VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÝCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER SYSTEMS

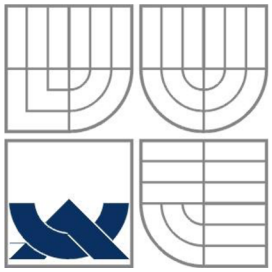
UMĚLÉ IMUNITNÍ SYSTÉMY PRO DETEKCI SPAMŮ

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

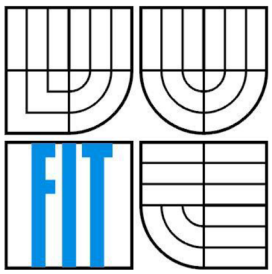
AUTOR PRÁCE
AUTHOR

Bc. Michal Hohn

BRNO 2011



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÝCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER SYSTEMS

UMĚLÉ IMUNITNÍ SYSTÉMY PRO DETEKCI SPAMŮ

ARTIFICIAL IMMUNE SYSTEMS FOR SPAM DETECTION

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE — Bc. Michal Hohn
AUTHOR

VEDOUCÍ PRÁCE — doc. Ing. Josef Schwarz, CSc.
SUPERVISOR

BRNO 2011

Zadání diplomové práce/10478/2010/xhohnm00

Vysoké učení technické v Brně - Fakulta informačních technologií

Ústav počítačových systémů Akademický rok 2010/2011

Zadání diplomové práce

Řešitel: **Hohn Michal, Bc.**
Obor: Bezpečnost informačních technologií
Téma: **Umělé imunitní systémy pro detekci spamů**
Artificial Immune Systems for Spam Detection
Kategorie: Umělá inteligence

Pokyny:

1. Zpracujte rešerši používaných technik pro detekci nevyžádané elektronické pošty- spamů.
2. Seznamte se s problematikou umělých imunitních systémů (UIS).
3. Navrhněte aplikaci UIS pro filtraci spamů. Zvažte možnost využití vhodných heuristik nebo dalších technik umělé inteligence v rámci UIS aplikace.
4. Navržený systém implementujte na vybrané softwarové platformě.
5. Zhodnoťte dosažené výsledky, včetně účinnosti navrženého systému.

Literatura:

- Dle pokynů vedoucího.

Při obhajobě semestrální části diplomového projektu je požadováno:

- Splnění bodů 1 - 2 zadání.

Podrobné závazné pokyny pro vypracování diplomové práce naleznete na adrese http://www.fit.vutbr.cz/info/szz/

Technická zpráva diplomové práce musí obsahovat formulaci cíle, charakteristiku současného stavu, teoretická a odborná východiska řešených problémů a specifikaci etap, které byly vyřešeny v rámci ročníkového a semestrálního projektu (30 až 40% celkového rozsahu technické zprávy).

Student odevzdá v jednom výtisku technickou zprávu a v elektronické podobě zdrojový text technické zprávy, úplnou programovou dokumentaci a zdrojové texty programů. Informace v elektronické podobě budou uloženy na standardním nepřepisovatelném paměťovém médiu (CD-R, DVD-R, apod.), které bude vloženo do písemné zprávy tak, aby nemohlo dojít k jeho ztrátě při běžné manipulaci.

Vedoucí: **Schwarz Josef, doc. Ing., CSc.**, UPSY FIT VUT
Datum zadání: 20. září 2010
Datum odevzdání: 25. května 2011

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
Fakulta informačních technologií
Ústav počítačových systémů a sítí
612 66 Brno, Božetěchova 2

L.S.

doc. Ing. Zdeněk Kotásek, CSc.
vedoucí ústavu

Abstrakt

Tato práce se zabývá tvorbou hybridního systému, založeného na agregaci umělého imunitního systému s vhodnými heuristikami, aby co nejúspěšněji detekoval nevyžádanou poštu. V práci jsou popsány hlavní principy biologického a umělého imunitního systému, klasické techniky pro rozpoznávání spamu včetně několika klasifikátorů. Navržený systém je testován na známých datových korpusech a je provedeno srovnání výsledných experimentů.

Abstract

This work deals with creating a hybrid system based on the aggregation of artificial immune system with appropriate heuristics to make the most effective spam detection. This work describes the main principles of biological and artificial immune system and conventional techniques to detect spam including several classifiers. The developed system is tested using well known database corpuses and a comparison of the final experiments is made.

Klíčová slova

Elektronická pošta, email, spam, ham, umělý imunitní systém, lymfocyt, klasifikátor.

Keywords

Electronic mail, email, spam, ham, artificial immune system, lymphocytes, classifier.

Citace

HOHN MICHAL: *Umělé imunitní systémy pro detekci spamů*, diplomová práce, Brno, FIT VUT v Brně, 2011

Umělé imunitní systémy pro detekci spamů

Prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně pod vedením doc. Ing. Josefa Schwarze, CSc.
Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....
Bc. Michal Hohn
25.5.2011

Poděkování

Rád bych poděkoval vedoucímu diplomové práce doc. Ing. Josefu Schwarzovi, CSc. za cenné rady, konstruktivní připomínky a ochotu věnovat mně dostatek času při konzultacích během tvorby mé diplomové práce. Zvláště bych rád poděkoval rodině a mým blízkým za všestrannou podporu během studia.

1 Obsah

2 Úvod

V dnešní době existuje nepřeberné množství komunikačních kanálů, ať se jedná o *VoIP* či *Skype*, *Instant Messaging(ICQ, Jabber,..)*, různé sociální sítě (*Facebook, Twitter,..*) a v neposlední řadě elektronická pošta (email, resp. E-mail). Email patří mezi nejvíce používaný komunikační prostředek ve firemní sféře.

Jakákoliv varianta má své úskalí a v případě emailů je to nevyžádaná pošta, neboli spam. Existuje několik nástrojů a technik, které se dokáží s tímhle nešvarem, více či méně efektivně, vypořádat. Když si představíme, kolik času zabere zaměstnanci četba spamu, procesorový čas na filtrování, či přeposílání, dostáváme se na číslo několika desítek miliard dolarů ročně a to už je globální problém.

Umělé imunitní systémy jsou relativně novou oblastí informačních technologií. Cílem této diplomové práce je využití umělých imunitních systémů pro návrh progrmu pro filtraci spamů v režimu off-line. Cílem práce tedy není on-line filtrace elektronické pošty, tak jak je popsáno v [url-neu].

V kapitolách 3 až 6 je systematická rešerše problematiky biologického a umělého imunitního systému včetně jednotlivých technik klasifikátorů. Vlastní přínos mé práce je prezentován v kapitolách sedm až devět.

3 Biologický imunitní systém

Informace jsou čerpány z [Tschudin]

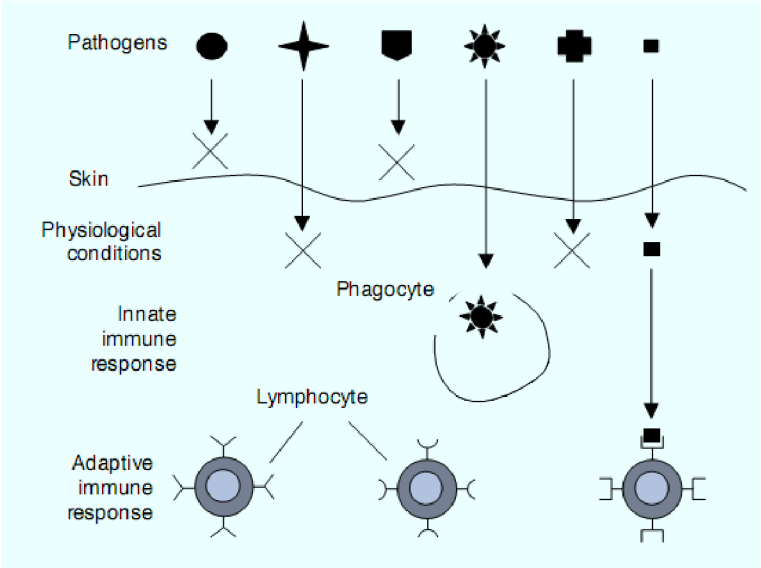
3.1 Co je to ?

Každý živočich na světě má evolucí přizpůsobený biologický imunitní systém (dále jen imunitní systém) svému prostředí. Imunitní systém je velice inteligentní multifunkční stroj na detekci nejrůznějších virů, plísní a bakterií. Tyto cizorodé látky zvané *patogeny,* jsou rozpoznány a následně eliminovány.

„Samotná detekce je velice komplikovaná, neboť se *patogeny* v průběhu existence svého druhu vyvíjejí a různě mutují. Imunitní systém musí být tedy schopen rozpoznat a eliminovat útočníky, se kterými se již dříve setkal, ale také útočníky nové, které ještě nezná.“ Cit.[url-neu].

3.2 Vrstvy imunitního systému

Imunitní system se skládá ze čtyř základních vrstvev. Každá z nich plní jinou úlohu a má jinak „předprogramovanou“ funkci. Podle houževnatosi, odolnosti a mutace *patogenu* může proniknout hlouběji do lidského těla. Obrázek č.1 zachycuje pořadí v jakém se aktivují složky imunitního systému.



Obrázek č.1: Vrstvy imunitního systému, zdroj: [Castro]

3.2.1 Fyzická vrstva

Kůže tvoří pevnou vrstvu, která odděluje vnější od vnitřního prostředí a chrání lidské tělo proti přímému proniknutí škodlivých látek, *patogenů* do biologického systému a třeba i proti UV záření apod.

Dále za lidský filtr lze považovat chloupky v nose a nebo mandle v dutině ústní. Tyto receptory čistí vzduch, který vdechujeme.

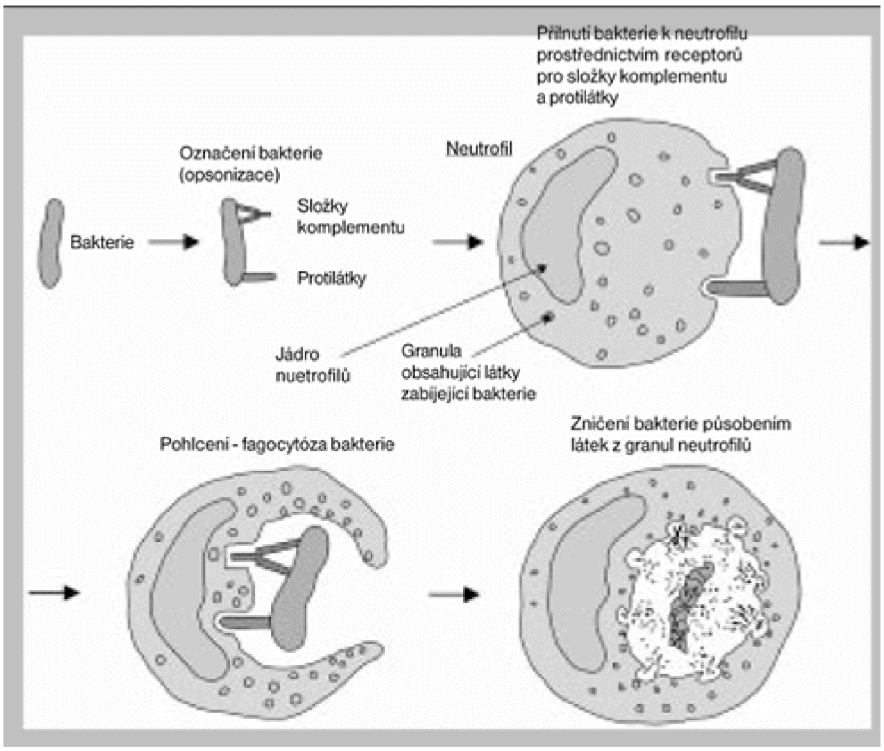
3.2.2 Fyziologická bariéra

Pod kůží se nachází kyselina mléčná, nejrůznější nasycené mastné kyseliny apod. Jelikož kůže i podkožní část mají pH kyselé, tak inhibují růst bakterií. Další fyziologickou bariérou je žaludek, tak má pH opravdu velice nízké (pH 0.9 < pH < 3.0). Toto nehostinné prostředí eliminuje opravdu velké množství organismů.

3.2.3 Vrozený imunitní systém

Jak název napovídá, je to ta část imunitního systému, který je geneticky zakódován a zdědíme jej po našich rodičích. „Důležitá vlastnost vrozené imunity je, že **není specifická**. Funguje na nejnižší biologicko-chemické úrovni a její chování by se dalo popsat jako "předprogramované". Na rozdíl od adaptivní imunity reaguje na *patogen* jen obecnou reakcí a z dlouhodobého hlediska nemá vliv na vývoj schopností imunitního systému jako celku.“[bc-neu] Vrozený imunitní systém se skládá z několika druhů buněk, které plní nejrůznější funkce:

[Motol] **Makrofágy** a **Neutrofily** jsou buňky, které se nazývají *fagocyty*. Tedy buňky schopné *fagocytózy*, to je proces, kdy fagocyt pohltí *patogen* a následně se uvnitř buňky spustí řetězec chemických reakcí, který ve svém důsledku vede ke zničení *patogenu* viz. obrázek č.2:. *Patogeny* pokryté protilátkami, nebo komplementem jsou lépe pohlceny fagocyty. „Komplement je plazmatický protein, který se váže na povrch bakterií, kde naruší membránu a fagocyty jsou přitahovány do místa infekce.“[fellner]



Obrázek č.2: Fagocytóza, zdroj: [Motol]

[Dunkova] **Dendritické buňky** jsou nejdůležitější antigen prezentující buňky, které mají schopnost stimulovat naivní T a B buňky a regulovat tak imunitní odpověď organismu. Jsou rozmístěny téměř ve všech tkáních organismu. Pohlcují antigeny a následně migrují do sekundárních lymfatických orgánů, kde zpracovaný antigen prezentují T buňkám, čímž je umožněn rozvoj účinné imunitní odpovědi.

[url-skin] **Langerhansovy buňky** aktivují T-buňky, a tím spouštějí primární T-buněčnou imunitní reakci. Hrají významnou roli při kontaktních alergiích, přijetí kožních transplantátů a dalších imunitních procesech v kůži.

[Troegel] **Imunoglobuliny** jsou protilátky, které obalí antigen a zabrání tak škodlivému účinku (činnosti). Obalením antigen označí „k likvidaci“ pro *makrofágy*. Existuje několik typů (IgG, IgA, IgE, IgD, IgM..) tedy milióny strukturálních možností.

3.2.4 Adaptivní imunitní systém

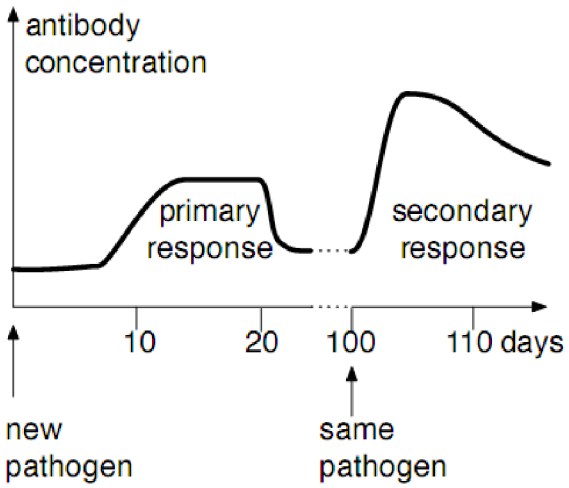
Adaptivní imunitní systém je **specifický**, to znamená, že útočí na předem známý druh patogenu a je aktivován pouze, je-li stimulován vrozenou imunitou. Adaptivní imunitní systém je tvořen dvěma základními lymfocyty. **Lymfocyt B (B-cells)** se tvoří v kostní dřeni (angl. Bone marrow). **Lymfocyt T (T-cells)** je taktéž vytvořen v kostní dřeni, ale poté dozrává v brzlíku (angl. Thymus gland).

Každý patogen je specifický svým *antigenem*, díky tomu každá cizorodá látka v biologickém systému je unikátně popsatelná. Každý lymfocyt má na svém povrchu *receptory* (pouze na jeden druh *antigenu*), kterými jednoznačně rozpoznává *patogeny.*

Při vytváření lymfocytů dochází i k vytváření všech možných variací antigenu, aby i případné *patogeny* byli detekovány, ale musí se dát pozor na různé kombinace, aby nedošlo k *autoimunitní* reakci.

[Oda2]Autoimunitní reakce je stav, kdy imunitní systém napadá sám sebe. Jak je patrno, je tenhle jev velice nežádoucí, ale jelikož jsou lymfocyty vytvářeny náhodně, proč je imunitní systém nedetekuje sám? Toto zajišťuje *self-tolerization*. V brzlíku kde lymfocyty „zrají“ existuje jeden druh lymfocytu (self-tolerized lymphocyte, někdy označované jako T-helper cells), který lymfocyty s podezřením na autoimunitní sebedestrukci eliminuje dříve, než je imunitní systém naváže na patogen, tedy jakoukoliv ničící akci.

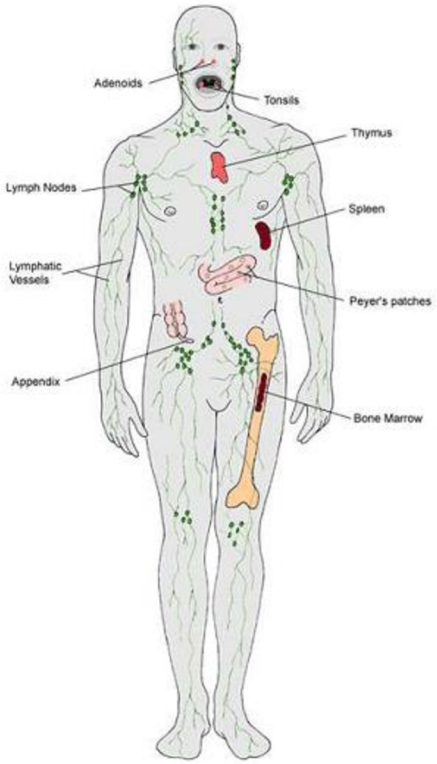
[url-nk] Lymfocyty po prvním střetnutí s cizorodou látkou zachovávají dlouhodobé informace, a proto jsou schopné při opakovaném střetnutí s tím samým antigenem reagovat rychleji a intenzivněji. Tento jev je znázorněn na obrázku č.3.



Obrázek č.3: Paměťový efekt adaptivního imunitního systému, zdroj: [Tschudin]

3.3 Souhrn

Jak jsme si mohli všimnout, v předchozích kapitolách, se nikde nevyskytuje vliv mozku na činnost imunitního systému. Činnost imunitního systému je decentralizovaná, to znamená, že každá část funguje bez řízení mozkem, pracuje samostatně a nezávisle na jiných částech. Obrázek č.4 zobrazuje orgány, které se podílejí na funkci a tvorbě buněk imunitního systému.



Obrázek č.4: Imunitní systém, zdroj [Tschudin]

4 Umělý imunitní systém

V minulé kapitole jsme si popsali jak zhruba pracuje biologický imunitní systém a díky těmto poznatkům, se budeme snažit vytvořit umělý imunitní systém (dále jen umělý systém).

4.1 Detekce

Stejně tak, jak v imunitních systémech jsme měli buňky vrozené imunity, které na základě antigenu vyhledávaly, respektive porovnávaly biologický podpis *patogenu* s antigenovým vzorem, můžeme i my v umělém systému rozpoznávat poslopnosti znaků, bitů nebo hashe. Naprostá shoda je v reálném systému těžko dosažitelná, například když na detektoru (analogie receptoru) budeme detekovat slovo *viagra* a do systému přijde pozměněné slovo *v1agr4*, tak slovo nebude rozpoznáno jako *patogen*.

