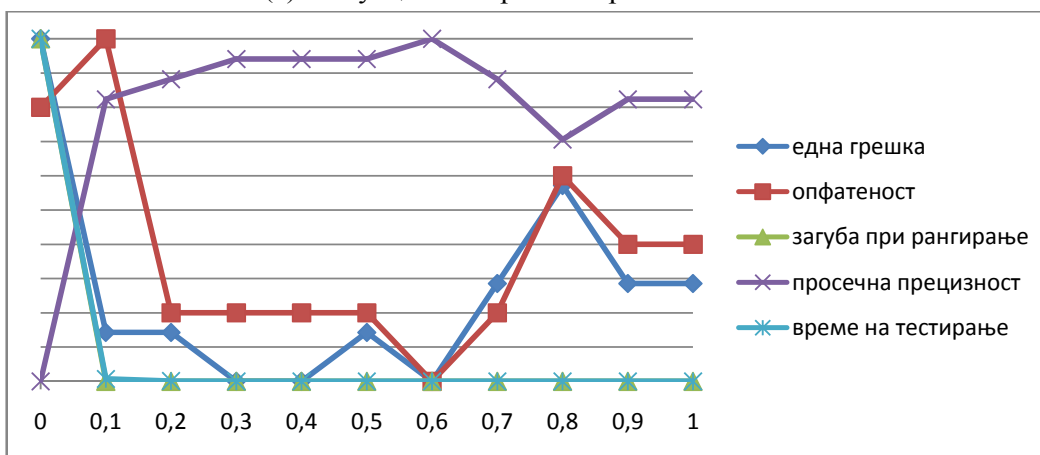
**Слика Б.12** Зависност на перформансите на дво-нивовскиот верижен метод во однос на прагот  $t$  ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за *yeast* податочното множество



(а) евалуациски мерки базирани на примерок

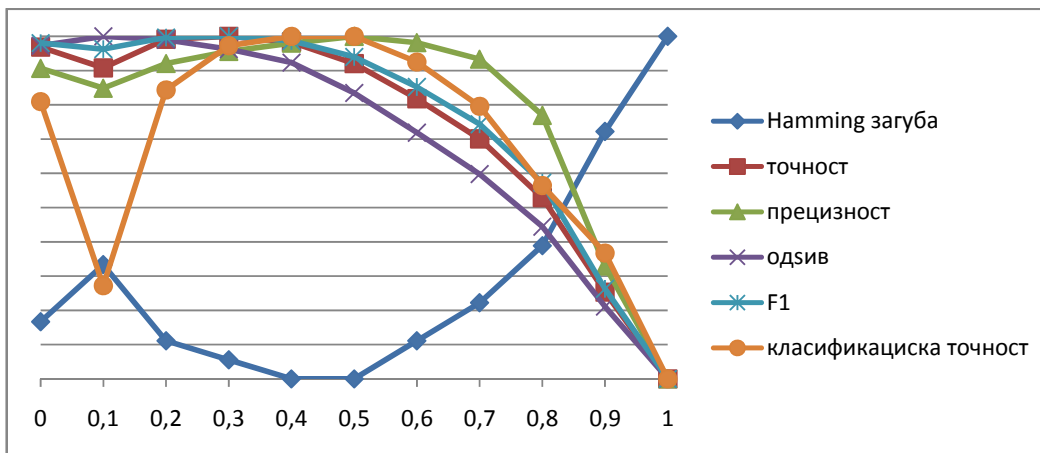


(б) евалуациски мерки базирани на ознака

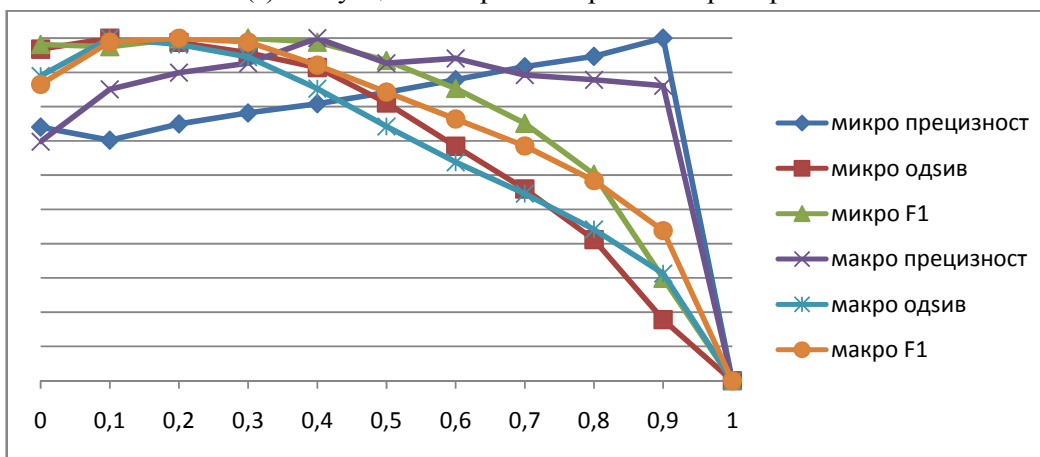


(в) евалуациски мерки базирани на рангирање и време на тестирање

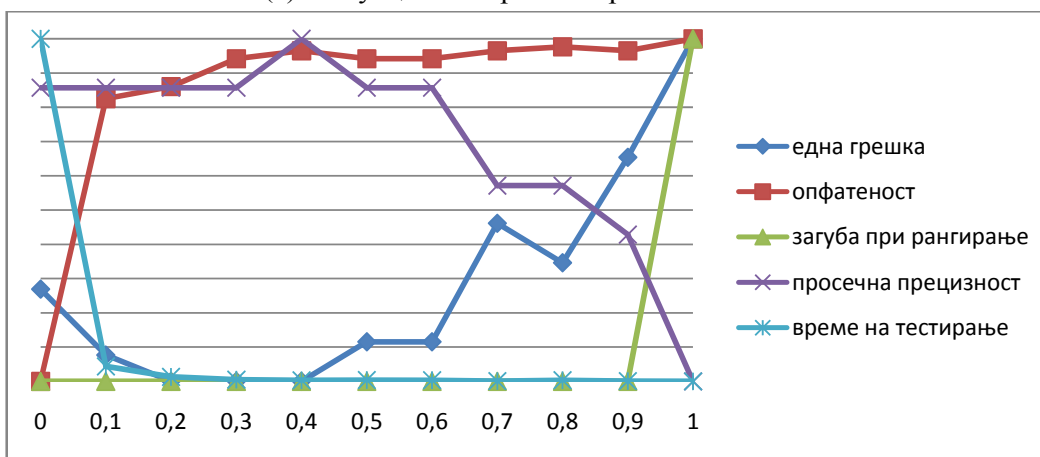
**Слика Б.13** Зависност на перформансите на дво-нивовскиот верижен метод во однос на прагот  $t$  ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за *medical* податочното множество