Abychom se tomuto problému vyhnuli, a nebo alespoň jej co nejvíce eliminovali, používá se výpočet *afinity* (vzdálenost, podobnost). Mezi nejznámější patří:

Nechť $A = (a_1, a_2) \in R^2,\ B = (B_1, B_2) \in R^2$

- *Euklidovská vzdálenost*

$$\delta(A, B) = \sqrt{(a_1 - b_1)^2 - (a_2 - b_2)^2} \quad (1)$$

- *Manhattanská vzdálenost*

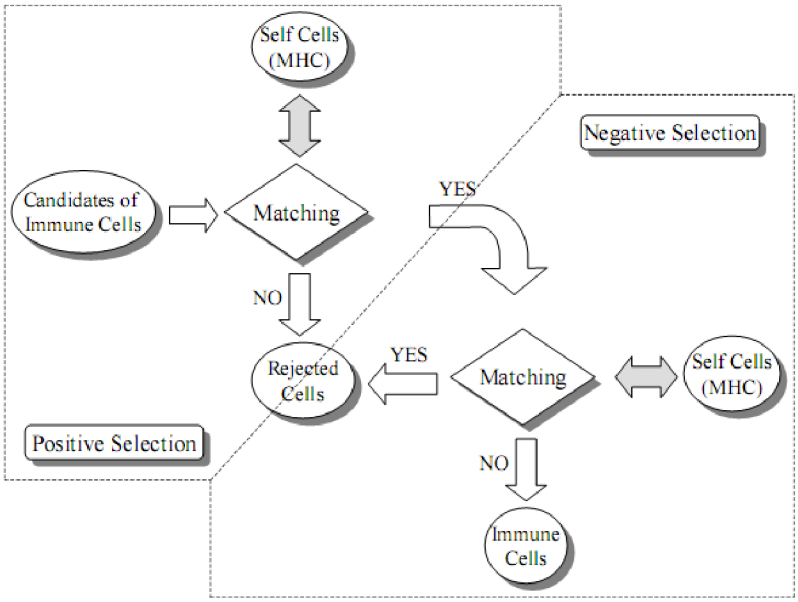
$$\delta(A, B) = |a_1 - b_1| + |a_2 - b_2| \quad (2)$$

- *Hammingova vzdálenost*

$$1010101010 = \text{receptor}$$
$$1110101011 = antigen$$
$$0100000001 = 2\ hammingova\ vzálenost$$

4.2 Algoritmy

V této podkapitole se popíšeme základní algoritmy, které se uplatňují v umělých imunitních systémech. Budeme uvažovat 3 základní množiny. Množina S, je množina obsahující prvky, které jsou systému vlastní (*sefl*), dále množina detektorů D a množina prvků N, které v systému nemají co dělat (*non-self*). Následující obrázek č.5 graficky znázorňuje algoritmus pozitivní/negativní selekce.



Obrázek č.5: Pozitivní/negativní selekce T lymfocytů, Zdroj: [Sim]

4.2.1 Pozitivní selekce

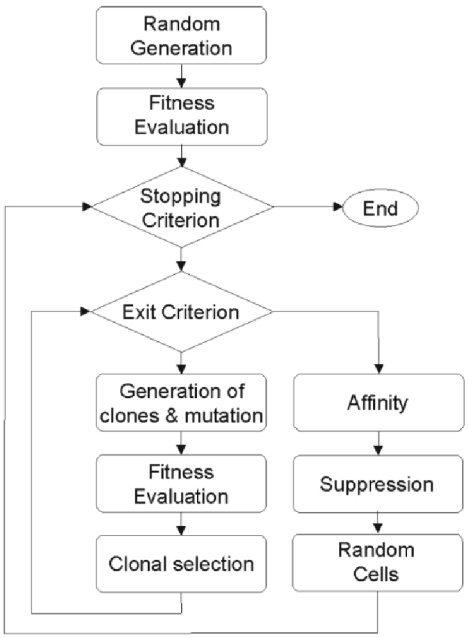
Pozitivní selekce se v biologickém imunitním systému uplatňuje na vyhnutí se akumulace zbytečných lymfocytů bez receptorů a nebo neproduktivních lymfocytů pro organismus. Buňky, které přežijí pozitivní selekci se stávají více efektivní ve vyhledávání antigenů. Tedy jedná se o selekci buněk schopných vazby na vlastní MHC (hlavní systém histokompatibility).

4.2.2 Negativní selekce

Negativní selekce popisuje proces čištění lymfocytů na základě schopnosti/úspěšnosti rozpoznávat pomocí receptorů **pouze** prvky z *non-self* množiny. Tedy pokud lymfocyt rozpozná jakýkoliv prvek z množiny *self*, je okamžitě eliminován. Probíhá již v kostní dřeni a odstraňuje autoreaktivní klony buněk.

4.2.3 Klonální selekční algoritmus

Klonální selekční algoritmus popisuje proces množení lymfocytů. Tedy, když lymfocyt úspěšně detekuje *patogen*, tak se zahájí jeho rozmnožení. U nových lymfocytů se vytvoří receptory nesoucí informaci, která byla vytvořena podle klonálního selekčního algoritmu, tím se zajistí vygenerování nejrůznějších variací. Pokud by v budoucnu došlo k modifikaci *patogenu*, tak aby byl imunitním systémem také detekován. Například v detekci spamů je to velice výhodný algoritmus, protože s výskytem tzv. *haxorů (h4x0r* – člověk, který zaměnuje znaky za znaky opticky podobné*)* je detekce variant na místě. Obrázek č.6 graficky znázorňuje klonální selekční algoritmus.



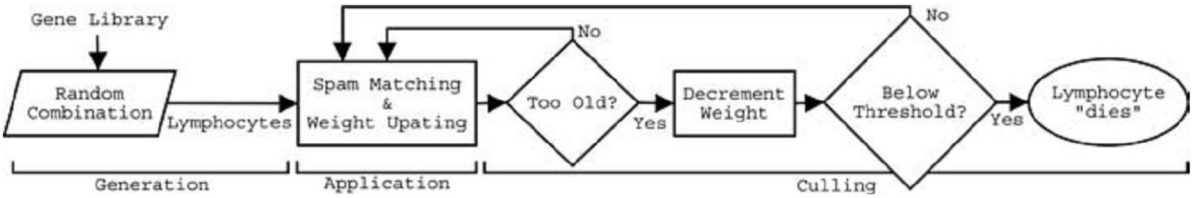
Obrázek č.6: Klonální selekční algoritmus, zdroj: [Canova]

4.3 Lymfocyt

[Oda1] Lymfocyt s detektory tvoří hlavní část umělého systému. Detektory mohou být tvořeny regulárním výrazem, nebo detekcí na konktérní slovo a jeho variace, nebo mohou obsahovat heuristiky, které zjistí například, že email je tvořen pouze velkými písmeny (tento příklad je velice nepravděpodobný, že by normální uživatel psal emaily pouze velkými písmeny). Každý lymfocyt obsahuje 2 váhové proměnné/atributy:

- spam_matched - počet označených spamů
- msg_matched - počet kontrolovaných zpráv

Obě hodnoty jsou při inicializaci lymfocytu nastaveny na nulu. Dále každý lymfocyt musí obsahovat čas svého vytvoření, aby bylo možné obměňovat staré lymfocyty za nové, nejen podle efektivity, ale i podle jejich stáří v systému. Na obrázku č.7 můžete graficky vidět algoritmus řídící životní cyklus lymfocytu.



Obrázek č.7: Životní cyklus umělého lymfocytu, zdroj[Oda2]

5 Email

[Bayliss]Elektronická pošta, běžně nazývaná *email*, je metoda odesílání zpráv uživatele z jednoho počítače (mabilního telefonu apod.) na druhý. Tyto zprávy obvykle obsahují individuální kousky textu, které je možno odeslat, i když příjemce není momentálně připojen k internetu.

Jakmile je email doručen do počítačového systému, je obvykle uložen do *poštovní schránky (mailbox)*. Dnes je obvyklé posílat emaily napříč uživateli s různým počítačovým systémem, mezi akademickými a výzkumnými institucemi, nebo mezi firmami. Rozlišujeme 2 kategorie emailů. Ty nechtěné již známe a korektní emaily, které se slangově nazývají *ham.*

5.1 Tělo emailu

[Matousek]Každá komunikace na internetu je řízena specifickým protokolem. Formát textových zpráv je prováděn pomocí specifikace popsané v RFC 2822 (RFC = request for comments – žádost o komentáře). Přenos zpráv obsahující různé multimediální data, které jsou ve standardu MIME (Multipurpose Internet Mail Extensions), jsou posílány přes RFC 2045, RFC 2046, RFC 2049. Email je složen ze dvou hlavních částí:

- **Obálka** (angl. Envelope) obsahuje informace odkud email přišel (MAIL FROM) a komu byl určen (RCPT TO).
- **Zpráva** (angl. Message) obsahuje vlastní text (body) a *hlavičku* (header). Pro naše potřeby detekci spamu poskytuje *hlavička* spoustu důležitých informací pro různé filtrační techniky, které jsou popsány následující kapitole.

Nejzákladnější a zároveň nejdůležitější položky z *hlavičky* jsou blíže popsány v tabulkách č.1 a č.2:

Skupina	Pole	Poznámka
Datum odeslání	Date:	vždy musí obsahovat lokální čas, povinné
Identifikace odesilatele	From:	povinné, identifikuje autora zprávy
	Sender:	kdo skutečně zprávu odeslal (např. sekretářka)
	Reply-To:	na kterou adresu se má odpovědět
Identifikace adresáta	To:	prvotní adresáti zprávy
	Cc:	(carbon copy, přes kopírák) informace pro další adresáty, zpráva není adresována přímo jim
	Bcc:	(blind Cc) adresáti, jejichž adresu neuvidí ostatní adresáti
Identifikace zprávy	Message-ID:	jednoznačný identifikátor zprávy generovaný počítačem, nemění se během přesměrování
	In-Reply-To:	v odpovědi nesou identifikaci původní zprávy
	References:	obsahuje reference z původní zprávy, někdy se to používá jako vlákno (thread) v diskuzních příspěvcích

Tabulka č.1: Některé hlavičky podle RFC 2822, zdroj: [Matousek]

Skupina	Pole	Poznámka
Informační pole	Subject:	krátký řetězec vyjadřující obsah zprávy, v odpovědích se předmět změní na Re: (res, lat.) a zkopíruje se předmět původní zprávy
	Comments:	další komentář těla zprávy
	Keywords:	klíčová slova oddělená čárkou
Přeposílání zprávy	Resent-Date:	původní datum zprávy
	Resent-From:	identifikace původního odesilatele
	Resent-To:	identifikace původních adresátů
Záznamy o cestě zprávy	Return-Path:	
	Received:	

Tabulka č.2: Některé hlavičky podle RFC 2822, zdroj: [Matousek]

5.2 SPAM

Termínem *SPAM* označujeme přijatou nevyžádanou poštu. V dnešní době představuje tento druh kriminality značný problém. Každý rok vynaloží firmy nemalé finanční prostředky na boj proti spamu. Do finančních ztrát se počítají jak provozní náklady na různé anti-spamové ochrany, tak čas který stráví zaměstnanci jeho čtením/mazáním. Dále většinou spam obsahuje zavirované přílohy a běžný Franta uživatel si zaviruje počítač a nebo dokonce se z jeho počítače stane bot (internetový robot, který plní příkazy svého pána/autora), který rozesílá další spamy bez vědomí uživatele.

[security]Od září roku 2004 vyšel v platnost takzvaný antispamový zákon 480/2004 Sb., který definuje, jak bojovat se spamem legislativním způsobem. Zákon reguluje nevyžádanou elektronickou inzerci a povoluje zasílat obchodní sdělení pouze podle takzvaného systému opt-in, tedy pouze s výslovným souhlasem adresáta. Proto se na konci seriózních reklamních emailů vyskytuje možnost kliknutí na odkaz, který vymaže uživatelovu emailovou adresu z jejich informačního systému.

V následujících podkapitolách si popíšeme několik druhů spamu. Informace a druhy spamů byly čerpány z [url-scam].

5.2.1 Junk Mail

Junk mail jsou klasické reklamní emaily, které nás vybízejí k nákupům nejrůznějších sortimentů zboží. Jejich obsah je častokrát tvořen spousty obrázků, tělo emailu napsané v jazyce HTML vytváří větší poutavý efekt, než „strohý“ text.

5.2.2 Chain letter

Chain letter (řetězový dopis) sice není považován za spam, ale ve své podstatě se ve společnosti za spam považuje, kvůli hromadnému přeposílání několika dalším lidem. Jsou to takové ty typické emaily zakončené „toto musíte přeposlat alespoň deseti lidem...“.

5.2.3 Mail Loop

Mail Loop (emailová smyčka) nastává, když jeden z automatických emailů spouští další a ten, který dostane email zpětně přeposílá zpět odpověď odesílateli, čímž vzniká uzavřená smyčka.

5.2.4 Stock Mail

Jedná se o podvodné emaily, které nalákají formou reklamy uživatele na velmi výhodný nákup akcií firmy, ve které spammer má předem nakoupen podíl, čili následný prodej mají se ziskem. Tato technika je také známá jako *Pump & Dump* (Vypumpuj a zahoď).

5.2.5 Porn & potent

Tato kategorie zabírá nevětší podíl ze všech zmiňovaných. Jedná se o rozesílání emailů s erotickým kontextem. Většinou obsahují zavirované přílohy. Dále nabízení různých přípravků na zlepšení sexuálních prožitků a prodlužování/zvětšování lidských částí těla.

5.2.6 HOAX

[Dzubak] V angličtině HOAX[:houks:] znamená falešná zpráva, mystifikace, podvod, žert apod. Ve smyslu spamu se jedná o šíření falešné zprávy. Nejčastěji obsahují informace o velmi nebezpečném neexistujícím virovém ohrožení a jeho devastující účinky na systém/soubory uživatele.

5.3 Emaily v číslech

Podle serveru Pingdom.com [pingdom] bylo v roce 2009 následující statistika, která byla zveřejněna 22. Ledna 2010:

- 90 trilionů - odeslaných emailů
- 247 bilionů – průměrný počet odeslaných zpráv za den
- 1,4 bilionů - uživatelských účtů
- 100 milionů – nových uživatelských účtů za poslední rok
- 81% - počet procent zpráv, které byly spam.
- 92% - vrchol počtu zpráv, které byly spam na konci roku.
- 24% - nárůst spamu od minulého roku
- 200 bilionů – počet spamů za den (za předpokladu, že 81% je spam).

Svět a spam:

Firma Symantec oznámila v dubnu roku 2010 ve své MessageLabs zprávě, které země jsou nejvíce postiženi spamem[gorum1]:

1. Itálie : 95.5%
2. Německo : 92.3%
3. Nizozemsko : 91.5%
4. Hong Kong : 91.0%
5. USA : 90.2%
6. Velká Británie : 89.4%
7. Austrálie : 89.4%
8. Kanada : 88.9%
9. Japonsko : 86.9%

Zdroje:

V dalším přehledu jsou uvedeny země, které jsou největším zdrojem ilegálních aktivit (spam, útoky na webové stránky, DDoS útoky) na internetu. Přehled obsahuje top 10 s počtem hacknutých počítačů v čtvrtém čtvrtletí roku 2009. V roce 2009 předběhla Čína Spojené státy[gorum2]:

1. Čína : 12.0%
2. USA : 9.5%

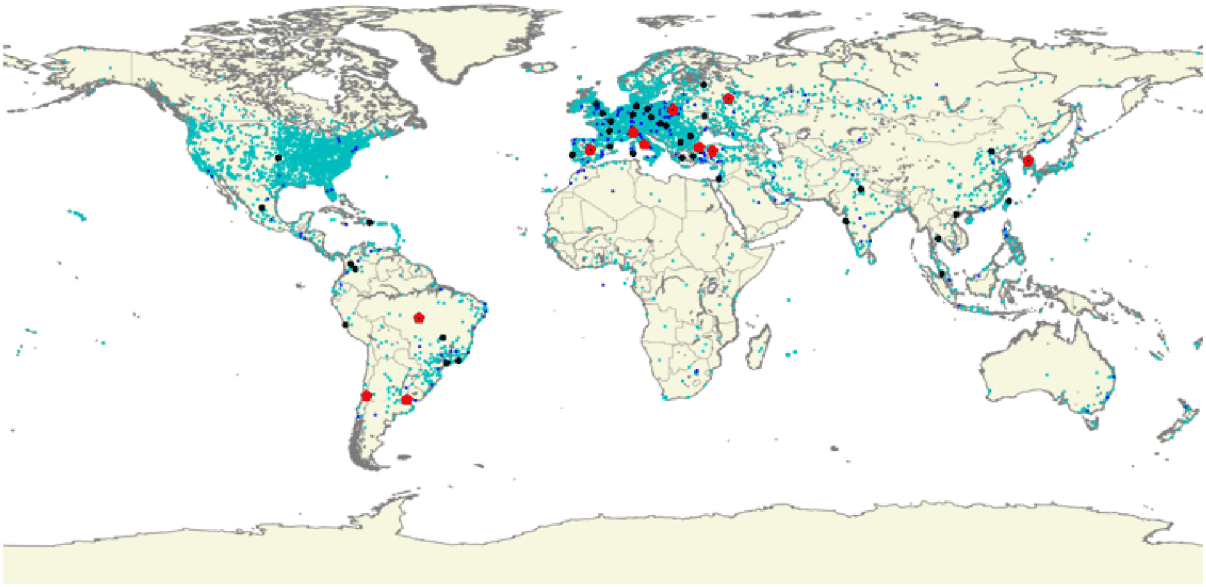
3. Brazílie: 8.5%
4. Rusko : 7.0%
5. Německo : 6.0%
6. Korejská republika : 5.0%
7. Itálie : 3.5%
8. Velká británie : 3.2%
9. Taiwan : 3.0%
10. Španělsko : 2.6%

Podle serveru www.spamhaus.org je ke dni 29. prosince 2010 následující pořadí (číslo u země reprezentuje počet aktivních spam problémů) [spamhaus]:

1. USA : 2407
2. Čína : 775
3. Rusko : 518
4. Velká británie : 309
5. Argentina: 262
6. Brazílie : 241
7. Německo : 236
8. Kanada : 208
9. Japonsko : 192
10. Itálie : 183

V mapě na obrázku č.8 je možné pozorovat statistiku průměrného počtu vytvořených spojení z každé IP adresy:

- **Světle modrá** : méně jak 100 spojení
- **Tmavě modrá** : 100 – 500 spojení
- **Černé kolečko** – velký objem dat a 500 – 2500 spojení
- **Červený pentagram** – obrovský útočný potenciál, více jak 2500 spojení



Obrázek č.8: Statistická mapa světa – zdroje spamu, zdroj: [dmz]

6 Techniky používané k detekci spamů

Než se pustíme do popisu jednotlivých technik, tak si musíme vysvětlit pár základních termínů. ***IP adresa*** hodnota/číslo, které jednoznačně určuje počítač v TCP/IP (Transmission Control Protocol/Internet Protocol) síti. ***SMTP*** (Simple Mail Transfer Protocol) je protokol, který zajišťuje přenos pošty z jednoho serveru na druhý. ***DNS*** (Domain Name Server) je hierarchický systém doménových jmen. Abychom si nemuseli pamatovat IP adresy serverů, tak nám toto velice usnadňuje zavedení *doménových jmen*, které jsou uloženy v *DNS serveru*, který překládá doménové jméno na IP adresu apod.

6.1 Blacklisting

[spammer] Blacklisting je technika, která rozhoduje, zda přijatý email je *ham* a nebo *spam*. Rozhodování se provádí na základě adresy odesílatele (hrozí zfalšování adresy), IP adresy, ze které email přišel na SMTP server.

Ve své podstatě se jedná o seznam (blacklist=černá listina) IP adres, o kterých se ví, že již v minulosti odesílali spamy. Jelikož emaily jsou ve tvaru `example@email.xx`, tak blacklistingu sekunduje DNS server. [Matousek] Databáze zdrojů spamu : `www.ordb.org`, `www.dslb.org`, `www.spamcop.net` a `www.spamhaus.org`.

6.2 Greylisting

[Faltynek][Matousek]Greylisting je metoda, kdy příchozí pošta z neznámých serverů je zadržena. Když přijde první email z nedůveryhodného serveru, tak je zpráva odmítnuta. SMTP nařizuje, že pokud nedojde k úspěšnému doručení, tak to má odesílající server zkusit znovu za určitou, předem přednastavenou hodnotu cca 24-48 hodin.

Spammeři používají k odesílání spamů jednorázové scripty a pak se snaží co nejrychleji zamést za sebou stopy, obvykle nedoručují poštu dvakrát. Databáze má tvar <IP adresa zdroje> <adresa odesílatele> <příjemce z obálky>. U greylistingu se udává úspěšnost kolem 95%.

6.3 Whitelisting

Whitelisting je přesný opak blacklistingu. Obsahuje seznam (whitelist – bílá listina) ověřených/důveryhodných IP adres. Jde ruku v ruce s graylistingem, když se ověří adresát, přesune se jeho IP adresa na tento seznam.

6.4 Zpožďování e-mailů

[Macura]Jistě jste se setkali s tím, že Vaše emaily chodí se zpožděním. Tento jev má 2 opodstatnění. První souvisí s vytížením serverů, když si představíme, že 50-70% spamů je odfiltrováno hned na vstupu, tak je to značná režie, když se bavíme v řádech desítek tisíc a víc emailů.

Druhý důvod je ten, že dochází ke zpožďování emailů z neznámých adres a upřednostňování adres známých. Můžeme tvrdit, že email přijde o 15 minut později, se přibližně odfiltruje 40% spamu.

6.5 Honeypot

Honeypot (kyblík s medem) je technika, kdy se na internet „vysadí“ falešné nebo uživatelovy skryté emailové adresy, které slouží k jedinému účelu, aby se *spam boti* „namlsali“. Princip je jednoduchý, když na tyto adresy kdokoliv cokoliv pošle, je automaticky vytvořen hash a podpis emailu a přidán do blacklistu.

6.6 Bayesův filtr

[Aguero] Filtrovat poštu je možné pomocí Bayesova filtru. Tento filtr rozhodne na základě pravděpodobnosti, že daný email s textem *w* musí být klasifikován jako spam.

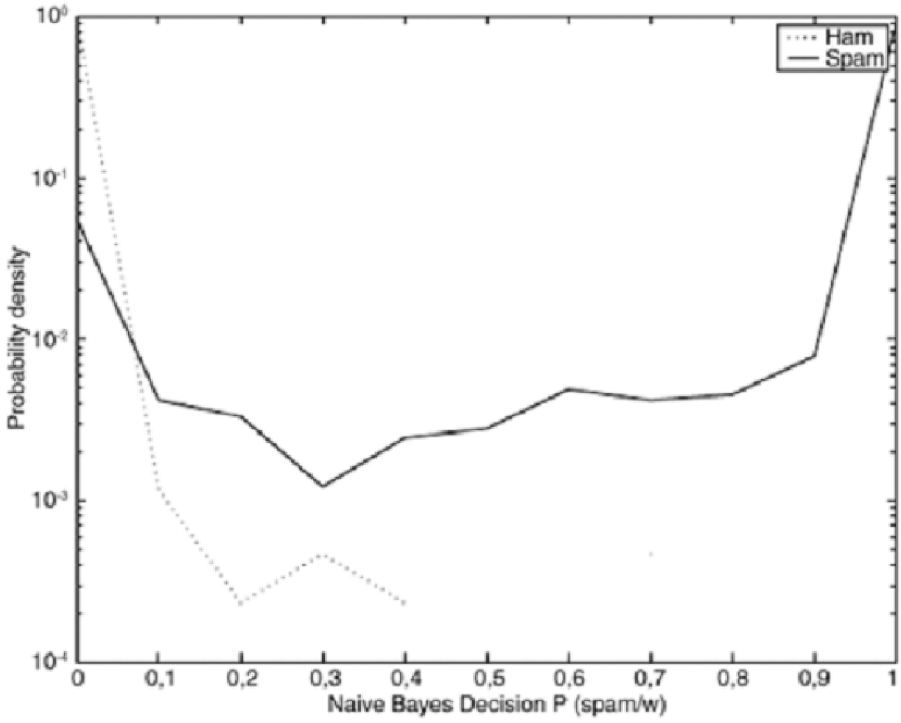
$$\mathrm{P(C = spam \,/\, w)} = \frac{P(C{=}spam)\ \prod_i P(w_i/C{=}spam)}{\sum_k P(C_k) \prod_i P(w_i/C_k)} \tag{3}$$

Statistika slov je vypočtena tak, že se zjistí počet výskytů slova v jednotlivých kategoriích { ham, spam} a podle rovnic (4) a (5) se vypočítá pravděpodobnost, do které kategorie slovo patří.

$$\mathrm{P(w/C = ham)} = \frac{\#_{in\_ham\_emails}}{\#_{in\_ham\_emails} + \#_{in\_spam\_emails}} \tag{4}$$

$$\mathrm{P(w/C = spam)} = \frac{\#_{in\_spam\_emails}}{\#_{in\_ham\_emails} + \#_{in\_spam\_emails}} \tag{5}$$

Na následujícím obrázku č.9 je zobrazena pravděpodobnost kombinace slova, která je provedena pomocí logaritmů, aby se zabránilo chybě podtečení při vytváření pravděpodobnosti.



Obrázek č.9: Bayesův filtr - Funkce hustoty pravděpodobnosti hamu a spamu pro trénovací data, zdroj: [Aguero]

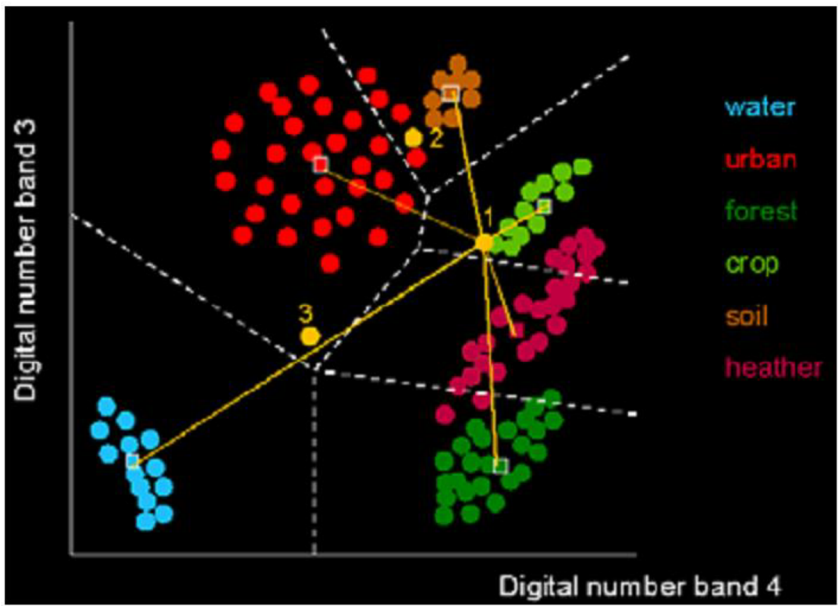
Ačkoliv formulace výpočtu skóre vypadá jednoduše, lze pomocí výpočtu úspěšně rozlišovat ham od spamu. Když se podíváme na obrázek č.9, je velice důležité vhodně zvolit práh. Při vhodně zvoleném prahu (0.4) je možné míru špatné klasifikace ham zpráv (FalseNegative) snížit na nulu. FalseNegative je závažnější než FalsePositive (pojmy vysvětleny v kap. 7).

6.7 Klasifikátory

Klasifikátor, nebo taky klasifikační metody slouží k roztřídění vstupní množiny dat do určeného počtu tříd. Jednodušší algoritmy obsahují jednoduchá třídící pravidla. Oproti nim ty složitější pracují s neuronovými sítěmi, nebo během zpracování prochází rozhodovacím stromem. U následujících obrázků body 1, 2, 3 reprezentují prvky, které se klasifikují popisovaným algoritmem, jejich cílem je graficky demonstrovat algoritmus.

6.7.1 Minimální vzdálenosti (Minimum distance)

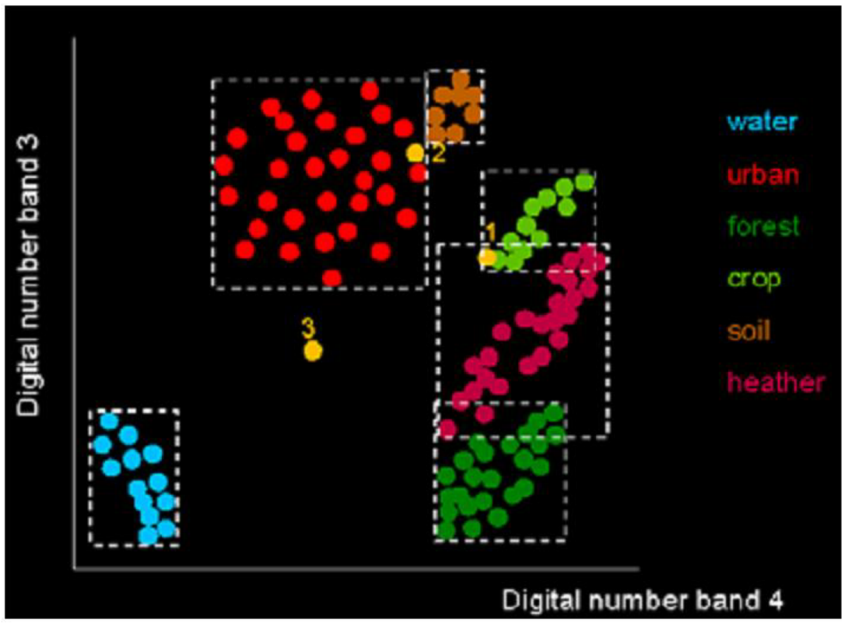
[Dolansky] Klasifikátor minimální vzdálenosti je definován jako průměrná spektrální hodnota (centroid). Příslušnost každého pixelu k patřičné třídě je určována na základě vzdálenosti od centroidu. Při určování se neuvažuje rozptyl hodnot. Rozptyl je možné omezit hranicí maximální odlehlosti.



Obrázek č.10: Klasifikátor minimální vzdálenosti středů shluků, zdroj: [Dobrovolny]

6.7.2 Pravoúhelníkový (Parallelpipeds)

[Langham] Ohraničením minimálních a maximálních hodnot ve všech hodnocených pásmech vzniknou *hyperkvádry*. Pixely mimo oblasti nejsou klasifikovány. Pokud dojde k překrytí částí *hyperkvádrů* je potřeba definovat pravidla, která zajistí zařazení do příslušné třídy.



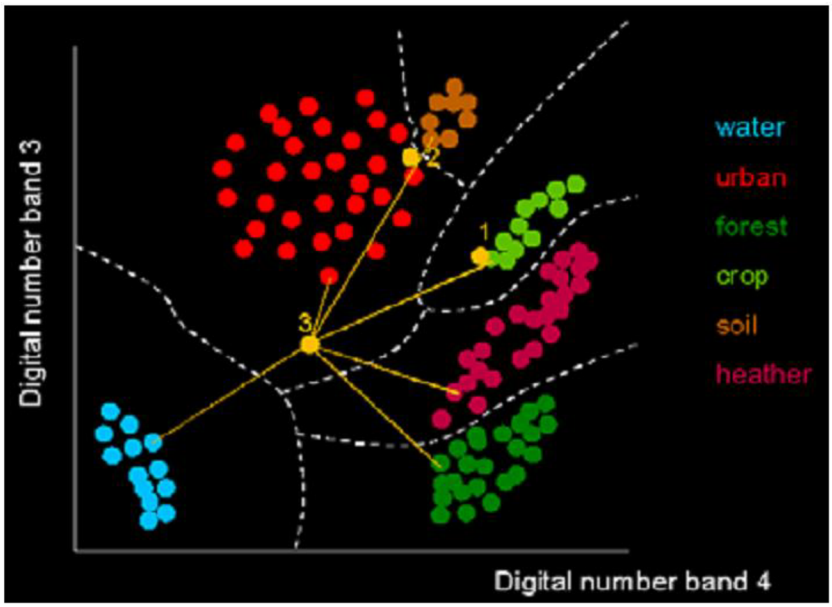
Obrázek č.11: Klasifikátor pravoúhelníků, zdroj: [Dobrovolny]

6.7.3 Nejbližší soused („K“ nejbližších sousedů)

[Dobrovolny] Jedná se o modifikaci klasifikátoru minimální vzdálenosti. Při klasifikování se hodnotí příslušnost také podle početního zastoupení okolních bodů určité třídy. Průběh algoritmu je

jednoduchý, prohledá předem stanovený počet (*k*) nebližších bodů v analyzovaném prostoru bez ohledu na trénovací množiny.

Když v okolí bodu (množina *k* sousedů) převažuje konkrétní třída, je bod do této třídy zařazen. Běžně se nastavuje parametr *k* v rozmezí hodnot 1-10, při použití vyšších hodnot dochází u výsledku k velikému podílu šumu. Dále je možné algorimus upravit způsobem, který by stanovil mezní vzdálenost souseda.



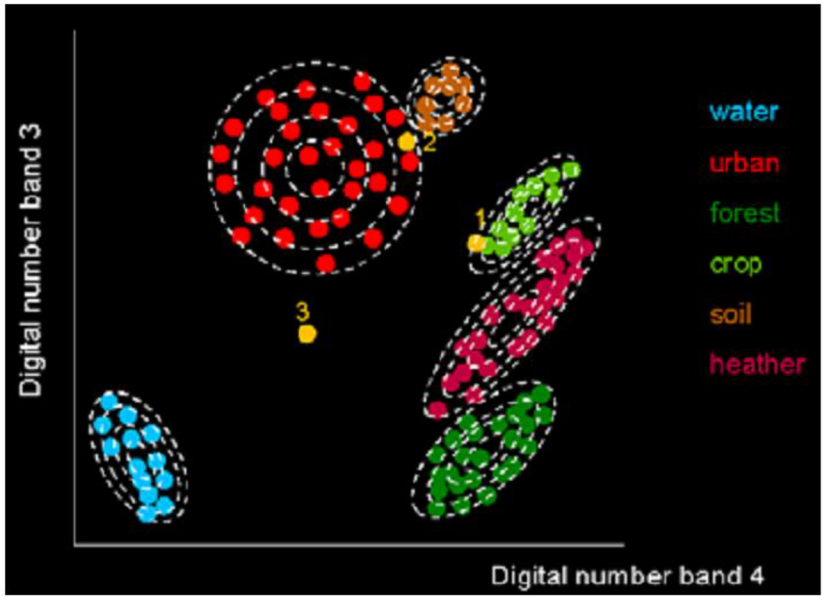
Obrázek č.12: Klasifikátor nejbližšího souseda, zdroj: [Dobrovolny]

6.7.4 Maximální pravděpodobnost (max. likelihood)

[Langham] Při zatřiďování pixelů se hodnotí:

- Rozptyl
- Korelace – vztah mezi dvěmi veličinami
- Kovariance – střední hodnota součinu odchylek obou náhodných veličin X, Y od jejich středních hodnot

Následně se vytvoří izolinie pravděpodobnosti výskytu pixelu s určitou hodnotou, poté je bod zařazen do třídy, ve které má největší pravděpodobnost výskytu.



Obrázek č.13: Klasifikátor maximální pravděpodobnosti – 2D, zdroj: [Dobrovolny]

6.7.5 K-means clustering

[IZU-10] *K-means clustering* (shlukování) je jedním z nejznámějších a nejpoužívanějších algoritmů používáných ke shlukování (*k* shluků). Algoritmus je založen na předpokladu, že *n*-rozměrné vektory $\vec{x} = [x_1, x_2, x_3, .., x_n]$, resp. koncové body těchto vektorů tvoří v *n*-rozměrném prostoru shluky, a že každý shluk *i* je reprezentován prototypem (vektorem „těžiště“ shluku) $\overrightarrow{W_i}$.

Algoritmus dale předpokládá, že shluků je *k*, a že do těchto shluků se má rozdělit všech *p* vektorů z trénovací množiny $T = \{\overrightarrow{x_1}, \overrightarrow{x_2}, \overrightarrow{x_3}, .., \overrightarrow{x_p}, \}$. Na výsledek učení má vliv použitá metrika – Euklidovská, Hammingova, atd.

Algoritmus k-means byl čerpán z [Zboril] a má následující kroky:

1. Inicializuj *k* prototypů, náhodně z trénovací množiny

$$\underset{W_j}{\rightarrow} = \underset{X_P}{\rightarrow}, j \in \langle 1, k \rangle, p \in \langle 1, P \rangle \qquad (6)$$

2. Každý vektor X_p z trénovací množiny přiřaď do shluku C_j , $j \in <1, k>$, jehož prototyp W_j má od vektoru X_p nejmenší vzdálenost:

$$\left|\overrightarrow{X_p} - \overrightarrow{W_j}\right| \leq \left|\overrightarrow{X_p} - \overrightarrow{W_i}\right| \qquad i, j \in <1, k> \qquad (7)$$

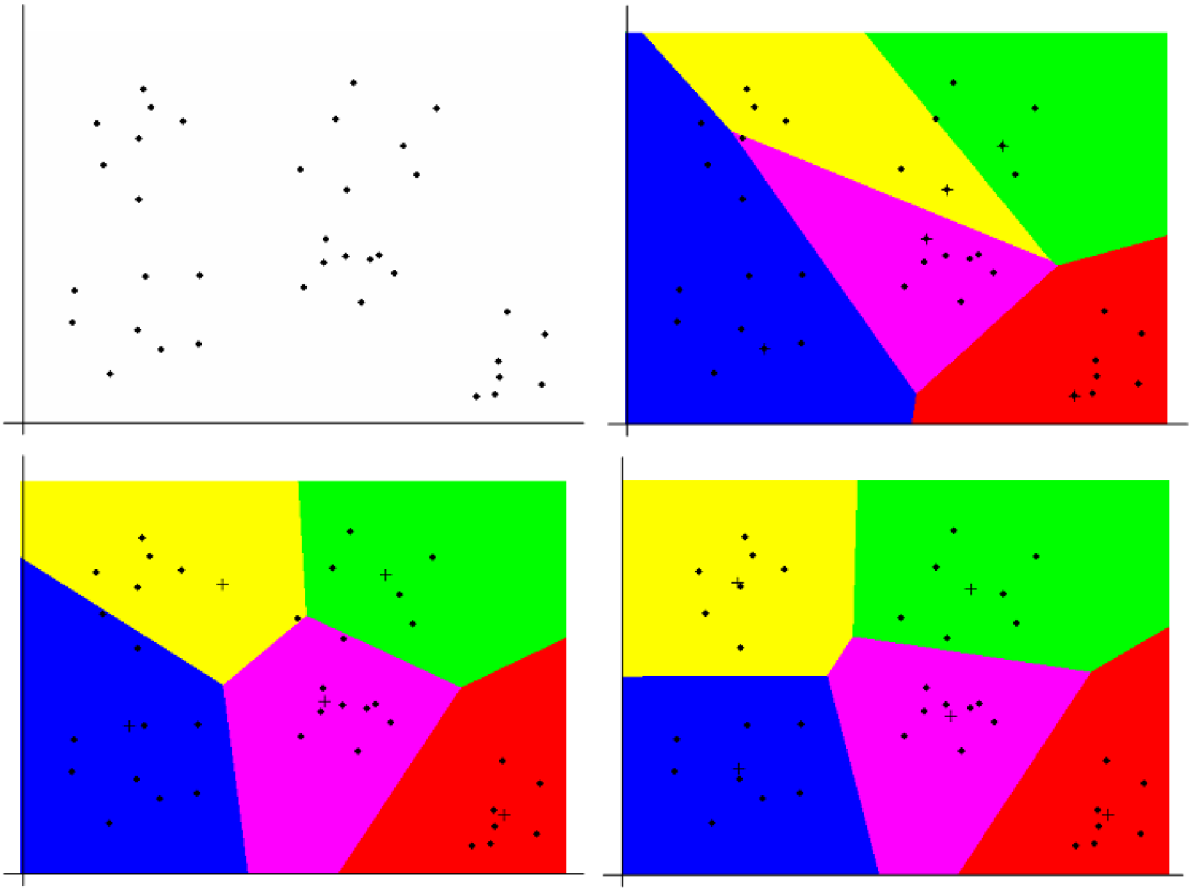
3. Pro každý shluk C_j $i, j \in <1, k>$ přepočítej prototyp W_j tak, aby byl těžštěm koncových bodů všech vektorů, které jsou k tomuto shluku právě přiřazené:

$$\overrightarrow{W_J} = \frac{\Sigma_{\overrightarrow{X_i} \epsilon C_j} \overrightarrow{X_i}}{\overrightarrow{X_i} \epsilon C_j} \qquad (8)$$

4. Vypočítej „chybu“ aktuálního stavu shlukování (součet „chyb“ všech shluků, které jsou dány součty čtverců vzdáleností všech vektorů jednotlivých shluků od středů těchto shluků):

$$E = \sum_{j=1}^{k} \sum_{\overrightarrow{X_i} \epsilon C_j} \left| \overrightarrow{X_i} - \overrightarrow{W_j} \right|^2 \quad (9)$$