(а) евалуациски мерки базирани на примерок

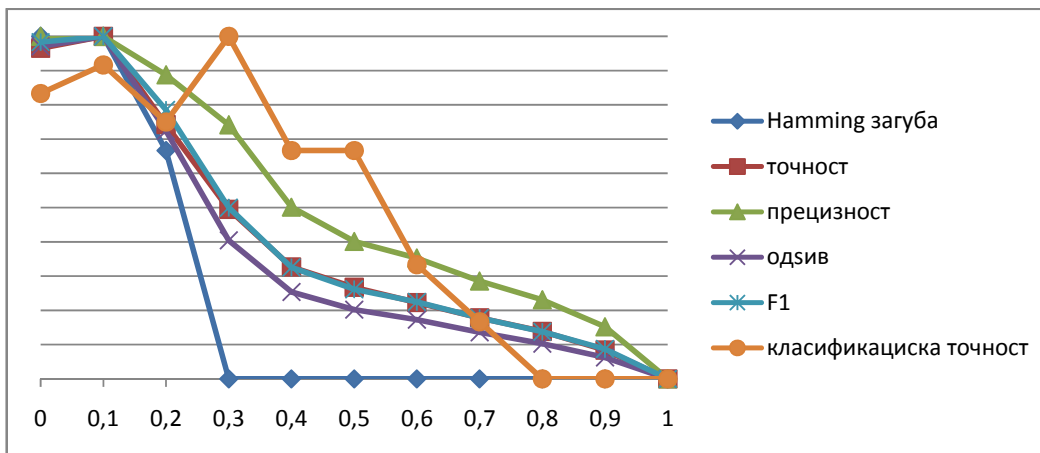


(б) евалуациски мерки базирани на ознака

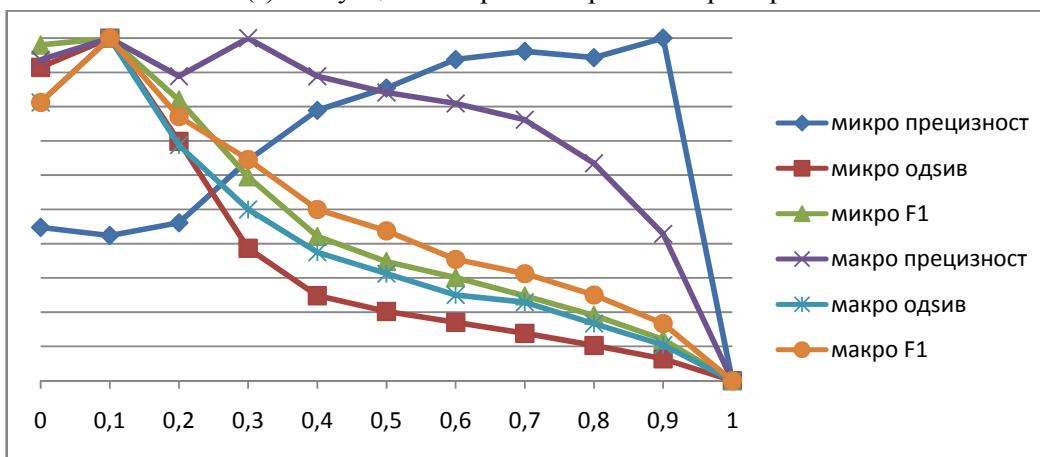


(в) евалуациски мерки базирани на рангирање и време на тестирање

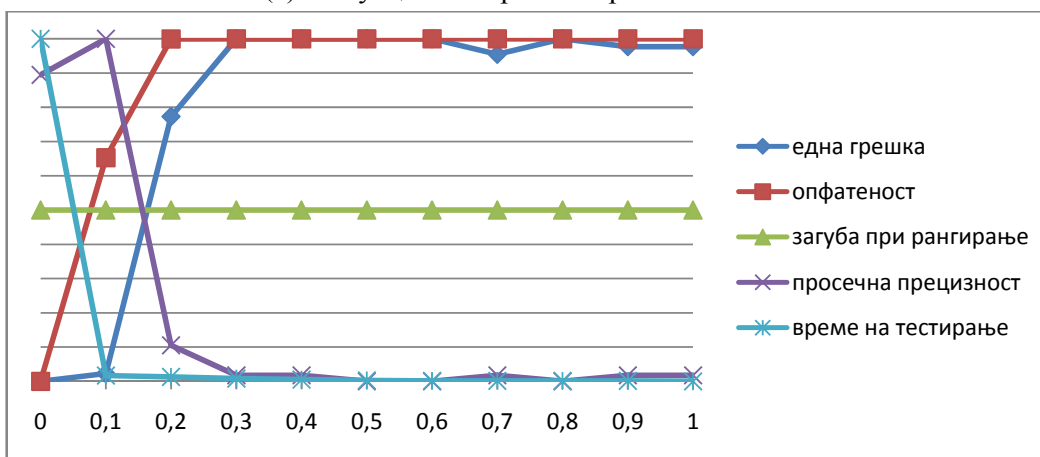
**Слика Б.14** Зависност на перформансите на дво-нивовскиот верижан метод во однос на прагот  $t$  ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за *enron* податочното множество



(а) евалуациски мерки базирани на примерок

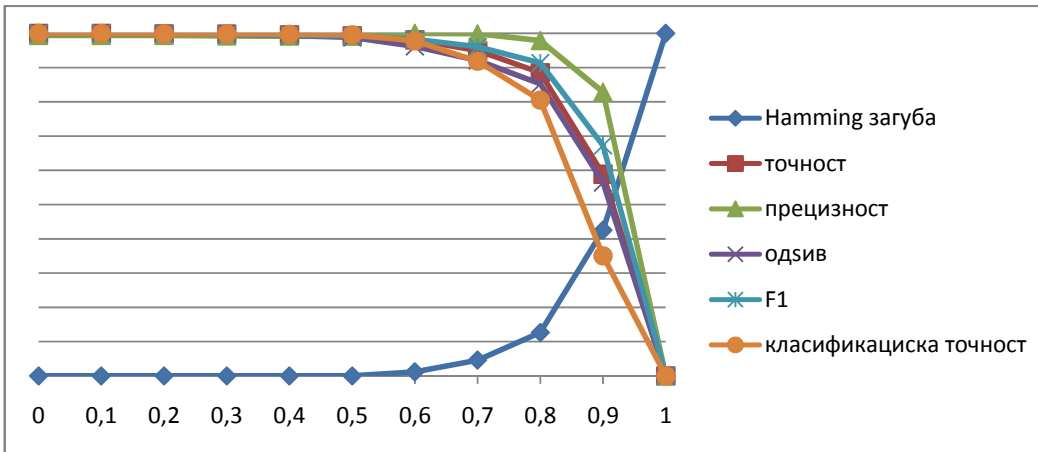


(б) евалуациски мерки базирани на ознака

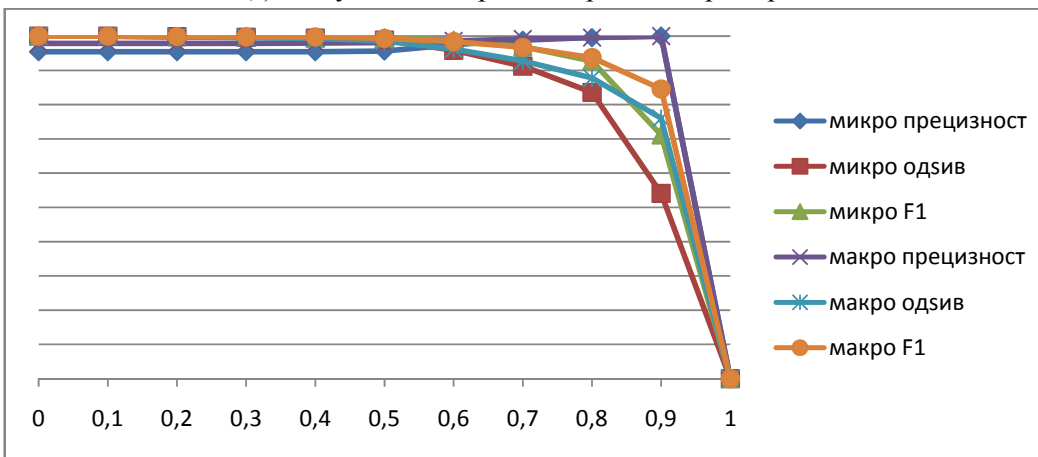


(в) евалуациски мерки базирани на рангирање и време на тестирање

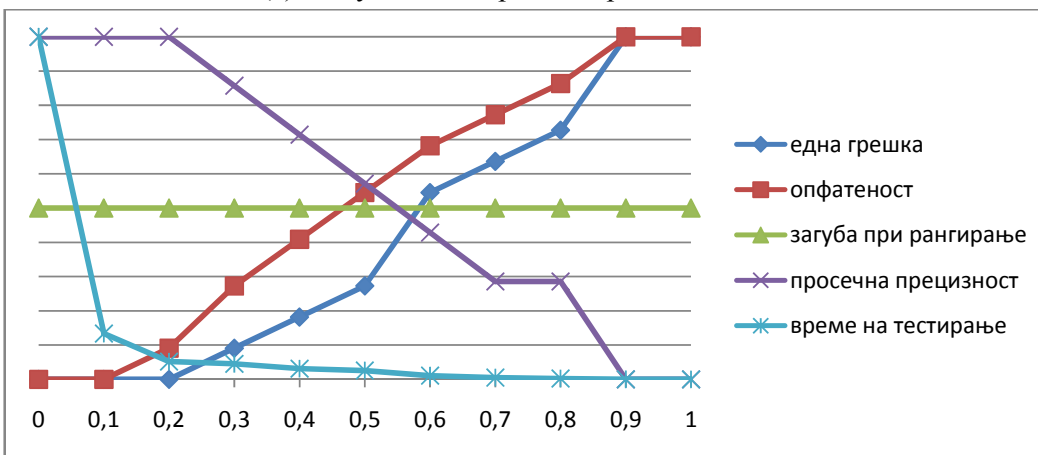
**Слика Б.15** Зависност на перформансите на дво-нивовскиот верижан метод во однос на прагот  $t$  ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за *corel5k* податочното множество