5. Pokud „chyba“ E klesla, nebo pokud byl některý vektor přiřazen k jinému shluku, vrať se na bod 2.

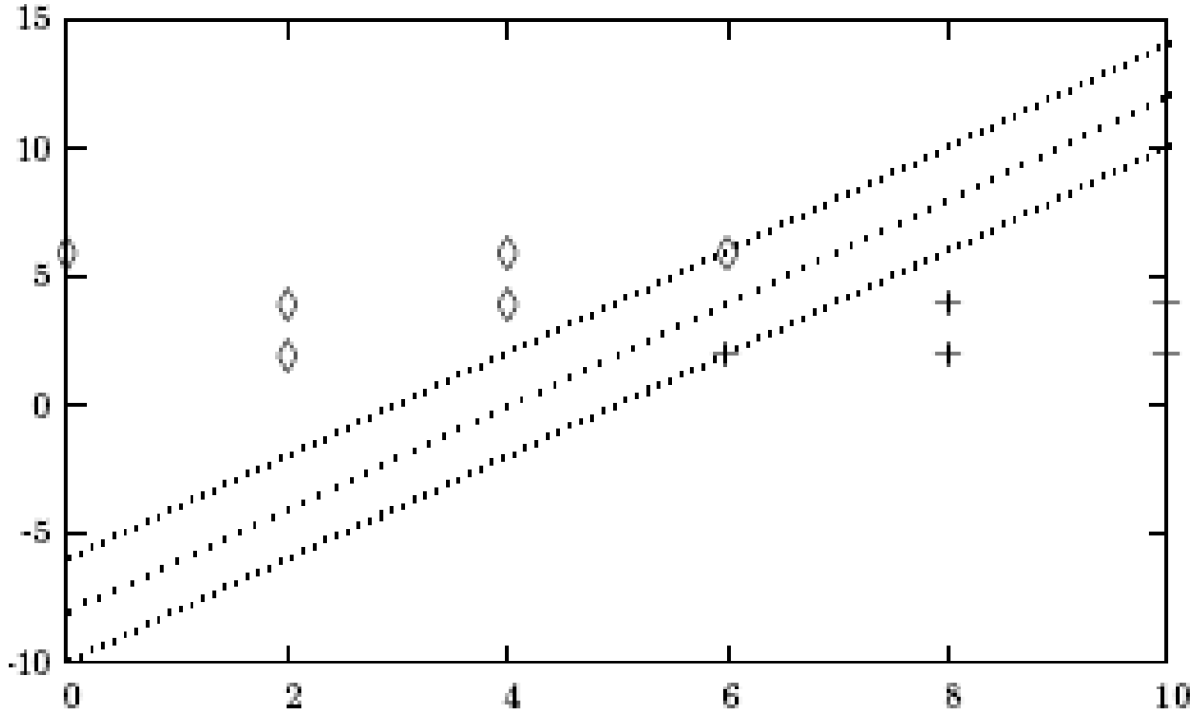


Obrázek č.14: Ukázka učení algoritmu k-means clustering, zdroj: [Zboril]

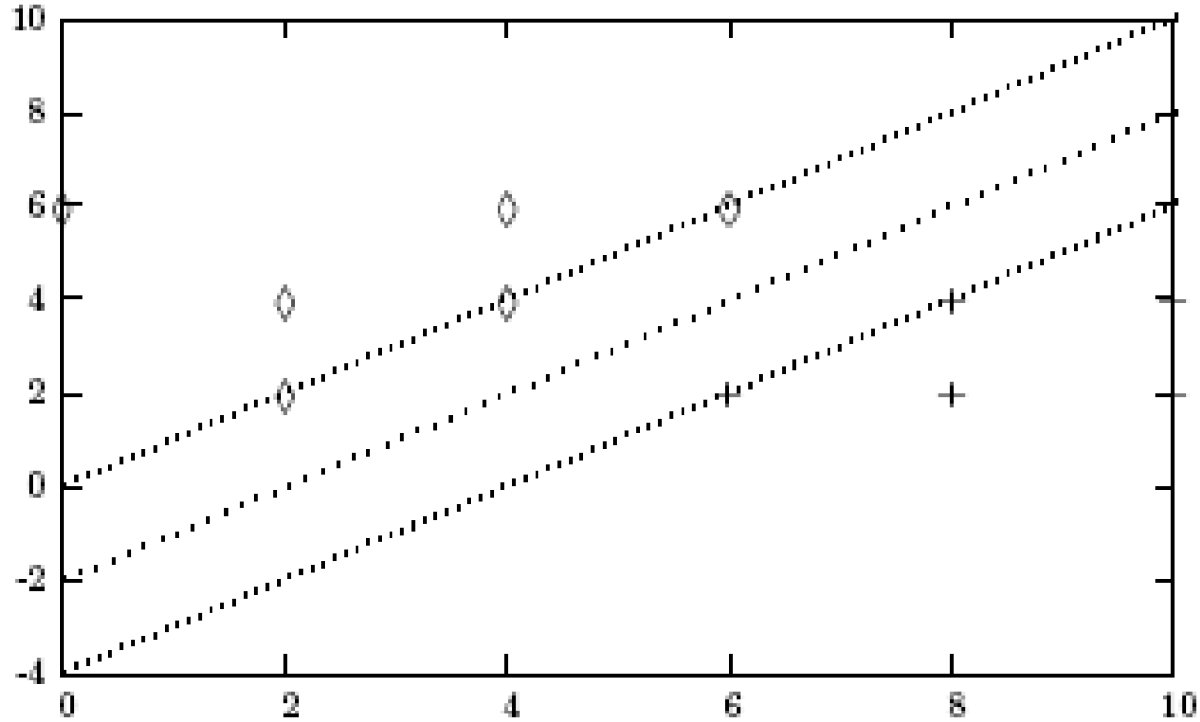
6.7.6 SVM (Support Vector Machines)

[Khorski] Support vector machines je jednou z nejpoužívanějších klasifikačních metod na text. Ve strojovém učení je trénovací vzorek množina vektorů o *n* atributech. Můžeme si představit jakýsi hyperprostor o *n* dimenzích a trénovací vzorek je množina bodů v tomto prostoru.

I když to vypadá složitě, tak nám stačí si představit v našem případě jenom 2 třídy (ham/spam). Tyto dvě třídy oddělíme hyper rovinou. Na obrázku č.15 je hyper rovina stanovena na extrémní prvky z obou tříd a proložena rovina středem jejich extrémů, na dalším obrázku č.16 je rovina proložena nejdále jak jen je to možné.



Obrázek č.15: Hyper rovina oddělující 2 třídy, zdroj: [Khorski]



Obrázek č.16 Hyper rovina oddělující 2 třídy nejdále od každé, zdroj: [Khorski]

6.7.7 Proměnný trigonometrický práh

[Abi]Proměnný trigonometrický práh (angl. Variable Trigonometric Threshold) je binární klasifikátor, který ve své strategii vybírá nejvíce významné předzpracované slovo, skóre tohoto slova se vypočítá následující rovnicí:

$$S(w) = |p_{ham}(w) - p_{spam}(w)| \quad (10)$$

Kde $p_{ham}(w)$ a $p_{spam}(w)$ jsou pravděpodobnosti slov *w* vyskytující se v trénovací množině spamu/hamu. Po projití emailu je vybráno 200 (nejlépe 650) zpráv z těchto množin, které jsou nejvíce zastoupeny a začnou se počítat vektory z párů slov (w_i , w_j). Trigonometrickým měřením se vypočítá úhel *alfa*, p_{ham} *axis: cos(alfa)* a analogicky p_{spam} *axis: sin(alfa).* Dále se vypočítá hodnoty spam positive *P(e)*a spam negative *N*(e):

$$P(e) = \sum_{(w_i,w_j)\in e} cos\big(\alpha(w_i, w_j)\big) \quad (11)$$

$$N(e) = \sum_{(w_i,w_j)\in e} sin\big(\alpha(w_i, w_j)\big) \quad (12)$$

Nakonec se rozhodne, zda je zpráva *e* ham/spam následujícím způsobem:

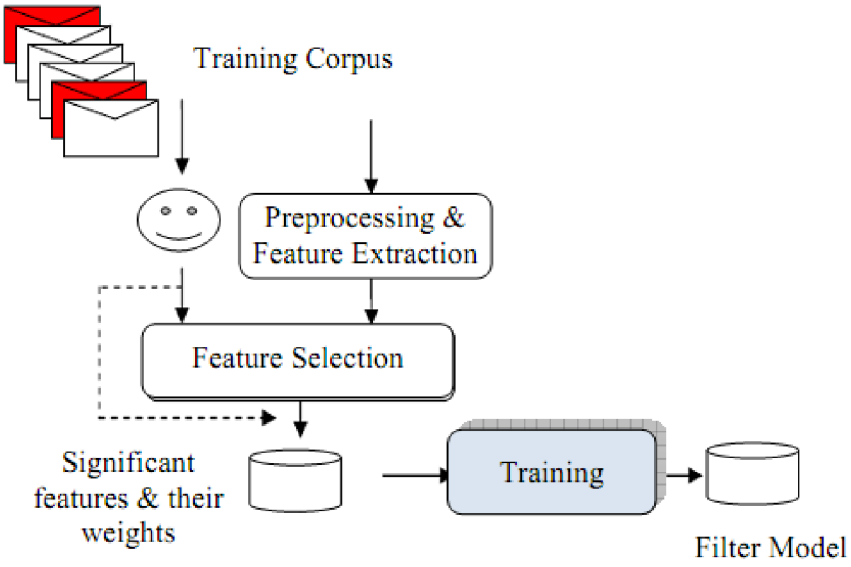
$$\begin{cases} e \in ham, & if\ \frac{P(e)}{N(e)} \geq \lambda_0 + \frac{\beta - np(a)}{\beta} \\ e \in spam, & otherwise \end{cases} \quad (13)$$

Kde $lambda_0$ je konstatní práh rozhodující o spam/ham, experimentálním měřením byla zjištěna hodnota 1,3. Dalším parametrem je *beta*, který byl použit v abstraktní klasifikaci experimentu k regulaci *np(a)*, který počítá počet označených abstraktních bílkovin. Proto předchozí rovnice může být redukována pouze na:

$$\begin{cases} e \in ham, & if\ \frac{P(e)}{N(e)} \geq 1{,}3 \\ e \in spam, & otherwise \end{cases} \quad (14)$$

6.8 Ant Colony

V roce 2009 uvedl na konferenci CEC (Congress on Evolutionary Computation) El-Sayed M. El-Alfy[El-Alfy] klasifikace spamu s optimalizací *ant colony*. Jak název napovídá, vychází se z mravenčí kolonie, respektive kooperativního chování mravenců, kdy se mohou načit co nejkratší cestu mezi hnízdem a zdrojem potravy. Mravenci spolu komunikují ukládáním chemické látky, zvané feromony. Jak se pohybují je tato látka ukládána na jejich cestách. Postupně je snižována na všech cestách v důsledku odpařování. Nicméně množství feromonů, které mravenec uloží na cestu je přímo úměrné kvalitě cesty, tedy kvalitnější/kratší cesty budou více označené a budou pro ostatní mravence atraktivnější. V případě umělých mravenců (agentů), cesty představují pro kandidáta vyřešení problému.



Obrázek č.17: Konstrukční model pro spamový filter, zdroj: [El-Alfy]

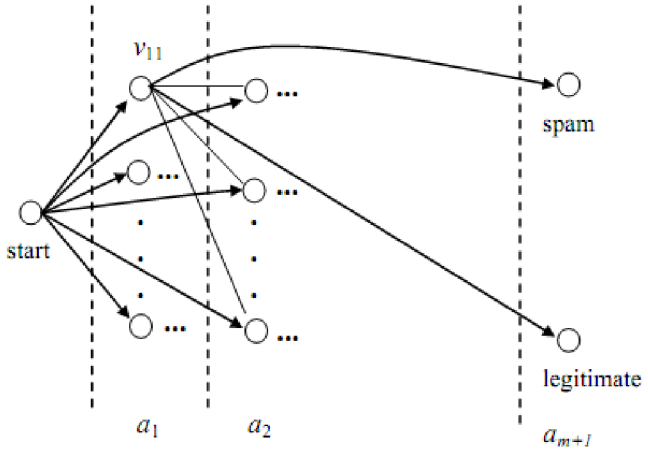
Zpracování probíhá ve třech fázích. **První fáze** je *předzpracování vstupních dat*. Zde je provedeno odstranění HTML tagů, zbytečných slov jako jsou spojky, zájména apod., dále vytvoření *tokenů*, tedy slov, frází a různých vzorů, které vyhledávají například *$*. Převedení slov do základních tvarů. Jestliže se token vyskytuje pouze párkrát v jakékoliv kategorii (spam/ham) je odstraněn.

Druhá fáze odpovídá *funkční selekci*, to znamená, že tokeny z předchozí fáze, jsou podle četnosti ohodnoceny váhou, a ty které převyšují určitý práh, jsou zachovány. Každý email je popsán množinou atributů $(a_1, a_2, ..., a_m, a_{m+1})$, kde m určuje počet prediktivních atributů a $m+1$ určuje, do které kategorie daná množina patří. Tyto atributy mohou být kategorické nebo numerické.

Třetí fáze je fáze *trénovací*. Před začátkem trénování jsou všechny numerické atributy diskretizovány. To znamená, že každému atributu je přiřazena konkrétní množina hodnot $a_i \in \{v_{i1}, v_{i2}, .., v_{ik_i},\}$, kde k_i je počet možných hodnot připadající na atribut a_i. Jsou vytvořeny indukční pravidla, které jsou podobné Ant-Miner algoritmu, které mají následující tvar:

$$IF\ t_1\ AND\ t_2\ ...THEN\ c$$

Kde t_i je term, respektive tvar atributu (= hodnota) a c je kategorie emailu. Pro vyhnutí se neplatným pravidlům, je každý atribut použit maximálně jednou. Chceme-li zjistit dostatečný počet pravidel, která mohou být použita v klasifikaci, je první problém reprezentován jako orientovaný graf skládající se z vrcholů a hran, jak je uvedeno na obrázeku č.18, kde neorientované hrany jsou obousměrné. Každý vrchol reprezentuje hodnotu v_{ij} od $i=1$ do $m+1$ a $j=1$ do k_i . Fiktivní uzel je přidán, aby reprezentoval start. Každá cesta reprezentuje vztah mezi dvěma vrcholy. Aby se zajistilo, že bude každý atribut použit v každém vrcholu maximálně jednou, jsou vrcholy mezi sebou propojeny, tím se získají další atributy.



Obrázek č.18: Částečný graf reprezentující spamový filter založený na *ant-colony,* zdroj: [El-Alfy]

Při spuštění algoritmu, jsou úrovně feromonu na všech cestách stejné a trénovací množina je inicializována na hodnoty z trénovací fáze a s prázdným seznamem pravidel. Za nejlepší pravidlo se považuje to, které je vytvořeno iterativně mravenci. Každý mravenec se řídí následující rovnicí:

$$p_{ij} = \frac{n_{ij} \cdot T_{ij}(t)}{\sum_{i=1}^{m} x_i \sum_{j=1}^{k_i} \left(n_{ij} \cdot T_{ij}(t)\right)} \tag{15}$$

Kde x_i je binární proměnná reprezentující, zda byl atribut a_i již použit ($x_i = 0$) a nebo ne ($x_i = 1$). $T_{ij}(t)$ je aktuální úroveň feromonu přiřazeného k termu $a_i = v_{ij}$. n_{ij} je předpočítaná a uložená hodnota z každého termu, vypočítaná následující rovnicí:

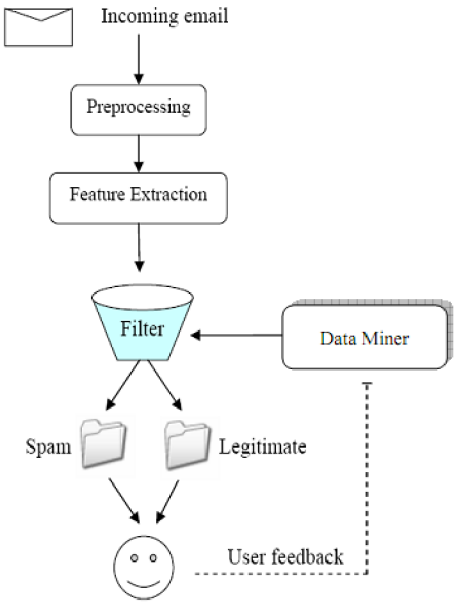
$$\mathrm{n_{ij}} = \frac{1 - \mathrm{H}(\mathrm{C}|\mathrm{a_i}=\mathrm{v_{ij}})}{\sum_{\mathrm{i}=1}^{\mathrm{m}} \mathrm{x_i} \sum_{\mathrm{j}=1}^{\mathrm{k_i}} \left(1 - \mathrm{H}(\mathrm{C}|\mathrm{a_i}=\mathrm{v_{ij}})\right)} \tag{16}$$

Kde $H(C|a_i = v_{ij})$ je entropie vypočítaná z:

$$\mathrm{H}(\mathrm{C}|\mathrm{a_i} = \mathrm{v_{ij}}) = -\sum_{\mathrm{c}\in\{\mathrm{spam,ham}\}} \left(\mathrm{P}(\mathrm{c}|\mathrm{a_i} = \mathrm{v_{ij}}) \cdot \log_2 \mathrm{P}(\mathrm{c}|\mathrm{a_i} = \mathrm{v_{ij}})\right) \tag{17}$$

Datová struktura - každý email je charakterizován 57 vstupními atributy a je označen spam/ham. Atributy 1-48 obsahuje informace o nejvíce se vyskytujících se slovech, speciálních znacích jako je třeba $ a nejdelší po sobě jdoucí sekvence znaků. Atributy 49-54 udávají procentuální zastoupení písmen, 55 a 56 udává průměrnou a maximální délku, respektive nepřerušenou posloupnost velkých písmen ve zprávě. Atribut 57 udává celkový počet velkých písmen.

Úspěšnost byla simulací v MATLABu naměřena v rozpětí 81 – 91 %. Na obrázku č.19 je znázorněn výsledná analogie zpracování zpráv.



Obrázek č.19: Nasazený model pro filtrování spamu, zdroj: [El-Alfy]

6.9 Používané nástroje

6.9.1 SpamAssasin

SpamAssasin je nekomeční nástroj na filtrování nevyžádané pošty pomocí regulárních výrazů, Bayesova filtru, využívá vlastní blacklist + online databáze, DNS a další. Jedná se o multiplatformní nástroj pracující pod aplikačním serverem *Apache*. Je možné jej integrovat do poštovního serveru a díky tomu získat možnost, automaticky filtrovat poštu.

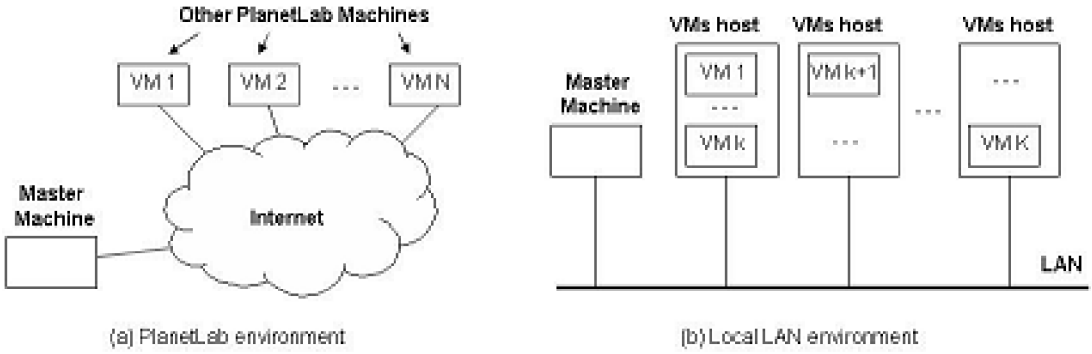
Klasifikace spamů probíhá tak, že SpamAssasin detekuje jemu známé znaky a podle závažnosti „přestupku“ přiděluje trestné body, po překročení stanovené hranice se rozhoduje *spam/ham.* Příklady „přestupku“: Otazník na konci předmětu, mail psán velkými písmeny, obsahuje jenom obrázek, ověření platnosti odesílatele [Krcmar].