(а) евалуациски мерки базирани на примерок

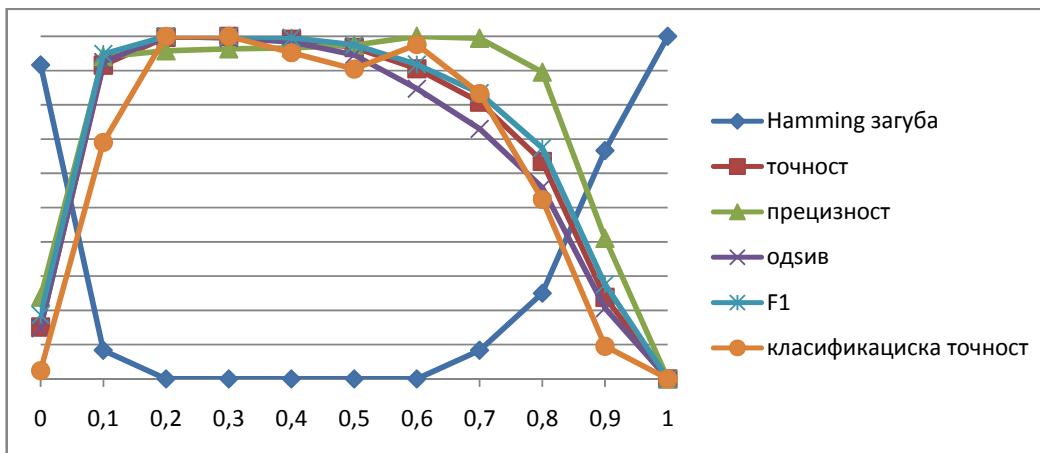


(б) евалуациски мерки базирани на ознака

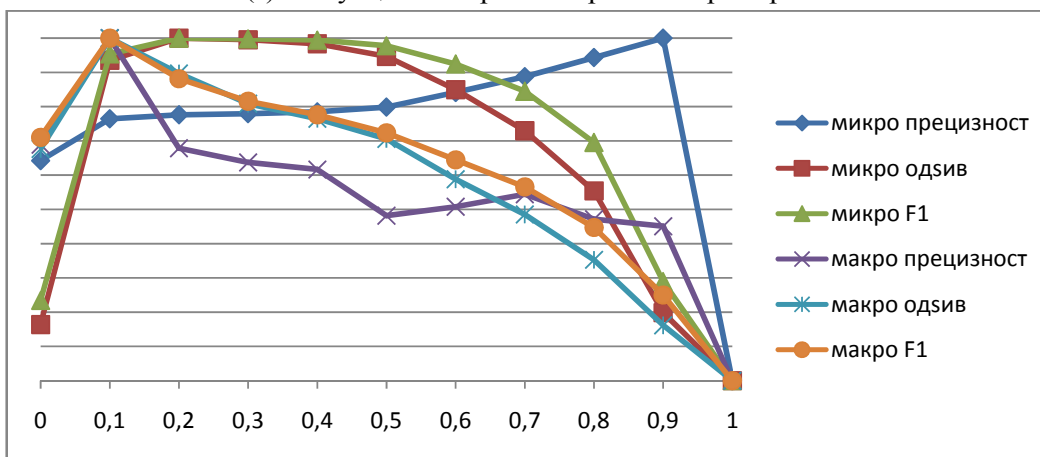


(в) евалуациски мерки базирани на рангирање и време на тестирање

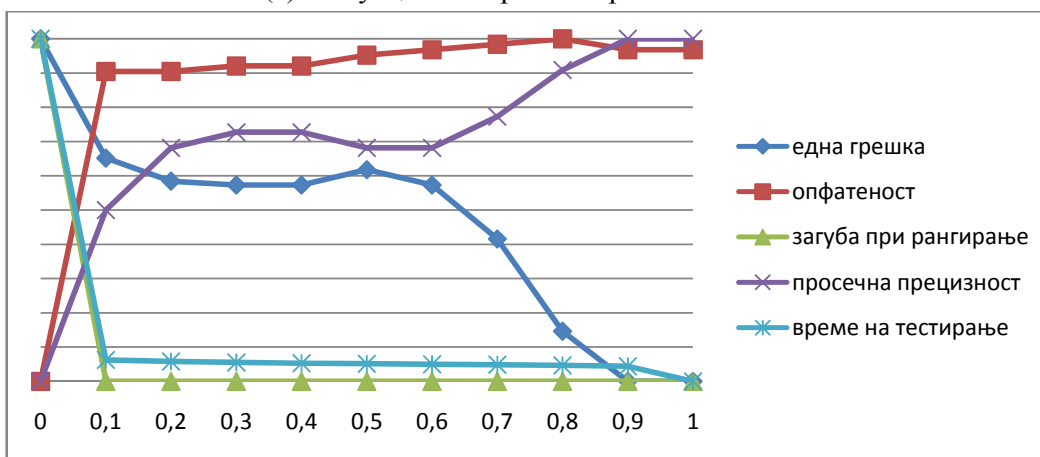
**Слика Б.16** Зависност на перформансите на дво-нивоовскиот верижан метод во однос на прагот  $t$  ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за *tmc2007* податочното множество



(а) евалуациски мерки базирани на примерок

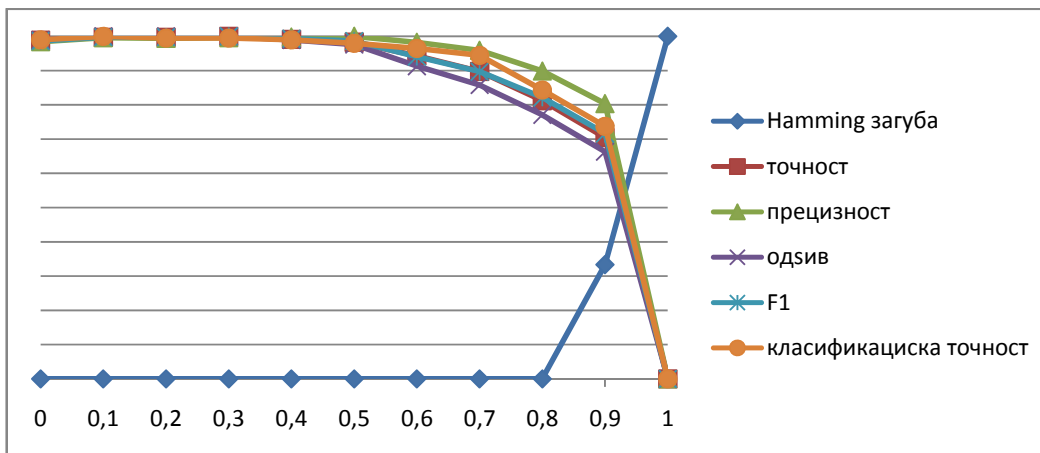


(б) евалуациски мерки базирани на ознака

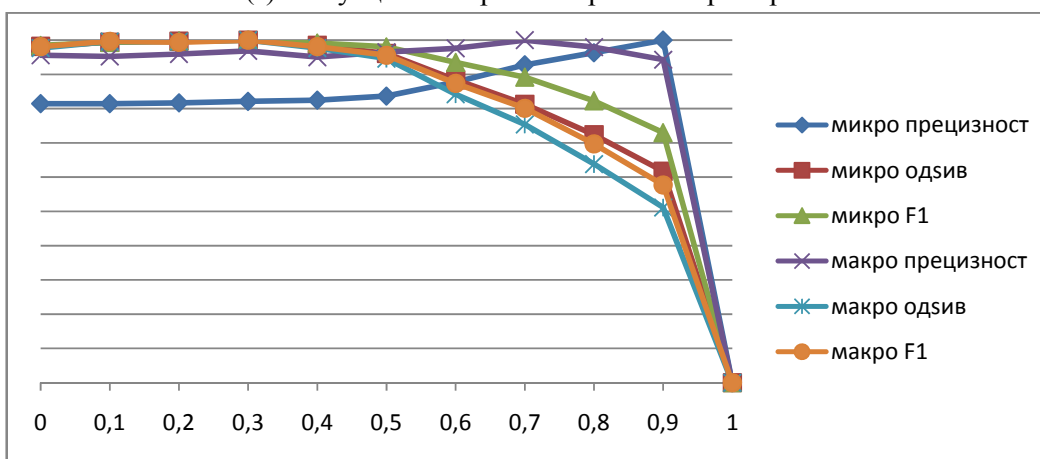


(в) евалуациски мерки базирани на рангирање и време на тестирање

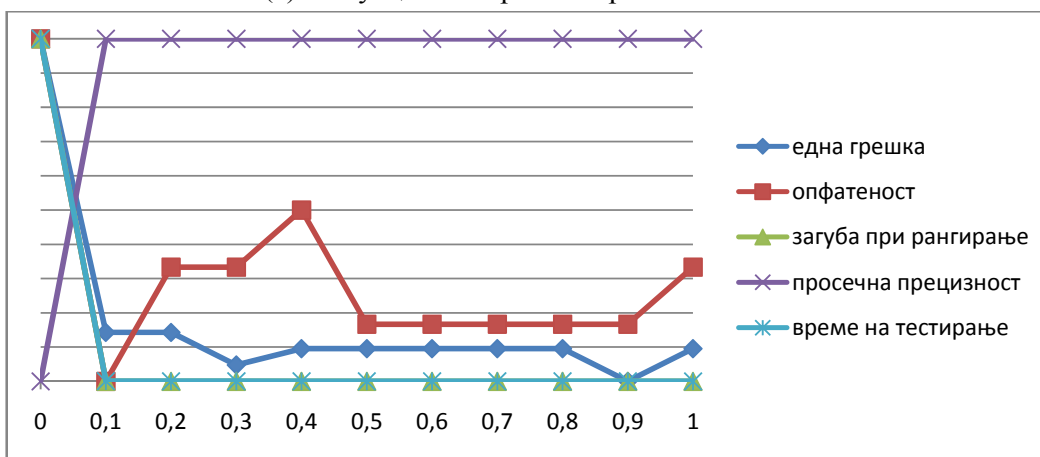
**Слика Б.17** Зависност на перформансите на дво-нивовскиот верижан метод во однос на прагот  $t$  ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за *mediamill* податочното множество