6.9.2 AntispamLab

[Seraf] AntispamLab může vytvořit a používán ve dvou testovacích prostředích. Obrázek č.20(a) ukazuje spuštění jednoho stroje PlanetLab (hlavní stroj), kterému se ostatní stroje jeví jako virtuální stroje (angl. Virtual Machine) dosažitelné přes internet. Nástroj využívá testovací systém na těchto virtuálních strojích, které vytváří síť e-mailových serverů, filtrů a simulovaných uživatelů a spammerů. Dále běží běžné testy, shromažďují a zpracovávají se filtrované výsledky.

Na obrázku č.20(b) je zobrazeno LAN prostředí. Jedná se o případ, kdy uživatel nástroje chce vytvořit více virtuálních počítačů pomocí obrazu virtuálního stroje. Toto je podmíněno tím, že

uživatel musí spustit lokální DNS server, který namapuje názvy virtuálních strojů na příslušné IP adresy



Obrázek č.20: Ukázka vytvořených dvou testovacích prostředí pomocí AntispamLab, zdroj:[Seraf]

7 Návrh systému na bázi UIS

7.1 Použité nástroje

[url-MS] Program byl vytvořen v nástroji Visual Studio 2010 a programovacím jazyce C#. Výhodou zmíněného nástroje je integrované prostředí, které zjednodušuje celý proces vývoje od návrhu až po nasazení.

Dále jsem použil framework .NET 4.0. V uvedeném frameworku nově přibyla třída *Parallel*, díky které lze velice jednoduše paralelizovat některé úlohy, což je více než na místě v dnešní době, kdy vícejádrové procesory jsou čím dál dostupnější v běžných PC.

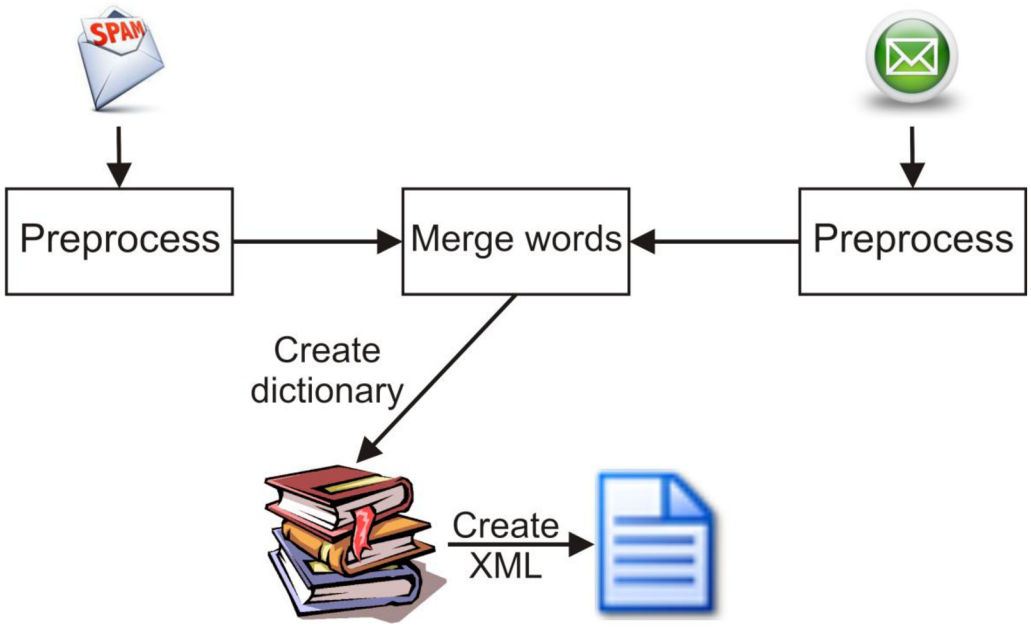
7.2 Učení

Učení lymfocytů se provádí na třech úrovních. Následující tabulka č.3 ukazuje bodové ohodnocení na jednotlivých úrovních.

Layers	User (from files)	AIS	User feedback (false negative / false positive)
Ham / Spam	+2 / -2	+1 / -1	+10 / -10

Tabulka č.3: Ohodnocení slov na jednotlivých úrovních

První úroveň je učení kandidátních lymfocytů, které jsou tvořeny slovy (unigramy), které se získají extraxcí ze souborů emailů, které jsou členěny na množinu spam souborů a ham souborů. Z pravidla platí, čím větší je počet trénovacích souborů, tím lepší obdržíme výsledky (za optimální se považuje 1000 souborů v každé množině). Každý soubor se zpracuje a extrahuje se z něj tělo zprávy. Tělo se rozdělí se na jednotlivá slova (unigramy) a každému se přiděli atribut value, který vystihuje četnost jeho výskytu v emailech když se prochází množina obsahující ham zprávy, tak se ke slovům přičítá hodnota +2, v případě spamů se přičítá -2. V příloze č.1 je možné se podívat na část výsledného XML souboru, který se po zpracování emailůvých souborů vytvoří, aby nebylo potřeba při každém spuštění programu, provádět učení. Průběh učení je reprezentován v blokovém diagramu č.21.



Obrázek č.21: Grafické znázornění fáze učení.

```
Algorithm: Preprocess HAM
    hams[] = {}
    foreach file in HAM do:
        body = regularExpresion getBodyFromFile(file)
        remove from body all HTML tags
        words[] = Split Body to words
        foreach word in words do:
            if word exist in hams then
                hams[word] += 2
            else
               //word doesn't exists in dictionary of  hams, due to
               insert word into dictionary of hams
                push word into hams
            end if
        end foreach
    end foreach
```

Algoritmus č.1: Učení nových slov z trénovací množiny obsahující korektní emaily

```
Algorithm: Preprocess SPAM
    spams[] = {}
    foreach file in SPAM do:
        body = regularExpresion getBodyFromFile(file)
```

```
        remove from body all HTML tags
        words[] = Split Body to words
        foreach word in words do:
            if word exist in spams then
                spam[word] -= 2
            else
               //word doesn't exists in dictionary of  spams, due
               to insert word into dictionary of spams
                push word into spams
            end if
        end foreach
    end foreach
```

Algoritmus č.2: Učení nových slov z trénovací množiny obsahující spam

```
Algorithm: Merge words
    INPUT: hams[], spams[]
    OUTPUT: result[] //dictionary of all words with value
    result[] = {}    //empty dictionary
    result = hams    // fill result by dictionary of hams
    foreach spam in spams do:
        if spam exist in result then
            //ham.value + spam.value of the same word
            result[spam] -= spams[spam]
        else
          //spam word doesn't exists in final dictionary, due to
          insert word into dictionary
            push spam into result
        end if
    end foreach
```

Algoritmus č.3: Výsledné vytvoření slovníku

Například při zpracovávání ham a spam souborů, jsme zjistili následující výskyty jednotlivých slov, viz tabulka č.4:

slovo	Výskyt v HAM (počet)	Výskyt v SPAM (počet)
hello	10	2
buy	1	7
time	3	1
problem	6	0
work	5	1
sick	4	8
rolex	0	11

Tabulka č.4: Příklad výskytu slov v trénovacích souborech.

Podle předcházející tabulky č.4 můžeme vypočítat výslednou hodnotu (Value) příslušného slova. Tedy pro slovo *hello* budeme počítat: `10*2 + 2*(-2) = 16`. Analogicky pro další slova vznikne tabulka č.5.

Word	Value	spam?
hello	16	false
buy	-12	true
time	4	false
problem	12	false
work	8	false
sick	-8	true
rolex	-22	true

Tabulka č.5: Výsledná tabulka s ohodnocením

Druhá úroveň se uplatní ve fázi testování emailů. Atribut value je inkrementována/dekrementována hodnotou +-1 a to tak, že když se email klasifikuje do třídy ham/spam, tak se upraví všechny atributy slov, které tělo emailu obsahuje. Tedy pokud uvažujeme zprávu, která bude obsahovat text: *„Hello, buy Rolex. BUY! BUY! BUY!“*. Bude tato zpráva pravděpodobně klasifikována jako spam a tabulkač.5 bude upravena na:

Word	Value	spam?
hello	**15**	false
buy	**-13**	true
time	4	false
problem	12	false
work	8	false
sick	-8	true
rolex	**-23**	true

Tabulka č.6: Upravená tabulka s ohodnocením po klasifikované zprávě.

Všimněme si, že hodnota *buy* se změnila pouze o *-1*. Je to dáno tím, že když se lymfocyt naváže na testovanou zprávu, tak se naváže právě jednou, tedy i update se provede pouze jedenkrát.

Třetí úroveň je v našem případě pouze teoretická, jelikož neimpementujeme reálný systém, kde by figuroval uživatel, ale kdyby jsme jej zakomponovali, tak na téhle úrovni by byla zpětná vazba od uživatele, která by rapidněji ovlivňovala databázi. Tohle můžeme znát například z komerčních emailů (Seznam.cz apod.), kde je tlačítko „Smaž jako SPAM“, respektive „Tohle není SPAM“.

7.3 Výběr lymfocytů

V běžných umělých imunitních systémech a konkrétně v [Oda1] se nejprve vytvoří lymfocyty z knihovny fragmentů a pak se nechají trénovat na trénovací množině dat. V našem případě máme již z trénovací fáze kandidátní slova/lymfocyty a z nich vybíráme a formujeme finální lymfocyty. Následující dva algoritmy popisují výběr lymfocytů z kandidátních lymfocytů- a jejich uložení v našem případě do datové struktury slovník/dictionary:

```
Algorithm: CreateLymphocyte
    INPUT: word, value
    detektor = word
    value = value
    msg_match = 1
        if value > 0 then
        spam_match = 0
     else
        spam_match = 1
     end if
```

Algoritmus č.4: Formování lymfocytů.

```
Algorithm: CreateLymphocytes
    INPUT: dictionary<string, int>   //string - word, int - value
    lymphocytes[] = {}
    foreach pair<string, int> in dictionary do:
        if Abs(pair.Second) > 10 then
          lymphocyte = CreateLymphocyte(pair.First, pair.Second)
          push lymphocyte into lymphocytes
        end if
    end foreach
```

Algoritmus č.5: Výběr a formování lymfocytů

Každý lymfocyt má atribut `spam_matched` a `msg_matched`, jak jsme si popsali v kapitole 4.3 (tyto atributy se aplikují v rovnici č.18 a díky nim je prostor pro možná rozšíření, viz. kap. 10) a `value`, která určuje ohodnocení lymfocytu slova, které interpretujeme jako receptor/detektor pro rozpoznání spamu a hamu a je jeho nejdůležitejší částí. V našem případě je detektor tvořen unigramem/slovem, o kterém na základě atributu `value` víme, zda se jedná o tzv. spam receptor/slovo, nebo ne. Jestliže je `value <= 0`, potom se jedná o spam receptorové slovo. Ve finální verzi používáme místo jednoduchého slova/unigramu jeho zápis ve formě regulárního výrazu. Lze to interpretovat jako verzi klonálního selekčního algoritmu ve tvaru *haxorovací funkce* (h4x0r, haxx0r, nebo taky haxxor = lidé, kteří přepisují písmena jejich „optickými" synonymy), která nám přepisuje písmena na alternativy, tedy uveďme si příklad:

- A/a => 4, @
- E/e => 3
- I/i => 1, !, |
- T/t => 7
- B/b => 8
- S/s => 5

A spousta dalších možností. Díky tomuto přepisu můžeme dostat například ze slova `aloha` následující výstupní množinu alternativ `{4loha, al0ha, aloh4, 4l0ha, 4loh4, al0h4, 4l0h4, @loha, ..}`. Kdyby jsme uvažovali ještě L/l => `|_` , nebo H/h => |-|, tak by nám dramaticky narůstala výstupní množina a výstupní regulární výraz typu `4loha|al0ha|aloh4|4l0ha|4loh4| ... |@l0h4|@l0h@` by byl relativně neefektivní na sestavení i vlastní provedení.

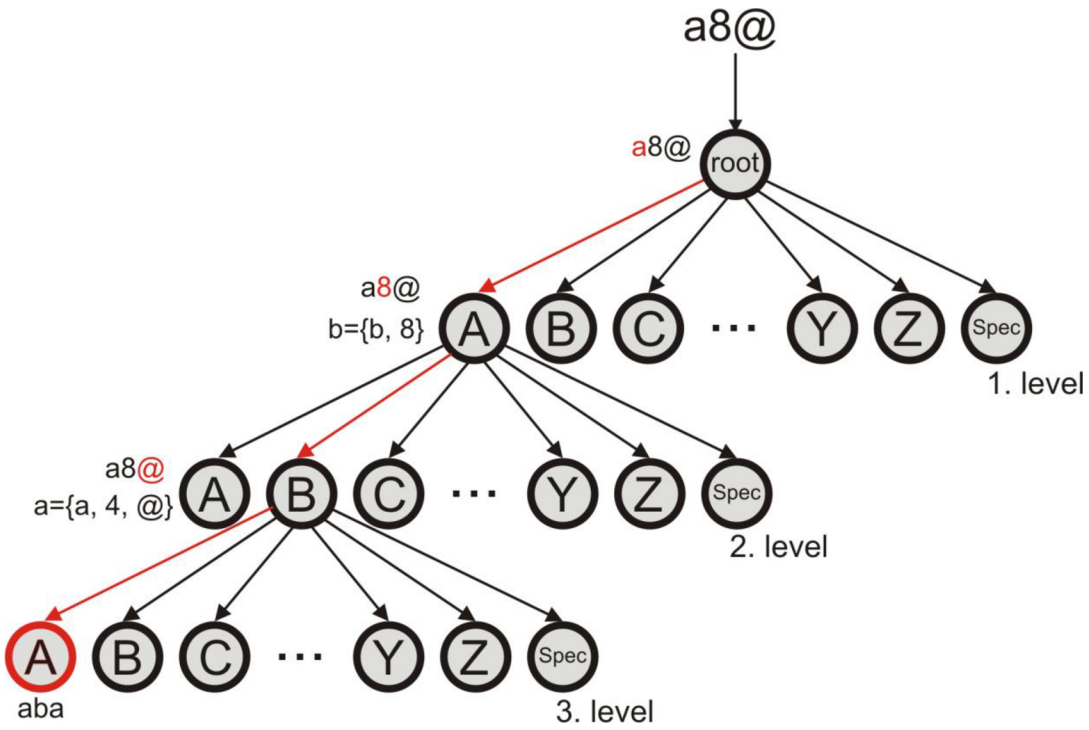
Proto se nabízí daleko přívětivější řešení, že se vytvoří také regulární výraz (ignorování case-sensitive) na základě uvedených pravidel, tedy například ze slova `elita` lze vytvořit `[e3]l[i1!|][t7][a4@]`. Takto máme elegantním způsobem pokryté všechny variace. Případné vylepšení je ještě `[e3]+l+[i1!|]+[t7]+[a4@]+`, pak je možné detekovat i různě dlouhé řetězce, kterými by se nás útočník snažil ošálit, třeba `eelita`.

Díky použití regulárního výrazu, jakožto detektoru, má lymfocyt „fyzicky" pouze jediný detektor, ale prakticky má počet detektorů, kolik variací, lze z daného slova získat. Jedinou nevýhodou použití regulárního výrazu jako detektoru je ten, že aplikování 10 000+ lymfocytů na email je velmi pomalé. Je to dáno prohledáváním všech možných variací, kterých regulární výraz nabývá.

Alternativou pro regulární výraz je vytvoření nebinárního stromu, který má 27 uzlů, kde 1-26 je určeno pro písmena a 27. uzel je pro speciální znaky, jako může být jednoduchý apostrof v anglických slovech, pomlčka, apod. Každý lymfocyt je ve stromu uložen tak, že slovo, které má

přiřazeno je detektor. 1. úroveň stromu určuje první znak detektoru. Tedy lymfocyt s detektorem obsahující slovo *ahoj* bude uložen na 4.úrovni.

Aby jsme dosáhli detekce pozměněných slov, provádí se v každém uzlu reverzní převedení znaku na písmeno. Například znaky {1, !, |} převedeme na *i*, tedy opačný postup haxorovací funkce, nazývejme tedy tento postup *reverzní haxorovací funkce*. Dalo by se říci, že ke každému uzlu, se provádí ad-hoc klonální selekční algoritmus. Na obrázku č.22 je znázorněna struktura takového stromu a vyhledání lymfocytu, který detekuje slovo *aba,* tedy varianta *a8@*.



Obrázek č.22: Uložení lymfocytů ve stromu a ukázka vyhledávání.

Díky stromovému uspořádání, nám dramaticky vzroste rychlost. Klasické prohledávání/porovnávání slov obsažených v emailu se slovy uloženými ve struktuře (uvažujme uspořádané pole) má kvadratickou časovou složitost O (N^2). Vyhledání slova nám zabere pouze tolik kroků, kolik obsahuje slovo písmen, tedy máme lineární časovou složitost O (N). Vkládání a hledání v stromové struktuře lze implementovat jednoduchou rekurzí.

Jednoduchý příklad vyhledání případného spamového slova *a8@* k existujícímu, uloženému lymfocytu *aba*. V kořenovém uzlu (root) se neděje nic zajímavého, prostě jednoduše se zanoří do větve *a*. Nyní jsme na 1. úrovni a na porovnání máme *8*, zavoláme *reverzní haxorovací funkci* a ta nám vrátí *b*, tak se zanoříme do větve *b*. Kdyby funkce nenašla žádný ekvivalent, tak by nám vrátila *Spec*. Na 2. úrovni je analogický postup. Jelikož jsme prošli všechna písmena, je pro nás poslední zanoření listem a zjistíme, že v daném listu je uložen lymfocyt, tedy k danému slovu, existuje lymfocyt. Výsledkem vyhledávání je buď lymfocyt a nebo `null`.

7.4 Testování zpráv a detekce spamů

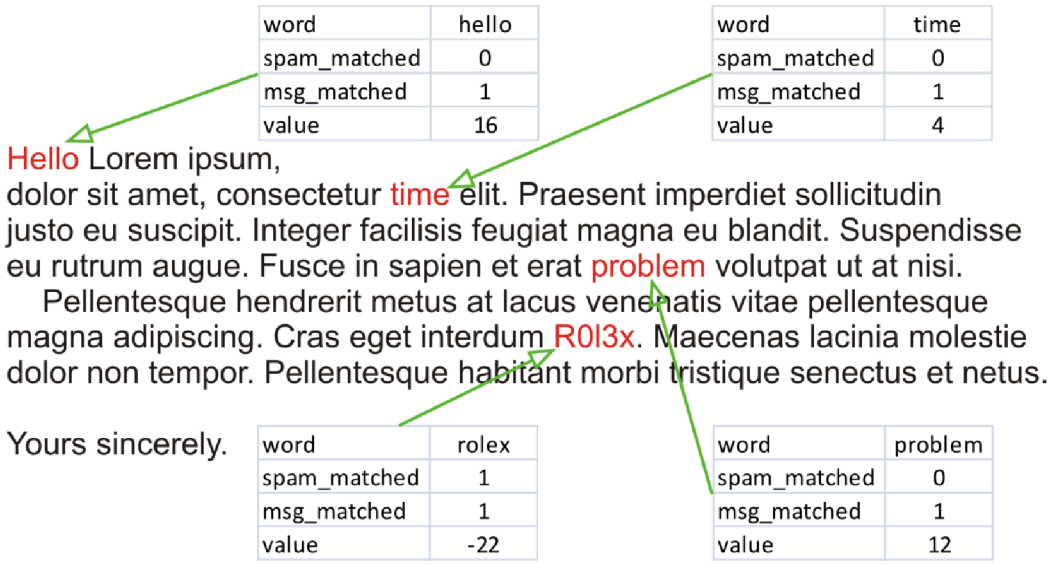
Pro fázi testování se z vybraného korpusu vyberou soubory ham a spam, které nejsou použity ve fázi učení a uloží se do složky *TEST*. Zpracování probíhá sériově, kdy se bere jeden email za druhým a jde do předzpracování, které je podobné jako ve fázi učení, tedy extrakce těla zprávy => odstranění HTML značek => převedení na malá písmena => extrakce slov.

Nadále probíhá vyhledávání lymfocytů, které odpovídají extrahovaným slovům ve stromové struktuře (vše v operační paměti, žádný dotaz do databáze). Výsledkem vyhledávání je seznam úspěšně navázaných lymfocytů na email, ze kterých je vypočítáno MyScore pomocí rovnice č.18.

$$MyScore = \frac{\sum_{success\_lymphocytes} \frac{spam\_matched}{msg\_matched} \cdot \log_2|value|}{\sum_{success\_lymphocytes} \frac{spam\_matched}{msg\_matched} \cdot \log_2|value| + \sum_{success\_lymphocytes} 1 - \frac{spam\_matched}{msg\_matched} \cdot \log_2|value|} \quad (18)$$

Rovnice (18): Výpočet MyScore z úspěšně navázaných lymfocytů.

Sumace je prováděna pro úspěšné lymfocyty, které rozpoznaly příslušné slovo/unigram ve zprávě/emailu. Váha $\log_2|\texttt{value}|$ zvýrazňuje vliv síly lymfocytu reprezentované hodnotou `value`. Pokud se zaměříme na algoritmus č.4, tak si všimneme, že `msg_match` je konstanta nastavená na hodnotu 1. A `spam_match` je proměnná, která nabývá hodnot pouze 0, nebo 1 v závislosti na hodnotě `value`. Uvažujme vytvořené lymfocyty z tabulky č.5 a ukažme si tělo emailu na následujícím obrázku č.23, který dostane náš systém na zpracování (text vygenerován pomocí www.lipsum.com).



Obrázek č.23: Tělo testovaného emailu a navázání lymfocytů.

Na obrázku č.23 můžeme vidět lymfocyty, které se úspěšně navázaly na email. Když máme množinu úspěšně navázaných lymfocytů, může spočítat MyScore.

$$MyScore = \frac{\sum_{success\_lymphocytes} \frac{spam\_matched}{msg\_matched} \cdot \log_2|value|}{\sum_{success\_lymphocytes} \frac{spam\_matched}{msg\_matched} \cdot \log_2|value| + \sum_{success\_lymphocytes} 1 - \frac{spam\_matched}{msg\_matched} \cdot \log_2|value|} \quad (19)$$

$$MyScore = \frac{\frac{0}{1} \cdot \log_2|16| + \frac{0}{1} \cdot \log_2|4| + \frac{0}{1} \cdot \log_2|12| + \frac{1}{1} \cdot \log_2|-22|}{\frac{0}{1} \cdot \log_2|16| + \frac{0}{1} \cdot \log_2|4| + \frac{0}{1} \cdot \log_2|12| + \frac{1}{1} \cdot \log_2|-22| + \left(\left(1 - \frac{0}{1}\right) \cdot \log_2|16| + \left(1 - \frac{0}{1}\right) \cdot \log_2|4| + \left(1 - \frac{0}{1}\right) \cdot \log_2|12| + \left(1 - \frac{1}{1}\right) \cdot \log_2|-22|\right)} \quad (20)$$

$$MyScore = \frac{\log_2|-22|}{\log_2|-22| + \log_2|16| + \log_2|4| + \log_2|12|} \quad (21)$$

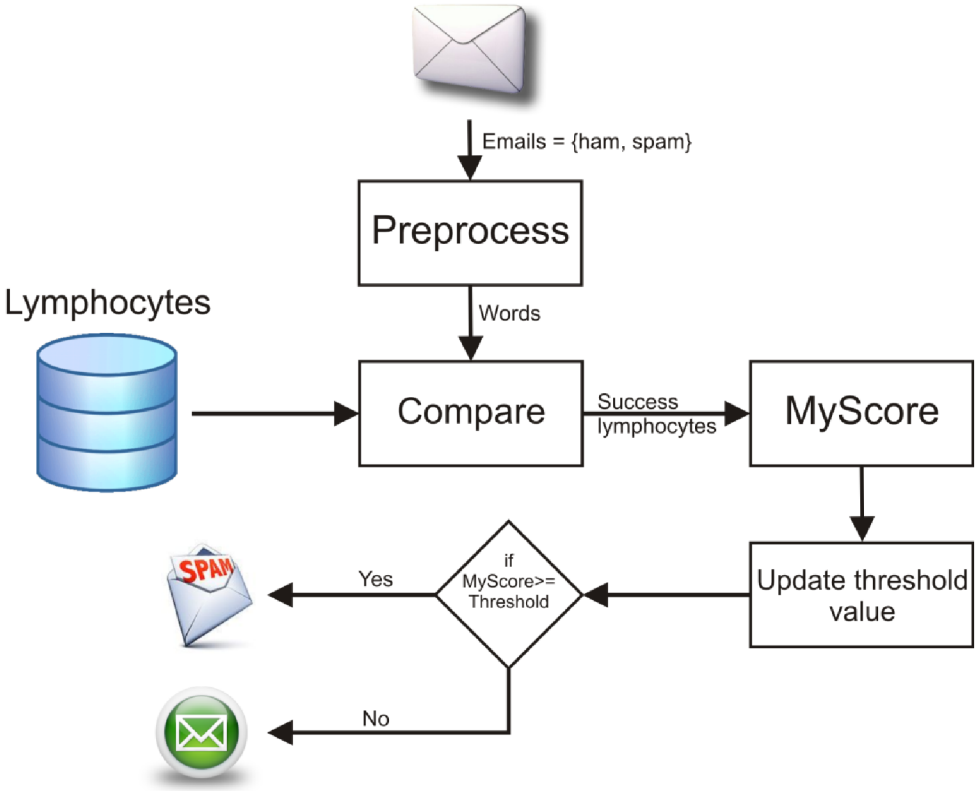
$$MyScore = \frac{4{,}459}{14.044} \quad (22)$$

$$MyScore = 0{,}3175 \quad (23)$$

Výpočet: Výpočet MyScore úspěšných lymfocytů na testovacím emailu.

Threshold, neboli práh se u tohoto systému nestanovuje uživatelem, ale naopak je vypočten systémem. Výpočet probíhá tak, že se vezme určitý počet emailů z trénovací množiny (optimálně 50x ham, 50x spam) a tyto emaily se nechají otestovat systémem s přednastaveným prahem.

Přednastavený práh se mění systematicky od 0,06 a krokově se inkrementuje o 0,03 až do hodnoty 0,75. Pro finální testování zpráv se použije se práh, při kterém byla úspěšnost klasifikace největší, tedy nejnižší počet zpráv s *FalsePositive* a *FalseNegative* klasifikací. Výpočet práhu se pak adaptuje následně opět po zpracování určitého počtu emailů (nastaveno 2500), v obrázku č.24 se jedná o blok „update threshold value“. Výhodou automatického výpočtu práhu systémem je, že se uživatel nemusí o nic starat, tedy systém je v tomto ohledu plně automní. Další výhodou je, že tato úloha je plně paralelizovatelná a je tedy možné využít plný výkon procesoru. Na obrázkuč.24 je graficky znázorněno blokové schéma klasifikace emailu.



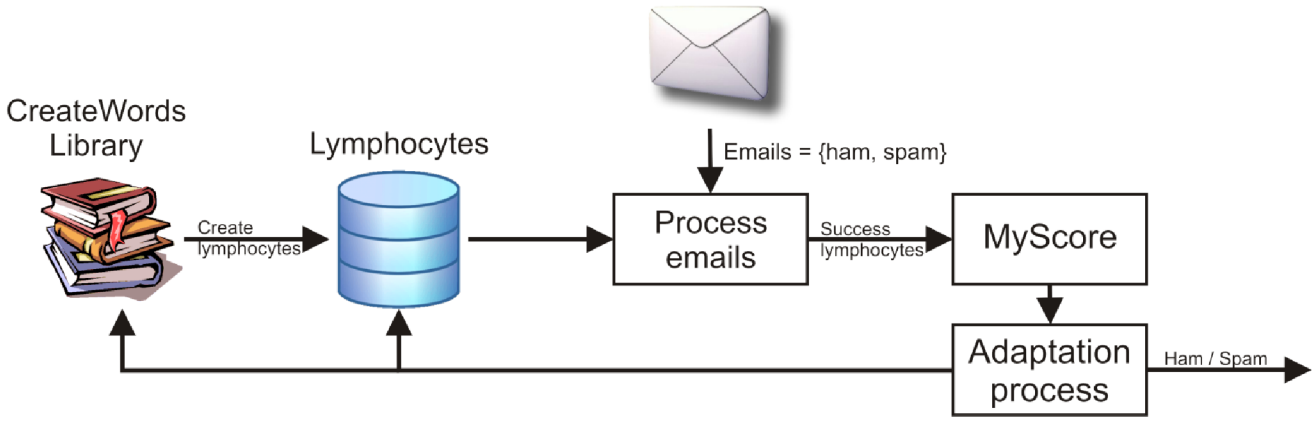
Obrázek č.24: Blokové schéma zpracování emailů

V systému je implementována ještě ***druhá úroveň klasifikace***. Ta se zabývá testováním *předmětu* emailu (angl. subject) a je spuštěna, když je vypočtená hodnota MyScore velice blízká vypočtenému prahu (nastaveno Threshold +- 0,05). V druhé úrovni se testuje:

- Je předmět psán velkými písmeny?
- Obsahuje předmět znaky jako jsou '$' a '%'?
- Obsahuje předmět typická spamová slova : penis, oral, sex, ...
- Jsou v emailu použity HTML tagy?

Na každou otázku, na kterou je odpověd *ANO*, tak je MyScore inkrementováno o určitou bodovou srážku, pokud je odpověď *NE*, tak u některých otázek dojde k dekrementování MyScore.

Na následujícím obrázku č.25 je zobrazeno blokové schéma celého systému. Ve schématu se vyskytuje blok „Adaptation process". Na jednoduchém příkladu si popišme co se něm provádí. Z výpočtu rovnice (23) víme, že `MyScore` = 0,3175 a z automatického výpočtu práhu vyplynulo, že `práh = 0,6`. Jelikož `MyScore` je menší jak prahová hodnota, je ukázkový email, klasifikován jako ham. Klasifikováním / identifikováním emailu, ale proces zpracování nekončí. Dále se vezmou všechna slova ze zpracovávaného emailu a k existujícím slovům (samozřejmě i lymfocytům) ve slovníku se k ohodnocení `value` přičte +1 (viz. druhá fáze učení), respektive dojde přičtení hodnoty `value`. Ostatní slova testované zprávy, která momentálně ve slovníku nejsou, se do něj vloží a vytvoří se nové ham lymfocyty (pozn. `value = 1`). Přirozeně, kdyby byl email klasifikován jako spam, je proces analogický, ale s hodnotou -1.



Obrázek č.25: Vývojový diagram celého systému.

```
Algorithm: Main
    INPUT: ham files, spam files, test files
    //CreateWords Library
    dictionary = Merge words(Preprocess HAM, Preprocess SPAM)

    //Create lymphocytes
    lymphocytes = CreateLymphocytes(dictionary)

    //calculate optimal threshold, chapter 7.4 Threshold
    threshold = FindOptimalTreshold()

    //Preprocess testing emails
    mails = Preprocess_test_files() //the same like prep.HAM & SPAM

    //process emails
    foreach mail in mails do:
        succ_Lyms[] = {}
        foreach word in mail.words do:
            //match word to lymphocyte
            //method find lymphocyte for the word, image no.22,also
        used inverse haxor funcion
            lym = lymphocytes.find(word)
            //if match is success
            if lym != null then
                push lym into succ_Lyms
            end if
```

```
        end foreach
        //now we have set of success lymphocytes

        //calculate MyScore, equation no. 18
        MyScore = CalculateMyScore(succ_Lyms)

        if (threshold+0,05>MyScore)&&(threshold-0,05<MyScore) then
            //Second phase of classification, chapter 7.4 Second
            //Level of classfication
            MyScore = SecondClassification()
        end if

        //feedback
        if MyScore >= threshold then
            //email is spam
            //if word exists in dictionary, then update value by
            //-1, else push word into dictionary with value -1
            dictionary.Update(mail.words, -1)
            //the same for lymphocytes
            lymphocytes.Update(succ_lyms, -1)
        else
            dictionary.Update(mail.words, 1)
            lymphocytes.Update(succ_lyms, 1)
        end if

        //write result on screen and update stats
        ShowResult()
    end foreach

    ShowStatistics()
    //end of algorithm
```

Algoritmus č.6: Algoritmus „celého“ cystému

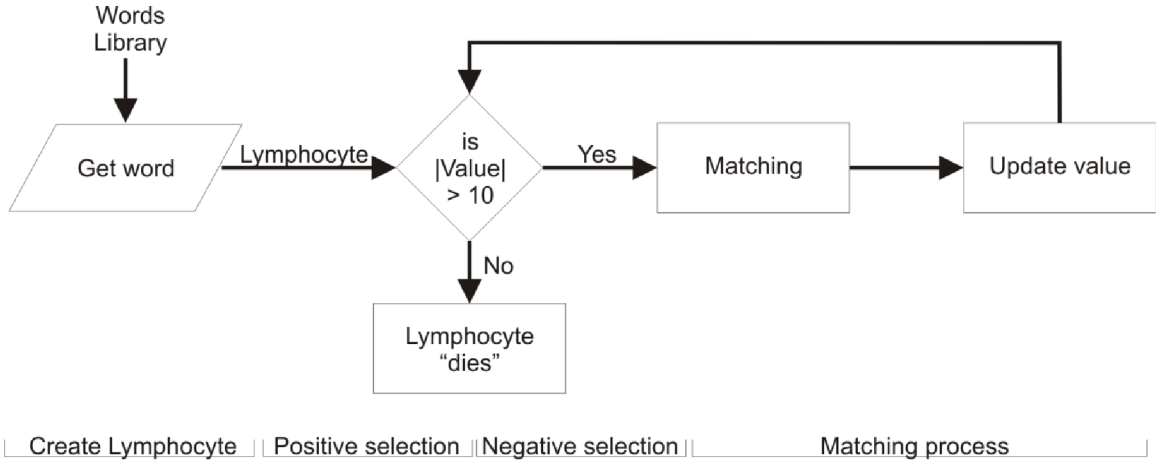
V algoritmu č.6 jsme si zjednodušeně ukázali pomocí pseudokódu, jak systém pracuje. Nejsou v něm zachyceny některé algoritmy, jako jsou detaily života lymfocytu (pozn. musí mít potřebnou `value`, aby mohl „žít“ viz. následující podkapitola). Dále práce s tvorbou statistik a vytváření souborů s histogramy (viz. kapitola 8). Nepovažuji také za nezbytné popisovat algoritmicky práci se stromovou strukturou.

7.5 Životnost lymfocytů

Každý umělý imunitní systém by měl mít stanovenou délku života každého lymfocytu. Někteří stanovují délku lymfocytu podle časového razítka vytvoření a délka života je napevno nastavena třeba na jeden den / týden. Nebo je možné mít vyhrazenou vnitřní proměnnou, která se po navázání lymfocytu na email dekrementuje a až dojde na 0, tak se otestuje efektivita lymfocytu v systému na základě efektivity vyhledat spam (vypočte se z spam_matched / msg_matched, existují pouze spam lymfocyty).

V našem případě si představme, že při spuštění systému se z našeho slovníku pro každé ohodnocené slovo vytvoří lymfocyt. Na tyto vytvořené lymfocyty se aplikuje pozitivní selekce a to tak, že se odstraní ty lymfocyty, jejichž `value` nemá hodnotu větší jak 10, nebo menší jak -10, respektive `value` leží v intervalu < -10, 10>. Tedy musí v první fázi učení, mít alespoň o šest výskytů více oproti druhé množině. Příklad, slovo se nám vyskytuje 9x v ham souborech a 3x ve spam souborech, tedy má ohodnocení +12 (pozn. jedná se o fázi učení, tedy každý výskyt slova je ohodnocen +-2 k `value`), tedy nebude eliminován, kdyby měl jenom +10, tak eliminován bude.

Za běhu systému, když se dostane `value` do intervalu <-10, 10> aplikuje se na lymfocyt negativní selekce, tedy odstraní se neefektivní lymfocyty, neefektivní v tom smyslu, že je velice malá pravděpodobnost, že budou v budoucnu využity. Oproti tomu, když slovo dostane ohodnocení do intervalu (-infinity, -10) nebo (10, infinity), bude vytvořen lymfocyt s `value` rovné ohodocení slova. Životní cyklus je znázorněn na obrázkuč.26.



Obrázek č.26: Životní cyklus lymfocytu.

Dále se nabízí otázka, zda má cenu kontrolovat smysluplnost slov, tedy porovnávat je vzhledem k nějakému anglickému/americkému slovníku (například Debian wamerican-insane 6.3). Když se podíváme opět na obrázek č.23, tak kromě těch pár slov, na které se navázaly lymfocyty, tak ostatní slova jsou nesmyslná (rutrum, augue, ..), a tedy jestli se vyplatí ukládat je do slovníku v systému?

Já jsem toho názoru, že uchovávat slova s nesmyslným významem má smysl, tím pádem je použití externího slovníku zbytečné. Protože, když si představím činnost spammera, který se snaží všemožnými způsoby zmást antispamové programy, například rozdělováním slov *vi ag ra*, tak ani jedna část nedává smysl a tedy by nebyli tyto části (*tokeny*) uloženy do mého slovníku. V případě False positive a zpětné vazby od uživatele by bylo možné natrénovat tyto fragmenty, aby byly klasifikovány za spamová slova. Někdy se fragmentů a slovům říká obecně ***tokeny***. Dále pokud má spamer k dispozici síť počítačů (botnet), ze kterých má možnost odesílat spam, tak je možné, že dostane příjemce, jeden a ten samý email z více emailových adres, tedy je možné opětovně použít tyto fragmenty.

8 Testování

Všechny souborů emailů, se kterými jsem pracoval, jsou uloženy na internetu a jsou veřejně dostupné. Konkrétně se jedná o korpusy SpamAssassin a Ling:

- SpamAssassin – 9 349 emailů (6 951 ham, 2398 spam)
  `http://spamassassin.apache.org/publiccorpus`
- Ling – 2 887 (2 406 ham, 481 spam)
  `http://labs-repos.iit.demokritos.gr/skel/i-config/`
- TREC 2005 Public Spam Corpus – 92 189 (39 399 ham, 52 790 spam)
  `http://plg1.cs.uwaterloo.ca/~gvcormac/treccorpus/`

Korpus Ling je pro nás výhodný pro testování a porovnání vůči korpusu SpamAssassin hned z několika důvodů. Tím, že budeme používat trénovací množinu vytvořenou pouze se SpamAssassina a testovat soubory z jiného korpusu můžeme otestovat, zda je systém se schopen adaptovat na jiný druh zpráv. Změnou druhu zpráv můžeme simulovat i časovou závislost, tedy v jednom časovém období dostavám určitý typ zpráv (trénovací množina) a pak dojde třeba ke změně povolání / pracovní pozice a tím se změní i tématika korektních emailů, spamy zůstávají více méně stejné.