(а) евалуациски мерки базирани на примерок

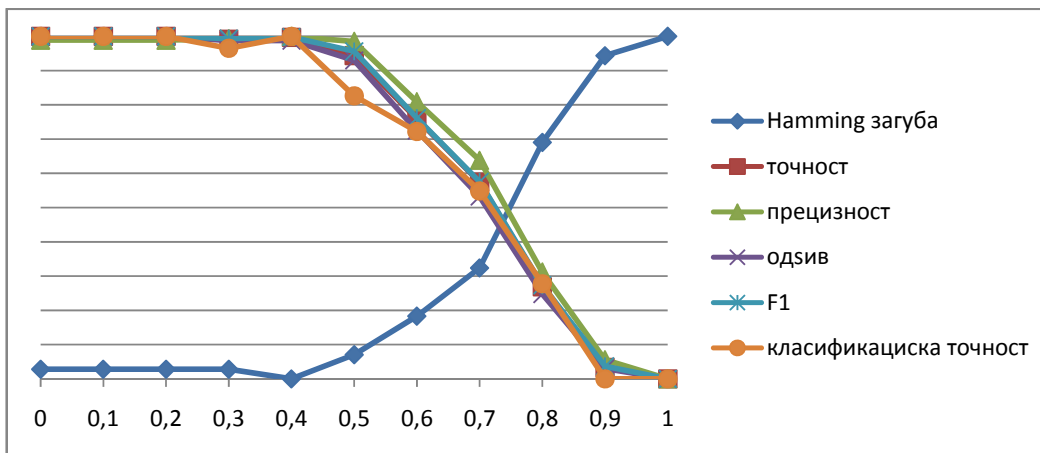


(б) евалуациски мерки базирани на ознака

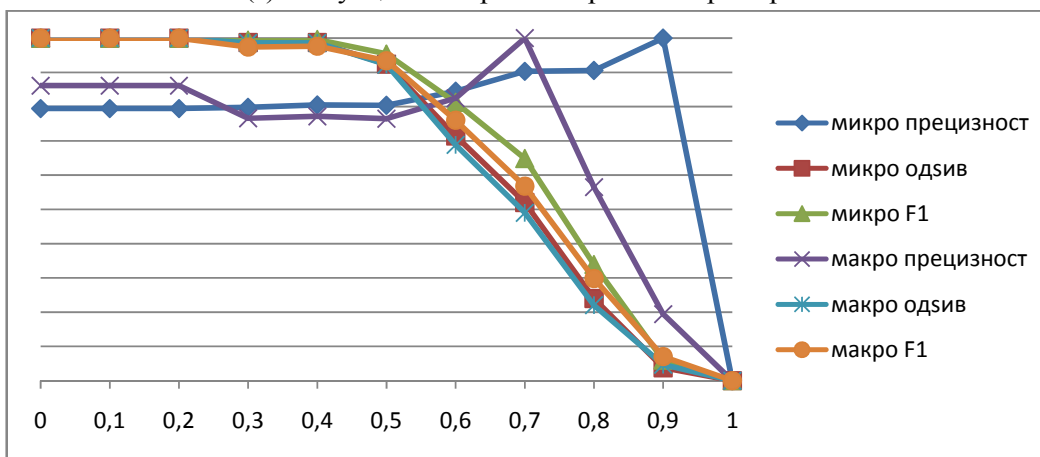


(в) евалуациски мерки базирани на рангирање и време на тестирање

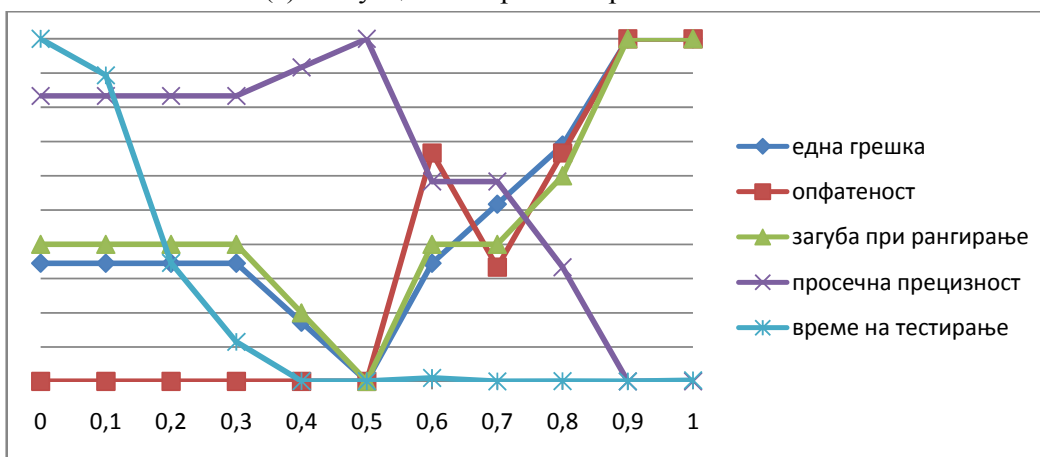
**Слика Б.18** Зависност на перформансите на дво-нивовскиот верижан метод во однос на прагот  $t$  ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за *bibtex* податочното множество



(а) евалуациски мерки базирани на примерок

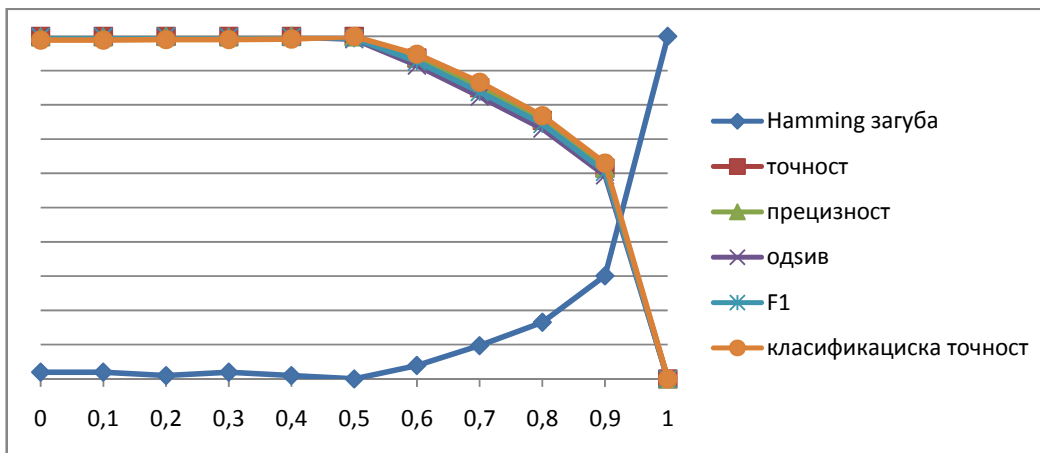


(б) евалуациски мерки базирани на ознака

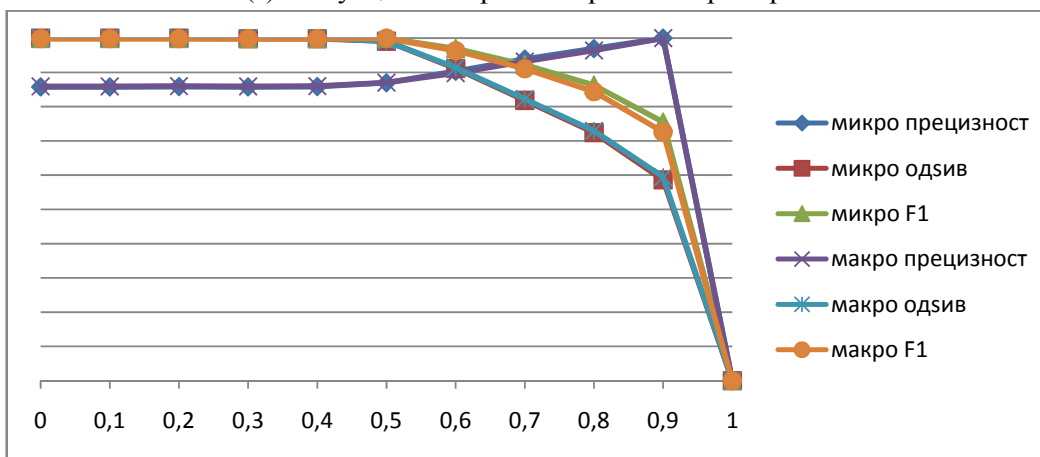


(в) евалуациски мерки базирани на рангирање и време на тестирање

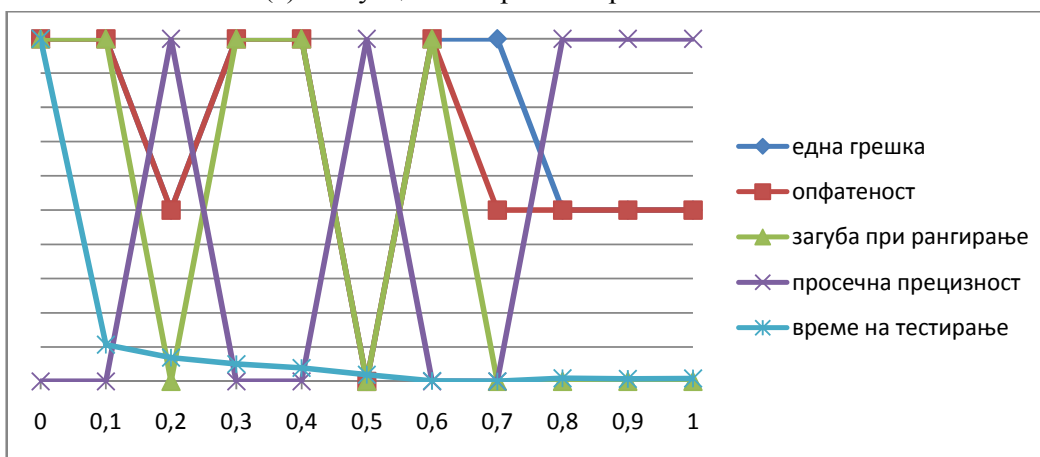
**Слика Б.19** Зависност на перформансите на дво-нивовскиот скратен верижен метод во однос на прагот  $t$  ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за *emotions* податочното множество