TREC 2005 Public Spam Corpus je určen ke srovnání s dnes běžně používanými programy/aplikacemi jako je Bogofilter, SpamAssassin, SpamBayes, SpamProbe a NaiveBayes. Abych byl schopen určit, zda klasifikace proběhla správně, či nikoliv, označil jsem veškerou vyžádanou poštu koncovkou `.ham` a analogicky k nevyžádané poště koncovkou `.spam`. Při zpracování emailů v kapitole 7.4 samozřejmě přistupuji k emailu tak, že o něm nic nevím a až po klasifikování se podívám na zmíněnou koncovku. Díky tomu můžu rozlišovat 4 stavy:

- Ham – správně rozpoznán ham
- Spam – správně rozpoznán spam
- False Positive – spam byl nesprávně klasifikován jako ham
- False Negative – ham byl nesprávně klasifikován jako spam

Tedy HAM + False Negative = skutečný celkový počet ham souborů a SPAM + False Positive = skutečný celkový počet spam souborů. Uvedené stavy jsou znázorněny i v tabulce č.7.

	classified as legitimate mail by system	classified as spam by system
real legitime mail	Ham	FalseNegative
real spam	FalsePositive	Spam

Tabulka č.7: Jednotlivé stavy klasifikace. Zdroj: [Pei-yu]

[Pei-yu] Naměřené výsledky budou zhodnoceny podle následujících kritérií a *All* je zástupcem součtu všech 4 kategorií, tedy *All* = *Ham* + *Spam* + *FalsePositive* + *FalseNegative*:

a) Accuracy:

$$Accuracy = \frac{\text{Ham+Spam}}{\text{All}} * 100\% \quad (24)$$

b) Recall: Indikátor určuje schopnost systému rozpoznat Spam.

$$Recall = \frac{\text{Spam}}{\text{Spam+ FalsePositive}} * 100\% \quad (25)$$

c) Precision: Indikátor určuje schopnost systému rozpoznat Spam správně.

$$Precision = \frac{\text{Spam}}{\text{Spam+FalseNegative}} * 100\% \quad (26)$$

d) Miss Rate: Množství spamu, které nebylo identifikováno jako spam.

$$Miss\ Rate = \frac{\text{FalsePositive}}{\text{Spam+FalsePositive}} * 100\% \quad (27)$$

e) Error: Množství emailů, které byli nekorektně identifikovány

$$Error = \frac{\text{FalseNegative+FalsePositive}}{\text{All}} * 100\% \quad (28)$$

Během testování emailů se ukládají výsledky do souborů `histogramReal.csv` a `stats.csv`. Do prvně zmíněného se ukládají počty výskytů ham a spam emailů, podle vypočtené hodnoty MyScore při rozlišení 0,01. V druhém uvedeném se ukládají detailnější statistiky, podle toho jak systém zpracovával emaily a jak se měnila úspěšnost (Accuracy) systému v závislosti na výsledku klasifikace. Zaznemánavají se výsledky pro každý zpracovaný email.

Pro trénování jsem zvolil korpus SpamAssassin a vytvořil jsem 3 trénovací množiny, které jsou složeny z 1000 ham a 1000 spam emailů (dále jen 1000:1000), 500 ham a 500 spam emailů (dále jen 500:500) a 250 ham a 250 spam emailů (dále jen 250:250). Korpus SpamAssassin má rozdělené emaily podle složitosti identifikování, tedy easy ham, hard ham a spam. Trénovací množina 1000:1000 byla vytvořena pro spam množinu náhodným namícháním a ham množina obsahuje cca 800 easy ham a 200 hard ham emailů. Další trénovací množiny o redukované velikosti, byly vytvořeny náhodným namícháním z množiny 1000:1000.

Připravil jsem 7 testů:

1) SpamAssassin 1000:1000 – testování úspěšnosti naimplementovaného systému na doporučeném množství trénovacích emailů (SpamAssassin doporučuje velikost trénovací množiny 1000:1000).
2) SpamAssassin 500:500 – testování úspěšnosti systému na redukované trénovací množině. (Porovnání úspěšnosti vůči SpamAssassin 1000:1000)
3) SpamAssassin 250:250 - testování úspěšnosti systému na redukované trénovací množině, cílem tohoto testu je zjistit, zda je systém schopen dospět ke stejným úspěšnostem, jako v předchozích testech.
4) Ling 1000:1000 – testování úspěšnosti systému, kdy trénovací množina je SpamAssassin 1000:1000 a testovací množina je korpus Ling. Cílem testu je zjistit, zda je systém schopen se adaptovat na nový druh zpráv (nový druh myšleno ve smyslu obsahu emailu). Jedná se

již o náročnější test, protože mezi korpusem SpamAssassin a Ling není definována korelace.

5) Ling 500:500 – testování úspěšnosti systému, kdy je redukovaná trénovací množina SpamAssassin 500:500. Opět cílem testu je, zda je systém schopen se adaptovat a dále zjistit, zda dosáhne stejných výsledků oproti testu Ling 1000:1000.
6) Ling 500:500 – testování úspěšnosti systému, kdy je redukovaná trénovací množina SpamAssassin 250:250. Opět cílem testu je, zda je systém schopen se adaptovat a dále zjistit, zda dosáhne stejných výsledků oproti 4. a 5. testu.
7) TREC 1000:1000 – trénovací množina byla použita také z korpusu TREC. Testování bylo prováděno na 10 000 emailech, kde 57% zaujímal spam. Dobré u tohoto korpusu je dále, že byly emaily seřazeny podle času, tedy podle toho, jak přicházely. Proto první (nejstarší) emaily byly vybrány jako trénovací.

V každém testu se provádí 10 běhů systému, to znamená, že se spustí systém, nechá se natrénovat (fáze učení) na připravené trénovací množině emailů (SpamAssassin 1000:1000, apod.). Poté se nechají systémem otestovat testované emaily (SpamAssassin nebo Ling). Až se dokončí testování emailů program vypíše všechny výsledky v přehledné formě (viz. příloha č.2). 2. testovací běh znovu otestuje testované emaily a takto se pokračuje až do 10. běhu. Výsledky ze všech deseti běhů se vynesou do tabulek. V tabulkách jsou červeně označeny nejhorší naměřené výsledky a zeleně nejlepší naměřené výsledky.

Následně se vytvoří 4 grafy, které znázorňují obálku histogramu (histogram počtu výskytů při jednotlivých hodnotách MyScore převeden do spojité funkce). Na grafech je zachycena obálka histogramu po prvním, druhém, pátem a desátém běhu systému.

Dále se vytvoří dva grafy zobrazující procentuální úspěšnost systému v závislosti na počtu zpracovaných emailů. V grafu je vyznačena procentuální úspěšnost (Accuracy), procentuální výskyt počtu FalsePositive klasifikovaných emailů a procentuální výskyt počtu FalseNegative klasifikovaných emailů. Grafy jsou vytvořeny z výsledků prvního a desátého běhu systému. Na konci každého testu je diskuze nad naměřenými výsledky. Co z grafu vyplývá a upozornění na zajímavé výsledky.

Testování probíhalo na mém osobním notebooku HP EliteBook 8530p.

Testovací konfigurace:

CPU : Intel Core 2 Duo T9400 (2,53 GHz, 6 MB L2 cache, 1066 MHz FSB)
RAM: 2x 2 GB 800 MHz DDR2 SDRAM, Dual Channel
GPU: ATI Mobility Radeon HD 3650, 256 MB
HDD: 250 GB, 7200 ot./min
OS: Windows Vista, 32bit
Index: 5,3 (Index uživatelských zkušeností se systémem Windows)

9 Dosažené výsledky

9.1 SpamAssassin 1000:1000

Z fáze učení slovník obsahuje 37444 ohodnocených slov, z těchto slov se vytvořilo 6564 ham lymfocytů a 8351 spam lymfocytů. Velikost slovníku po zpracování emailů je 82986 slov.

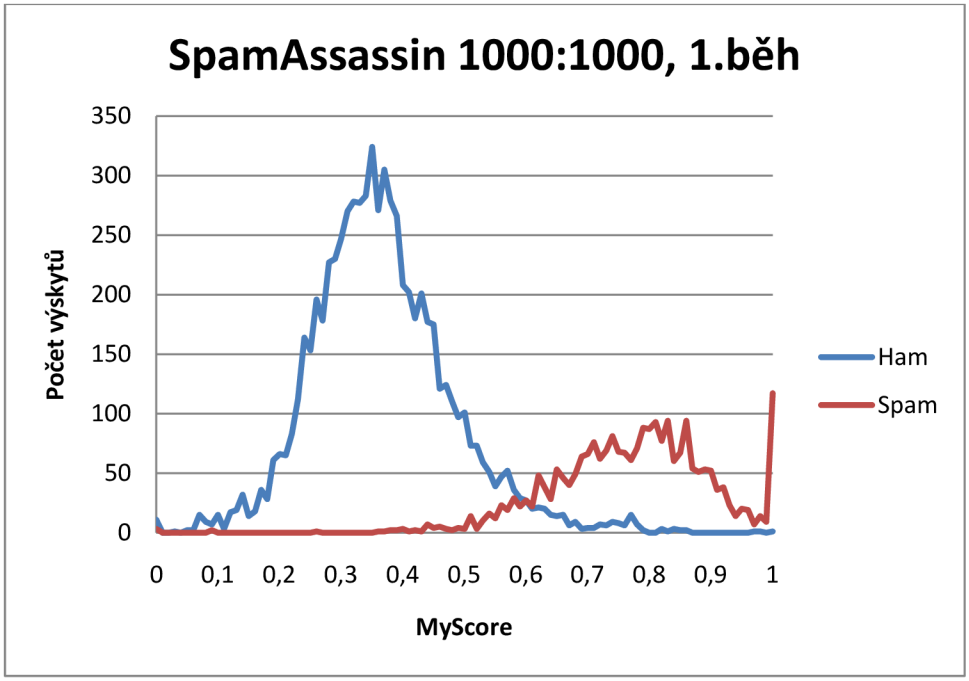
Běh	1.	2.	3.	4.	5.
Ham	6689	6709	6729	6449	6743
Spam	2244	2283	2272	2320	2266
FalsePositive	154	115	126	78	132
FalseNegative	262	242	222	502	208
Accuracy (%)	95,550	96,181	96,278	**93,796**	**96,363**
Recall (%)	**93,578**	95,204	94,746	**96,747**	94,495
Precision (%)	89,545	90,416	91,099	**82,211**	91,593
Miss Rate (%)	6,422	4,796	5,254	**3,253**	5,505
Error (%)	4,450	3,819	3,722	**6,204**	**3,637**
Ham lymfocytů (na konci běhu)	17891	27914	39720	39773	40456
Spam lymfocytů (na konci běhu)	18546	20445	23723	24169	24420
Čas zpracovaní/email (ms)	6,40	6,84	6,76	6,87	7,44

Tabulka č.8: SpamAssassin 1000:1000, běh 1.-5.

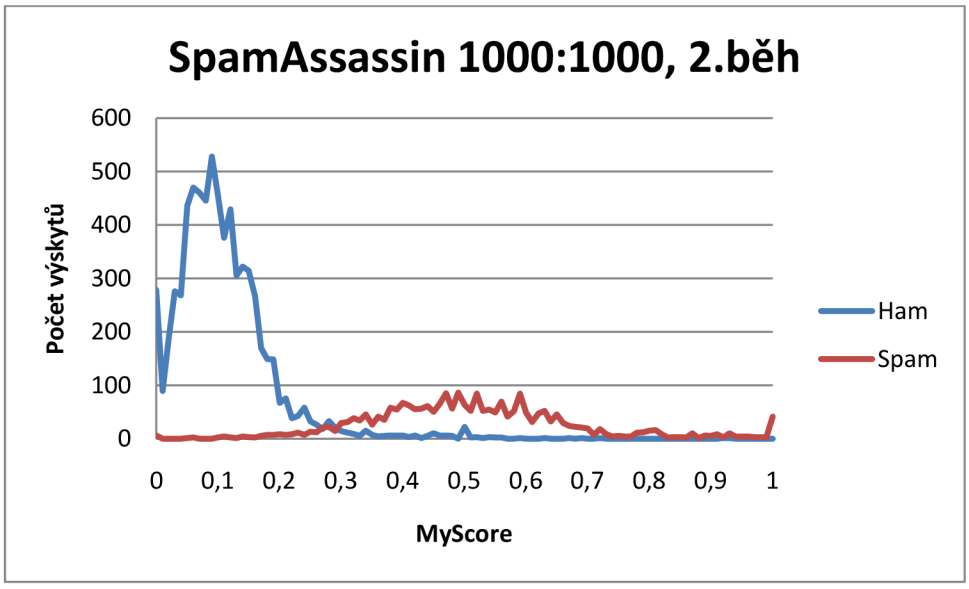
Běh	6.	7.	8.	9.	10.
Ham	6695	6742	6745	6750	6756
Spam	2289	2254	2247	2233	2230
FalsePositive	109	144	151	165	168
FalseNegative	256	209	206	201	195
Accuracy (%)	96,096	96,224	96,181	96,119	96,117
Recall (%)	95,455	93,995	93,703	93,119	92,994
Precision (%)	89,941	91,514	91,602	91,742	**91,959**
Miss Rate (%)	4,545	6,005	6,297	6,881	**7,006**
Error (%)	3,904	3,776	3,819	3,915	3,883
Ham lymfocytů (na konci běhu)	46430	46589	47560	47696	47871
Spam lymfocytů (na konci běhu)	33002	32930	34130	34110	34104
Čas zpracovaní/email (ms)	7,21	7,46	7,20	7,55	7,26

Tabulka č.9: SpamAssassin 1000:1000, běh 6.-10.

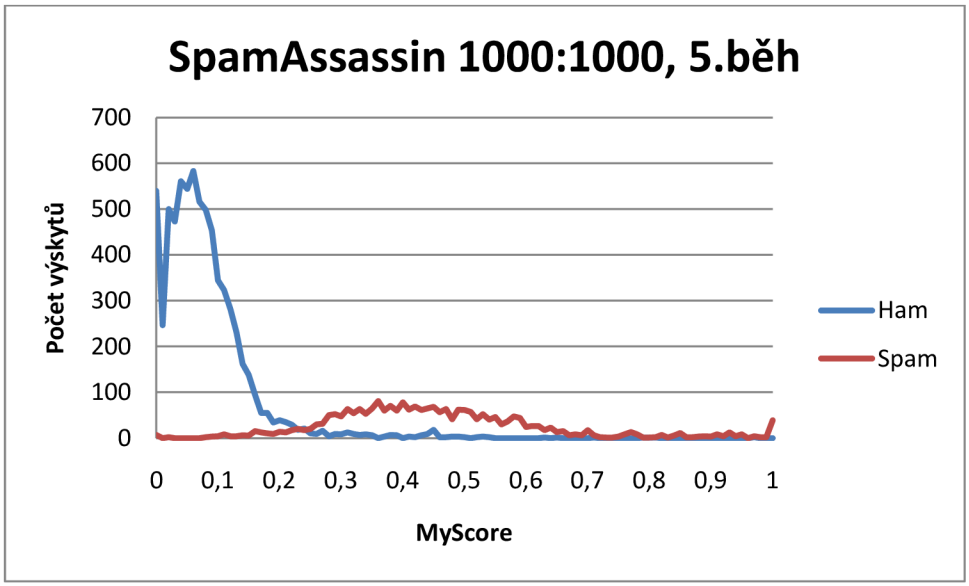
9.1.1 Obálky histogramů – SpamAssasin 1000:1000



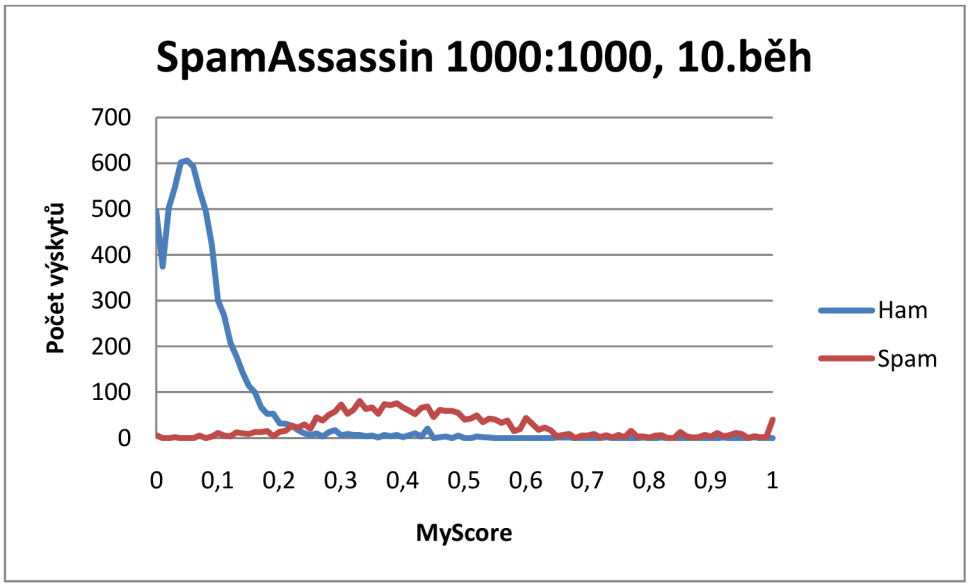
Obrázek č.27: Obálka histogramu – SpamAssassin 1000:1000, 1.běh