(а) евалуациски мерки базирани на примерок

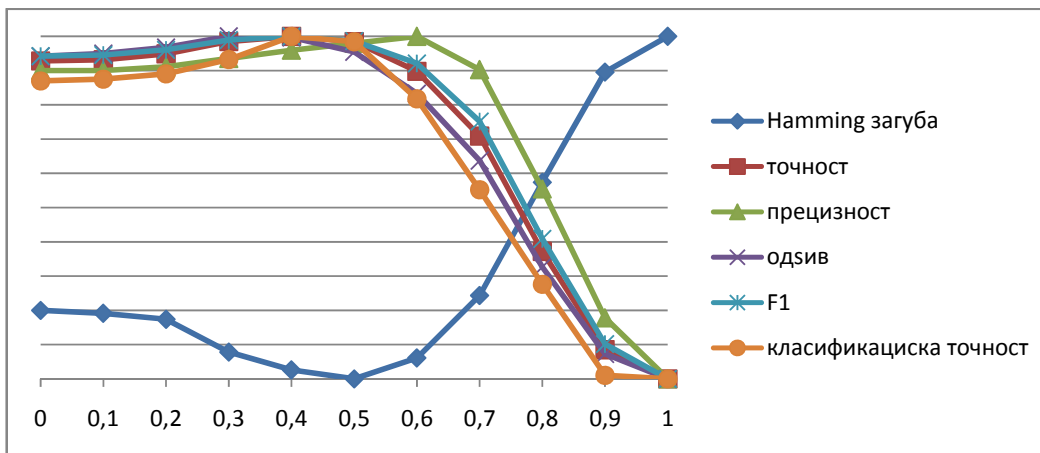


(б) евалуациски мерки базирани на ознака

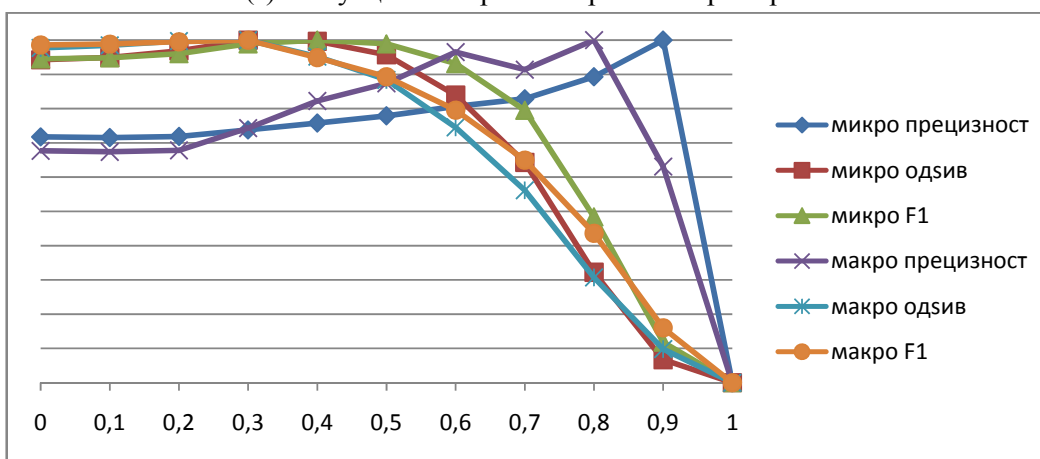


(в) евалуациски мерки базирани на рангирање и време на тестирање

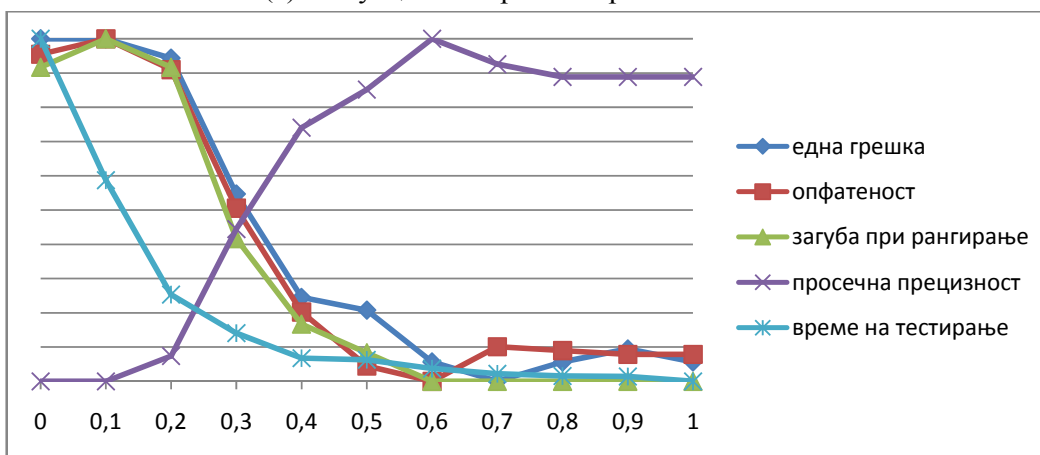
**Слика Б.20** Зависност на перформансите на дво-нивовскиот скратен верижен метод во однос на прагот  $t$  ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за *scene* податочното множество



(а) евалуациски мерки базирани на примерок

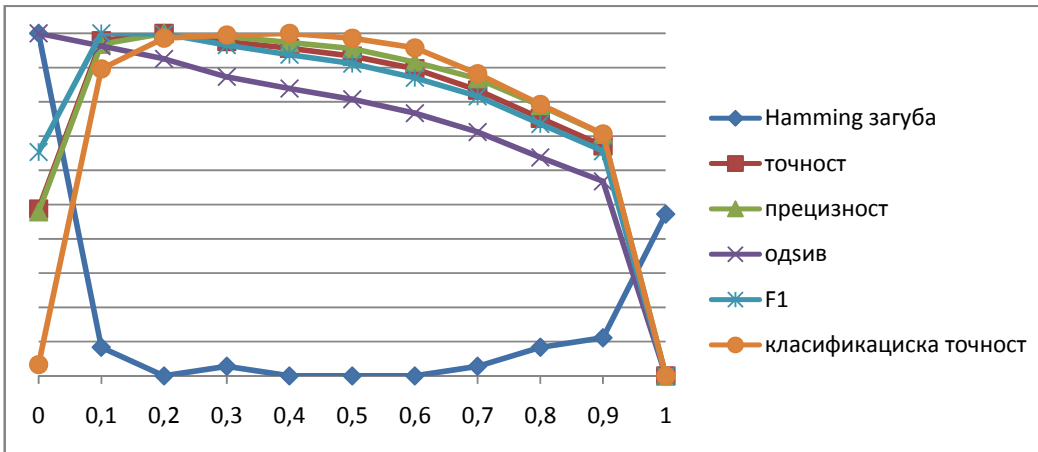


(б) евалуациски мерки базирани на ознака

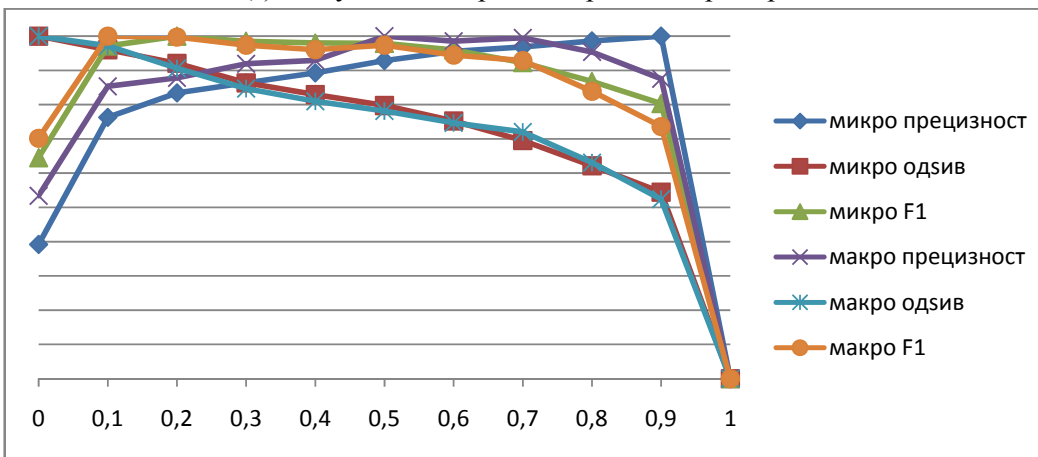


(в) евалуациски мерки базирани на рангирање и време на тестирање

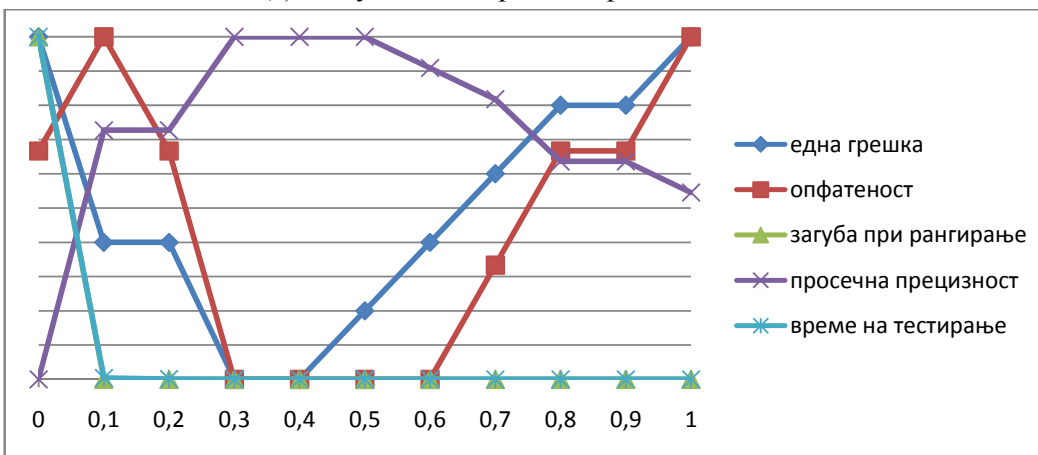
**Слика Б.21** Зависност на перформансите на дво-нивовскиот скратен верижен метод во однос на прагот  $t$  ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за *yeast* податочното множество



(а) евалуациски мерки базирани на примерок

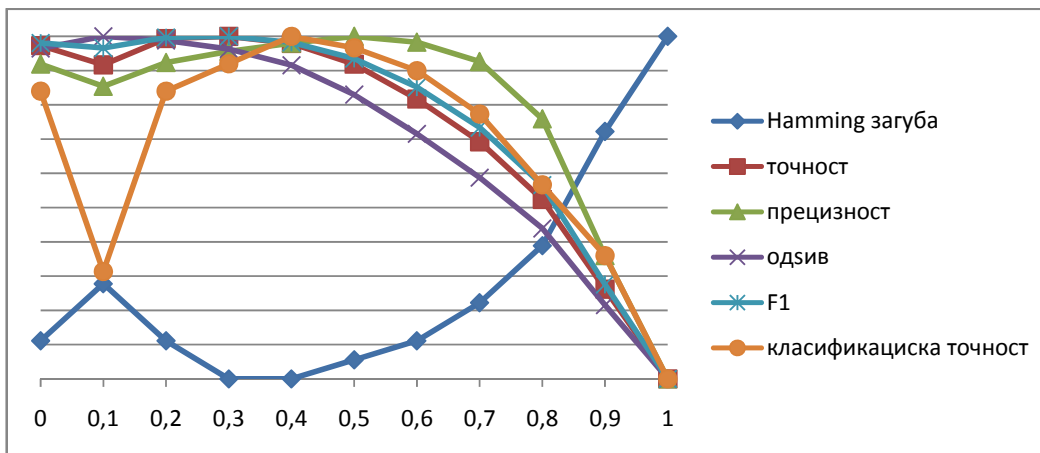


(б) евалуациски мерки базирани на ознака

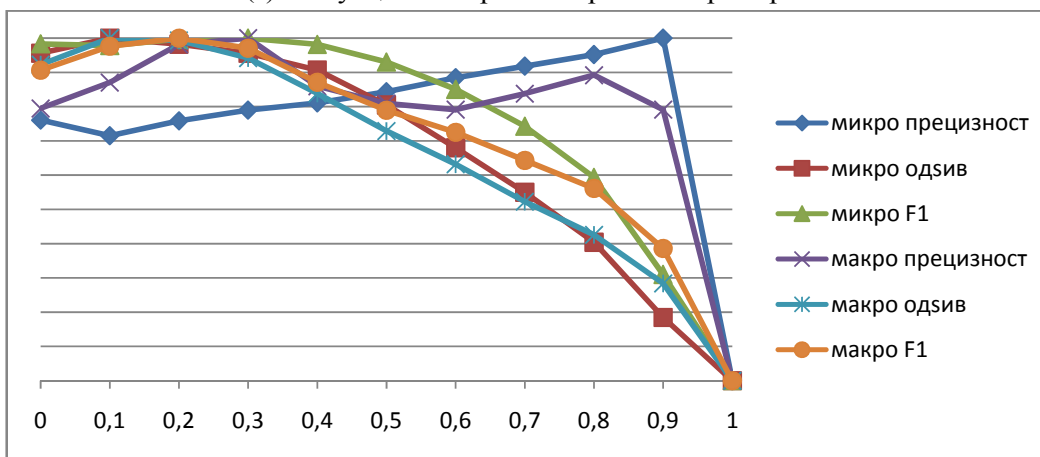


(в) евалуациски мерки базирани на рангирање и време на тестирање

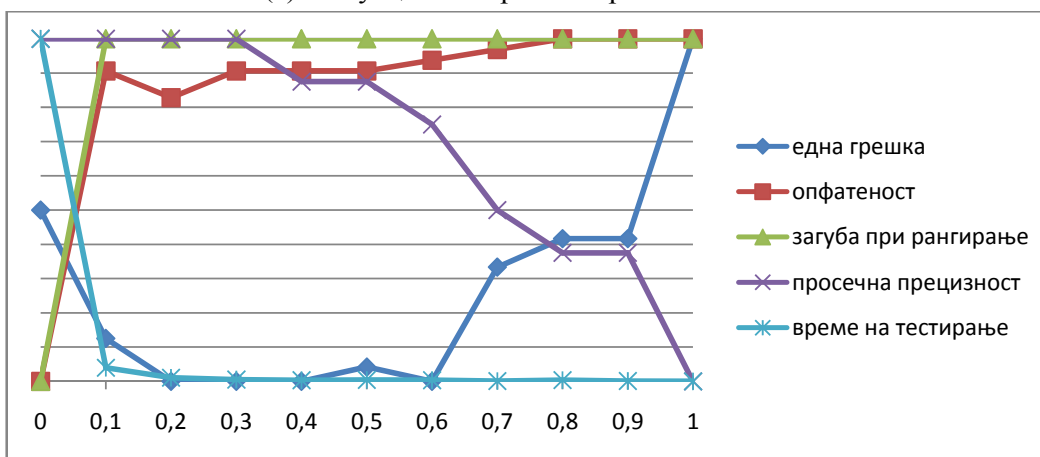
**Слика Б.22** Зависност на перформансите на дво-нивовскиот скратен верижен метод во однос на прагот  $t$  ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за *medical* податочното множество



(а) евалуациски мерки базирани на примерок

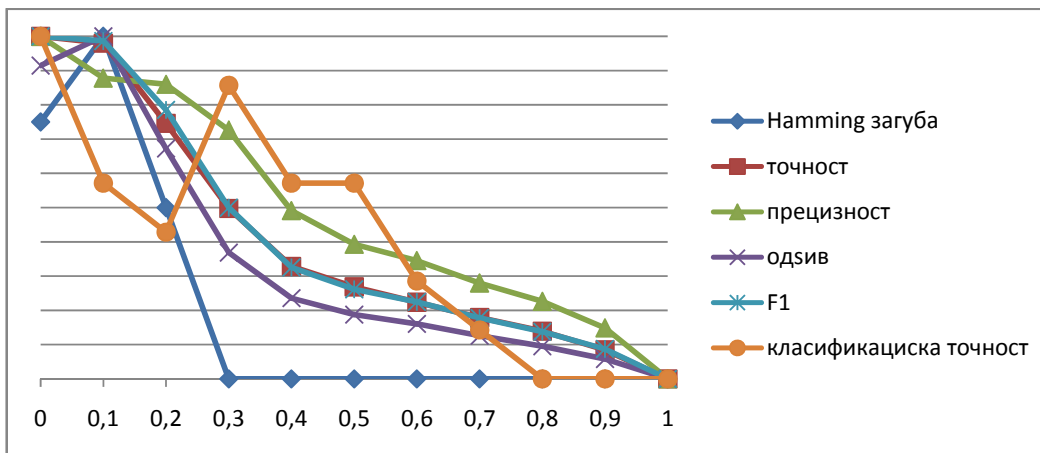


(б) евалуациски мерки базирани на ознака

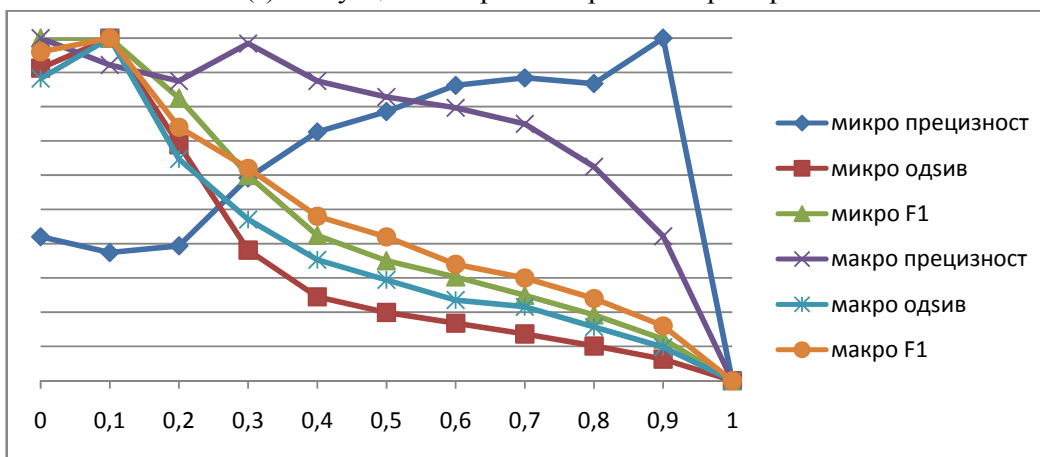


(в) евалуациски мерки базирани на рангирање и време на тестирање

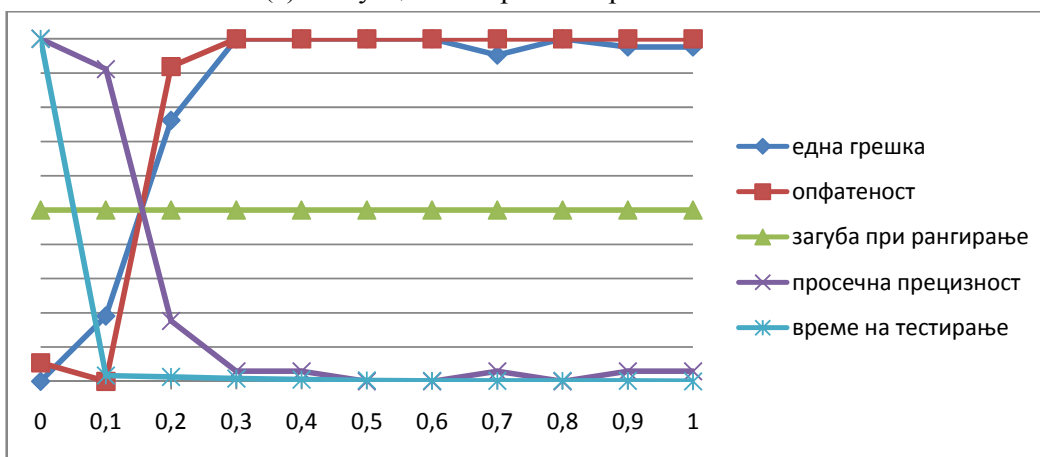
**Слика Б.23** Зависност на перформансите на дво-нивовскиот скратен верижен метод во однос на прагот  $t$  ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за *enron* податочното множество



(а) евалуациски мерки базирани на примерок

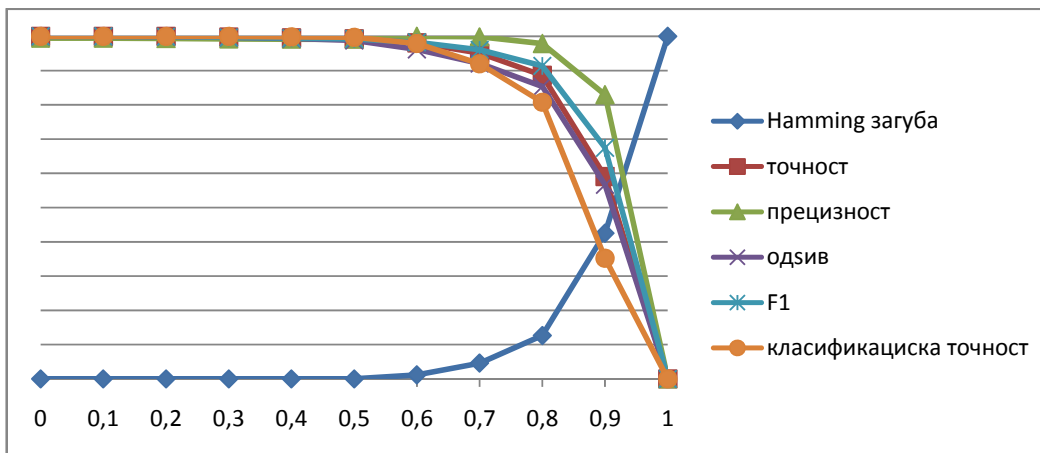


(б) евалуациски мерки базирани на ознака

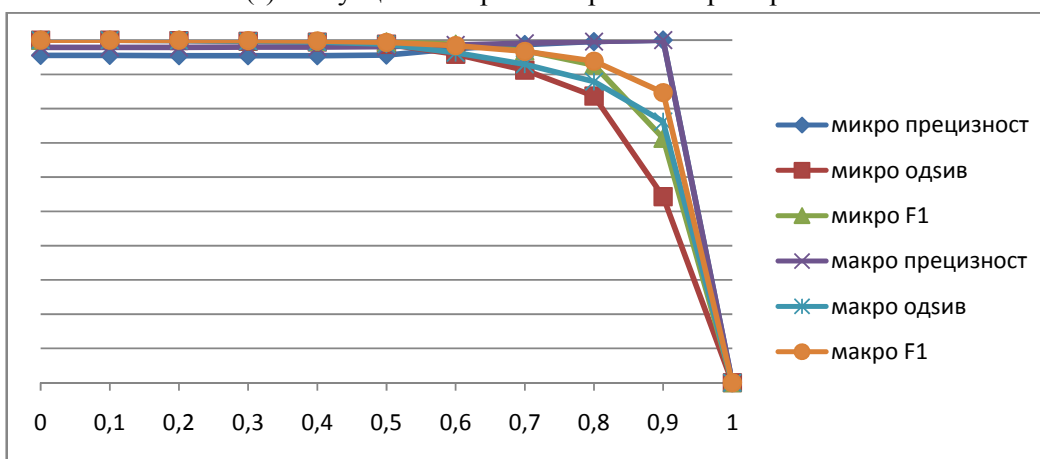


(в) евалуациски мерки базирани на рангирање и време на тестирање

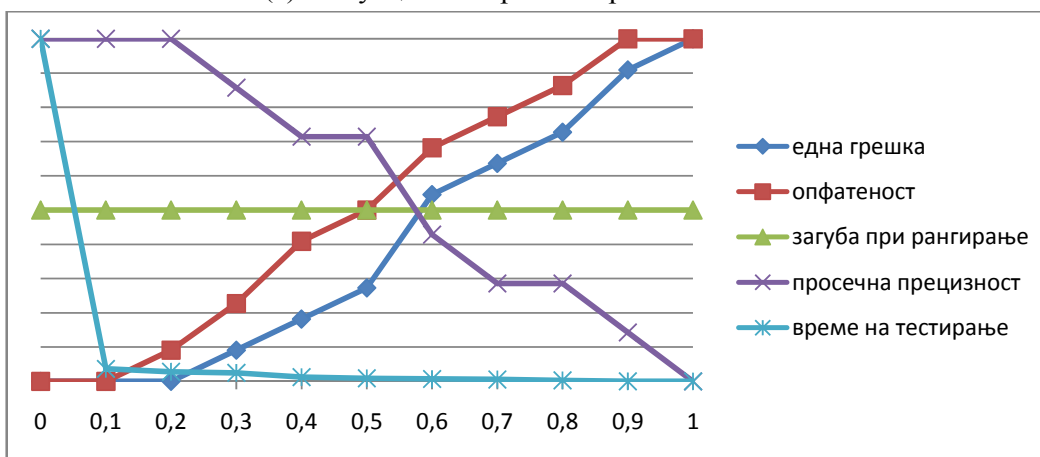
**Слика Б.24** Зависност на перформансите на дво-нивовскиот скратен верижен метод во однос на прагот  $t$  ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за *corel5k* податочното множество



(а) евалуациски мерки базирани на примерок

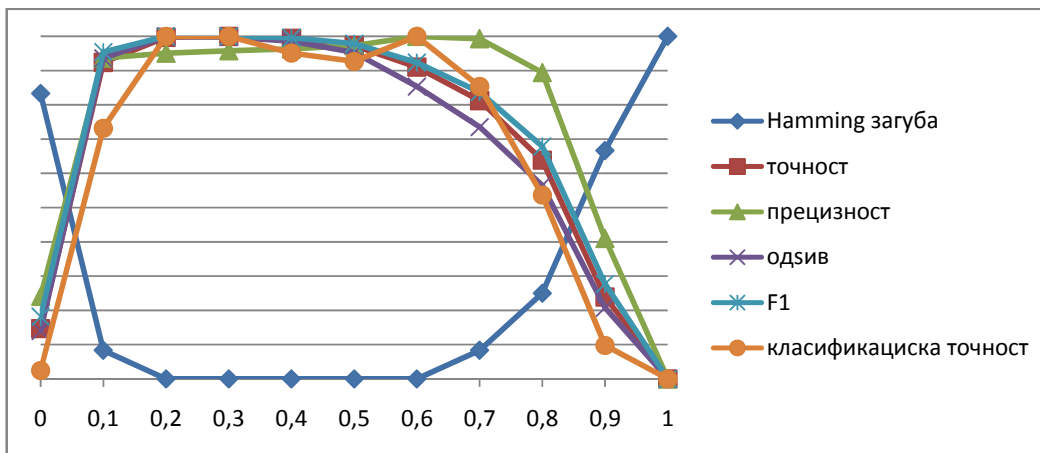


(б) евалуациски мерки базирани на ознака

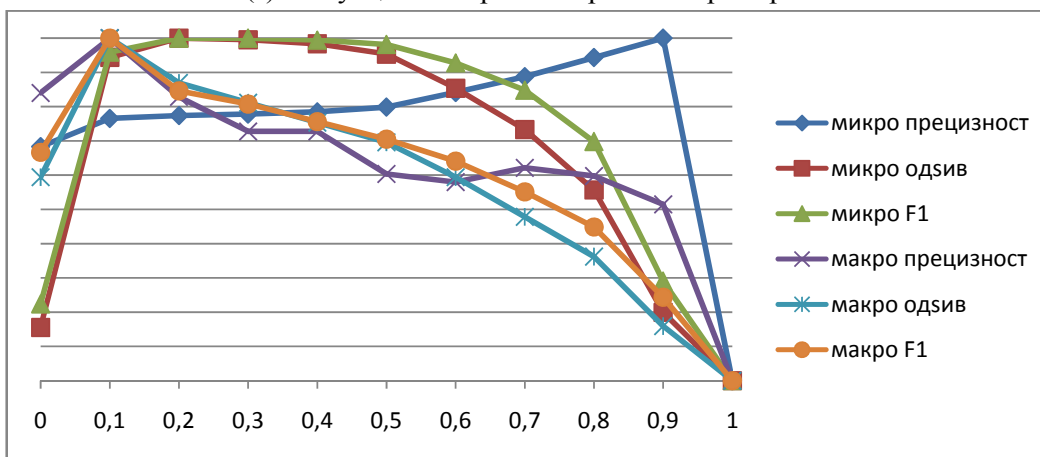


(в) евалуациски мерки базирани на рангирање и време на тестирање

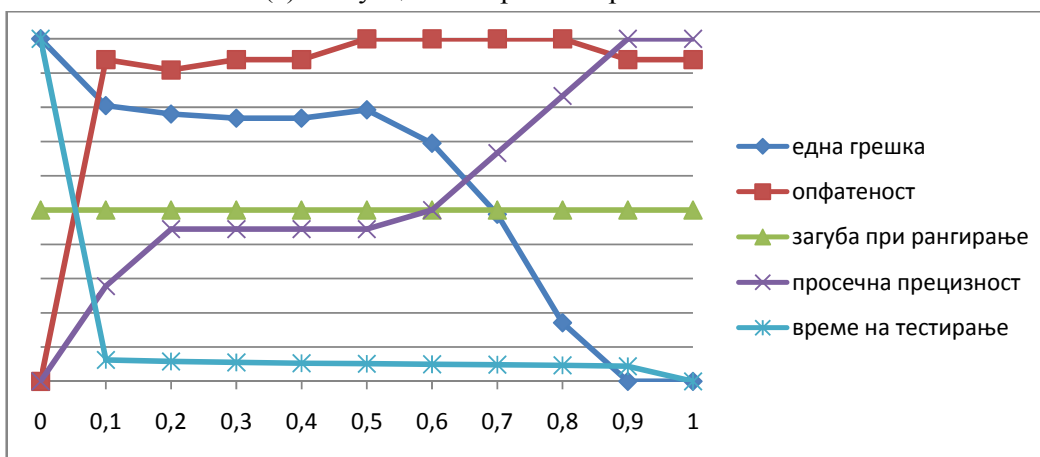
**Слика Б.25** Зависност на перформансите на дво-нивовскиот скратен верижан метод во однос на прагот  $t$  ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за *tmc2007* податочното множество



(а) евалуациски мерки базирани на примерок

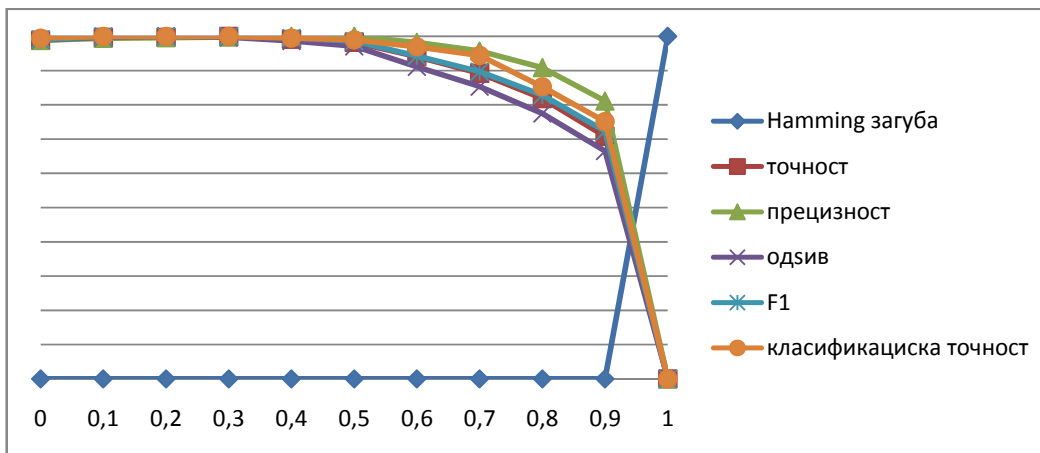


(б) евалуациски мерки базирани на ознака

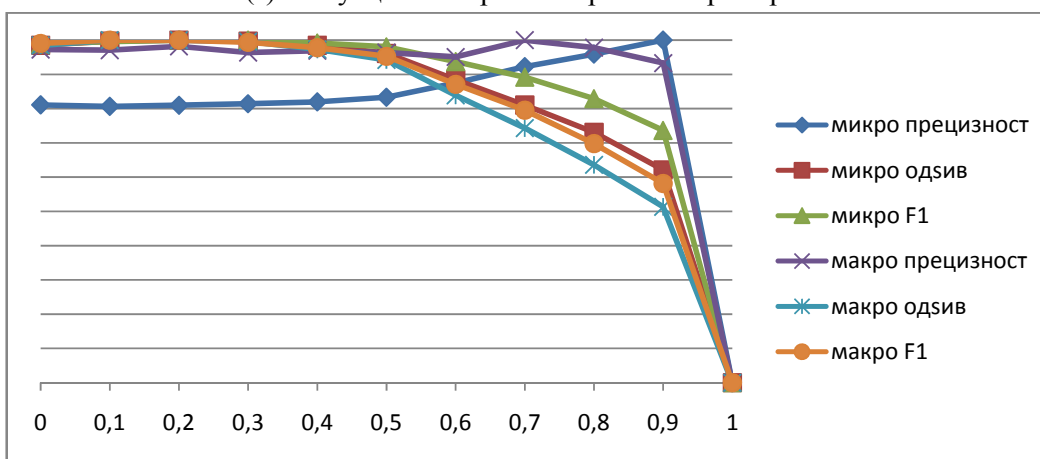


(в) евалуациски мерки базирани на рангирање и време на тестирање

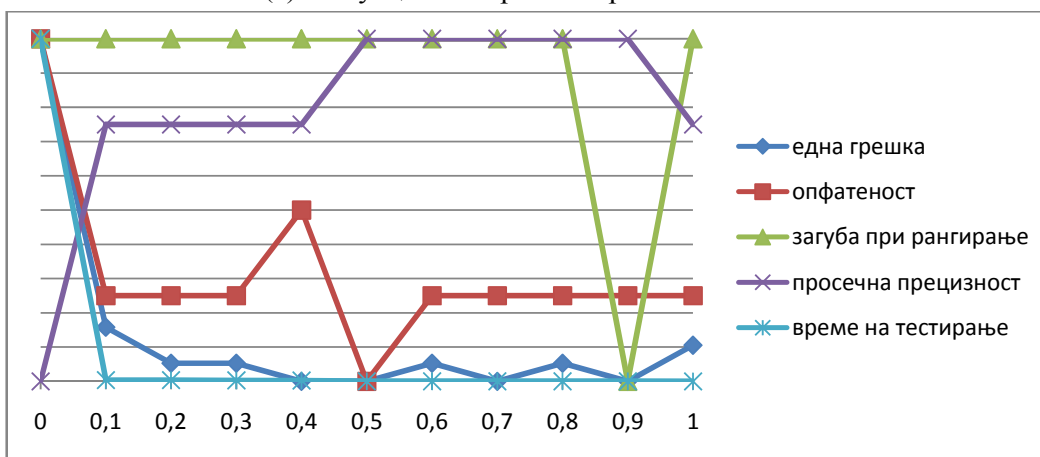
**Слика Б.26** Зависност на перформансите на дво-нивовскиот скратен верижен метод во однос на прагот  $t$  ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за *mediamill* податочното множество



(а) евалуациски мерки базирани на примерок



(б) евалуациски мерки базирани на ознака



(в) евалуациски мерки базирани на рангирање и време на тестирање

**Слика Б.27** Зависност на перформансите на дво-нивовскиот скратен верижен метод во однос на прагот  $t$  ( $t = 0, t = 0.1, t = 0.2, \dots, t = 1.0$ ) за *bibtex* податочното множество