Obrázek č.28: Obálka histogramu – SpamAssassin 1000:1000, 2.běh

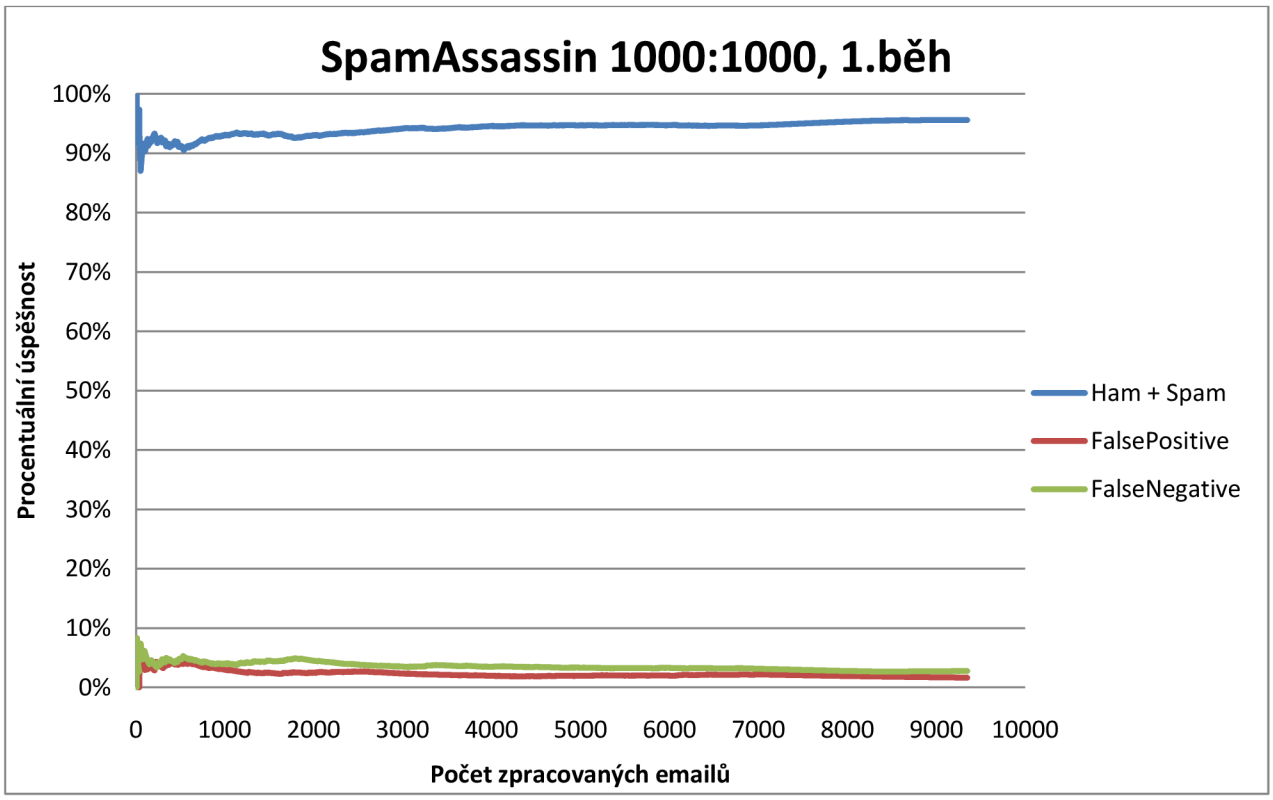


Obrázek č.29: Obálka histogramu – SpamAssassin 1000:1000, 5.běh

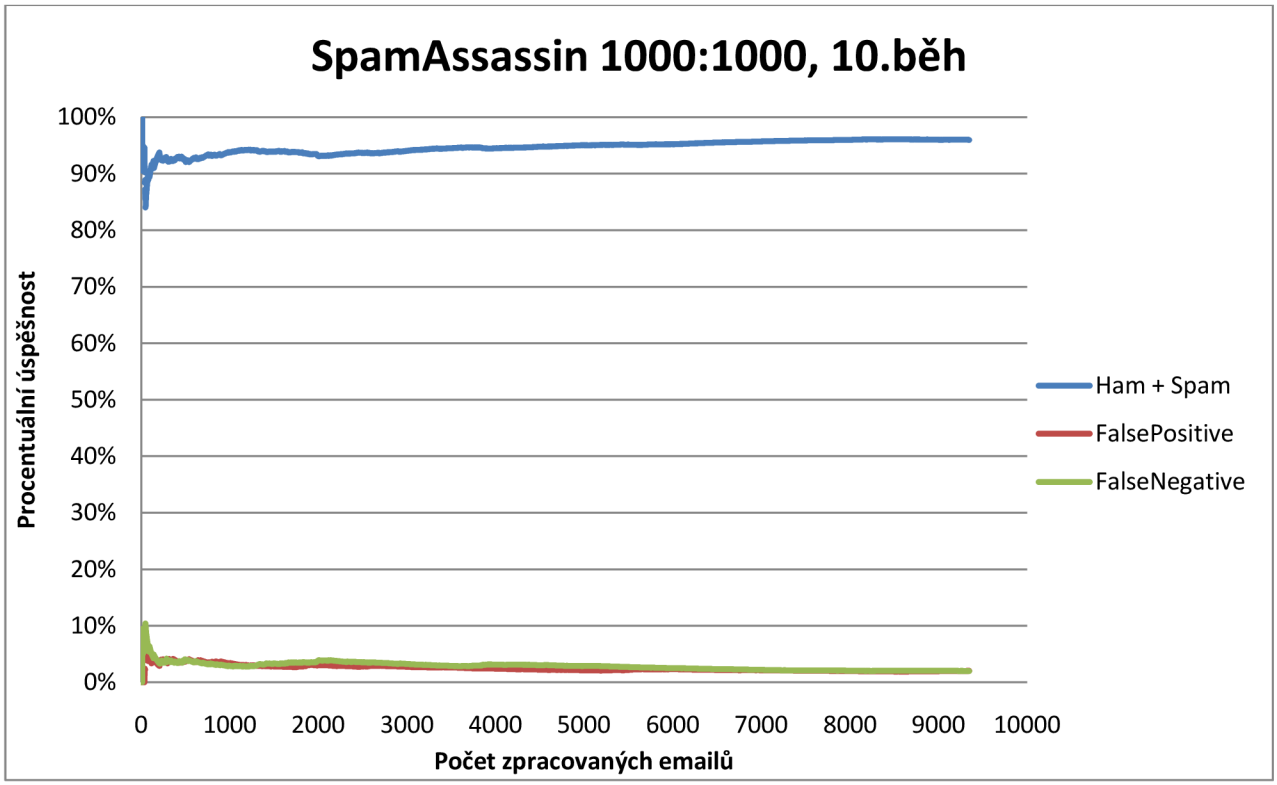


Obrázek č.30: Obálka histogramu – SpamAssassin 1000:1000, 10.běh

9.1.2 Procentuální úspěšnost – SpamAssassin 1000:1000



Obrázek č.31: Procentuální úspěšnost systému, SpamAssassin 1000:1000, 1.běh



Obrázek č.32: Procentuální úspěšnost systému, SpamAssassin 1000:1000, 10.běh

Diskuze – SpamAssassin 1000:1000

V obrázku č.27 můžeme vidět, že v prvním běhu systému, systém cca 120 emailů klasifikoval jako 100% spam (MyScore = 1). Na obrázku obr-sa1000-2] můžeme pozorovat jev, jak se mění práh z 0,6 na 0,3. Je to dáno tím, že si systém adaptoval nová slova a posun směrem k 0 je logický z důvodu, že přece jenom emaily obsahují obecně větší množství ham slov, jak spamových slov. Jelikož je úspěšnost (Accuracy) druhého běhu 96,818, lze usuzovat, že automatický výpočet práhu funguje správně. Pozitivní je, že když po 4. běhu byla úspěšnost pod 94%, dokázal systém v dalším běhu překročit opět 96% hranici.

Lze i vypozorovat „vymírání“ lymfocytů, na konci 5.běhu „žije“ v systému 40456 ham lymfocytů a na konci 6.běhu je 46430 ham lymfocytů. Ano, dá se namítnout, že ztráta 39 lymfocytů není nijak velká změna, ale musíme si uvědomit, že systém zpracoval v případě korpusu SpamAssassin 9349 emailů, tedy některé lymfocyty zahynuly a jiné se vytvořily. I spam lymfocyty uhynuly v 6.běhu oproti 5.

Z obrázků č.31 a č.32 můžeme vidět, že po celou dobu testování/zpracovávání emailů, byla úspěšnost nad 90% a procentuální počet výskytů FalsePositive a FalseNegative byl na stejné úrovní (cca 5%).

9.2 SpamAssassin 500:500

Z fáze učení slovník obsahuje 29045 ohodnocených slov, z těchto slov se vytvořilo 3397 ham lymfocytů a 3911 spam lymfocytů.

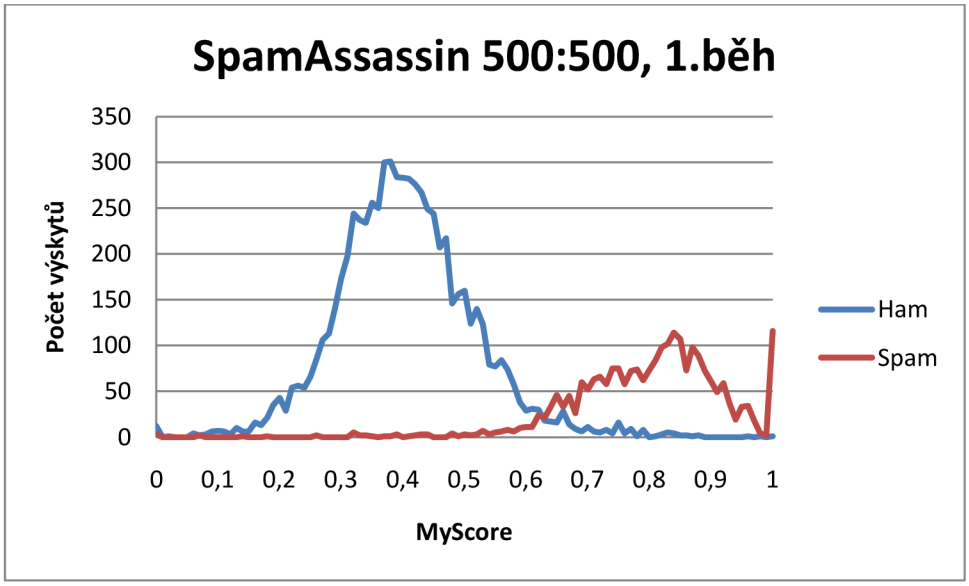
Běh	1.	2.	3.	4.	5.
Ham	6755	6765	6779	6747	6756
Spam	2263	2261	2240	2265	2264
FalsePositive	135	137	158	133	134
FalseNegative	196	186	172	204	195
Accuracy (%)	96,460	**96,545**	96,470	96,395	96,481
Recall (%)	94,370	94,287	**93,411**	94,454	94,412
Precision (%)	92,029	**92,399**	92,869	91,738	92,070
Miss Rate (%)	5,630	5,713	**6,589**	5,546	5,588
Error (%)	3,540	**3,455**	3,530	3,605	3,519
Ham lymfocytů (na konci běhu)	16247	26127	40227	40757	41091
Spam lymfocytů (na konci běhu)	8377	18216	23469	23719	23834
Čas zpracovaní/email (ms)	7,93	6,86	6,82	7,08	6,82

Tabulka č.10: SpamAssassin 500:500, běh 1.-5.

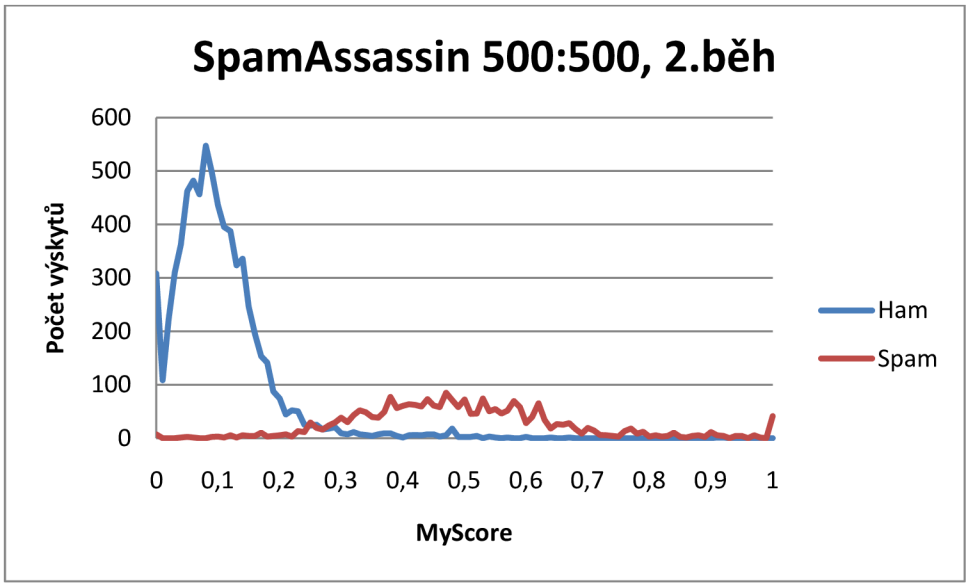
Běh	6.	7.	8.	9.	10.
Ham	6758	6751	6710	6052	6743
Spam	2257	2258	2281	2363	2263
FalsePositive	141	140	117	35	135
FalseNegative	193	199	241	899	208
Accuracy (%)	96,427	96,374	96,171	**90,010**	96,331
Recall (%)	94,120	94,162	95,121	**98,540**	94,370
Precision (%)	92,122	91,901	90,444	**72,440**	91,582
Miss Rate (%)	5,880	5,838	4,879	**1,460**	5,630
Error (%)	3,573	3,626	3,829	**9,990**	3,669
Ham lymfocytů (na konci běhu)	48110	48156	48266	47955	48178
Spam lymfocytů (na konci běhu)	32712	32726	33826	33999	33997
Čas zpracovaní/email (ms)	7,11	7,43	7,21	7,12	6,88

Tabulka č.11: SpamAssassin 500:500, běh 6.-10.

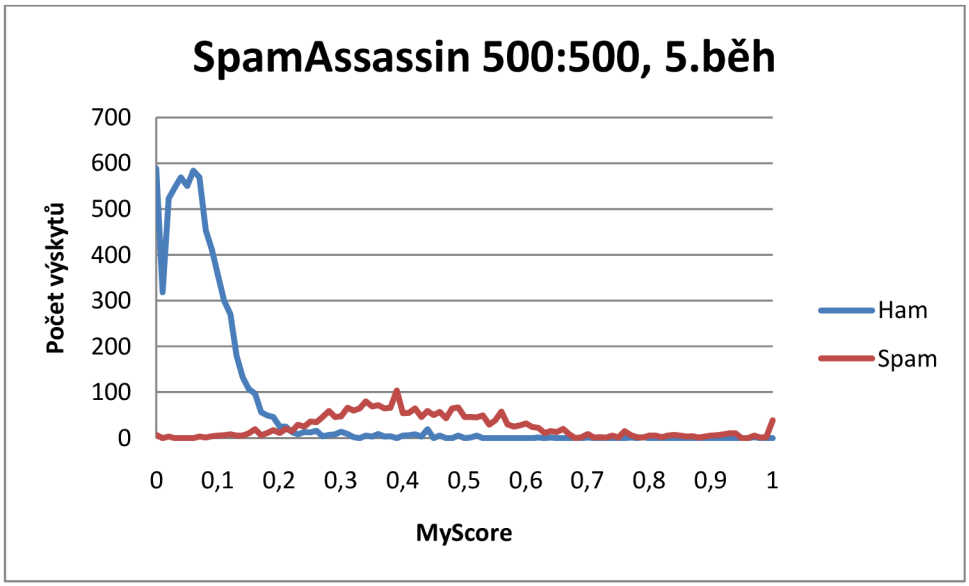
9.2.1 Obálky histogramů – SpamAssasin 500:500



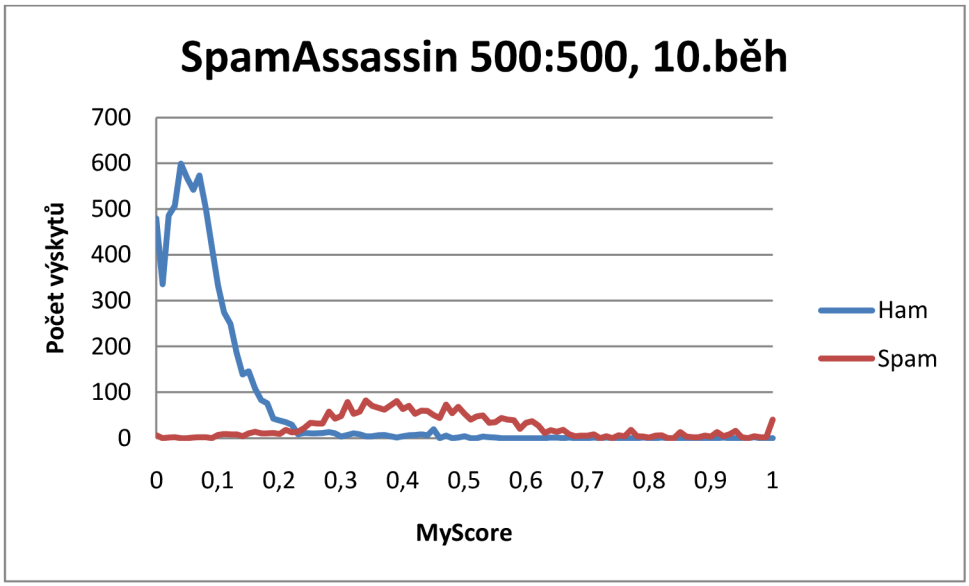
Obrázek č.33: Obálka histogramu - SpamAssassin 500:500, 1.běh



Obrázek č.34: Obálka histogramu - SpamAssassin 500:500, 2.běh

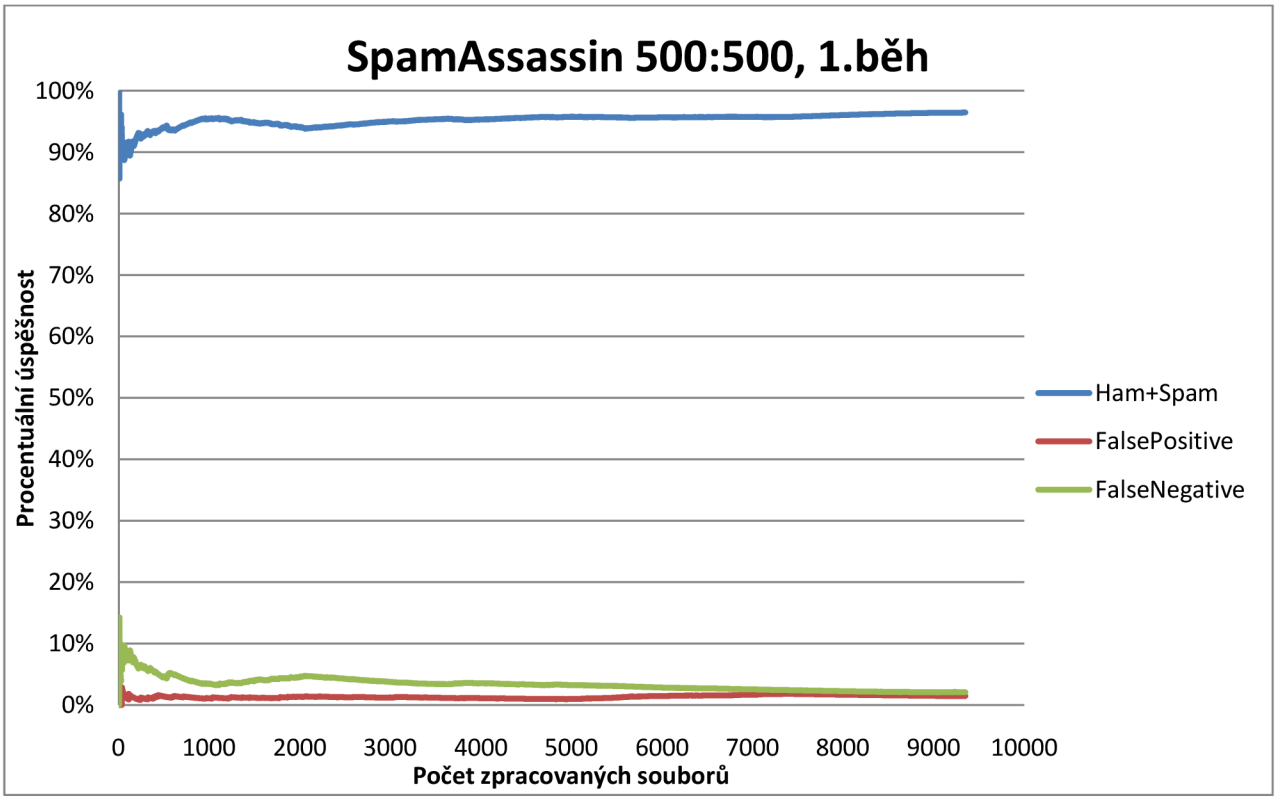


Obrázek č.35: Obálka histogramu - SpamAssassin 500:500, 5.běh

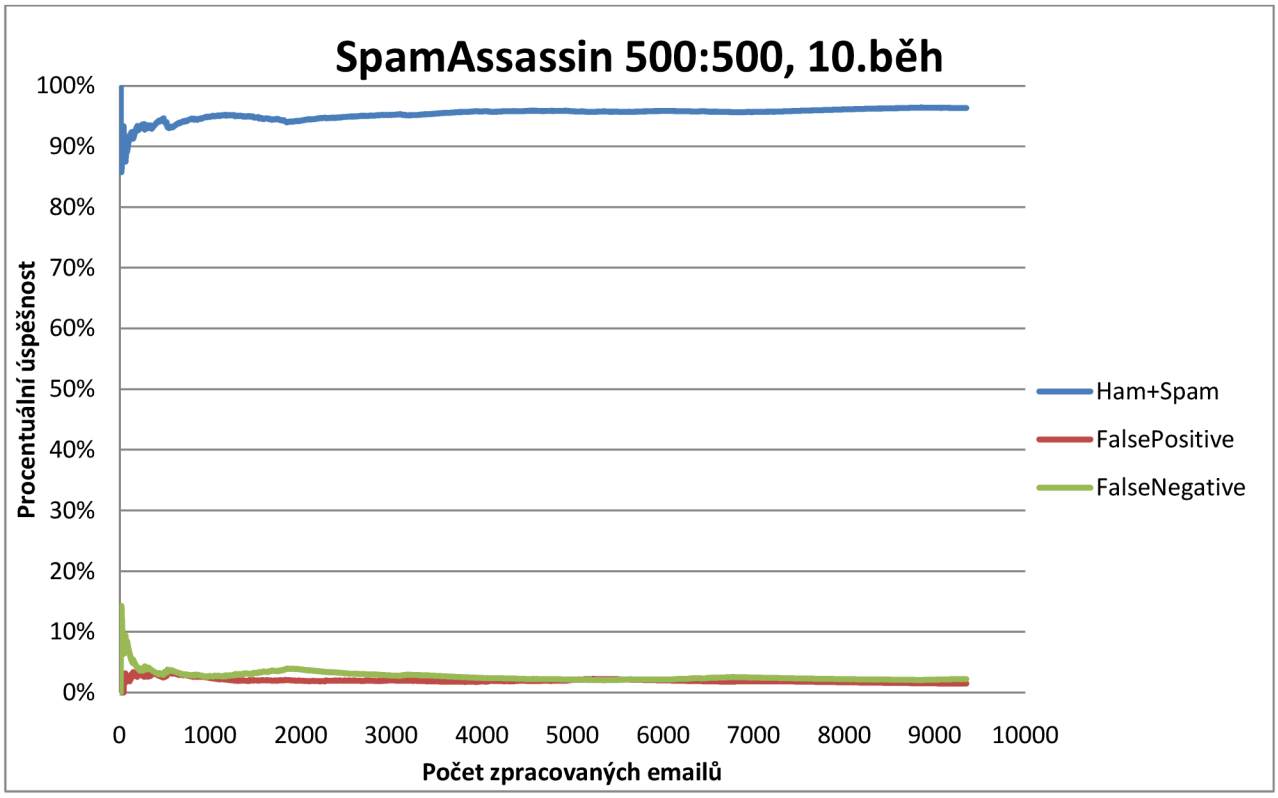


Obrázek č.36: Obálka histogramu - SpamAssassin 500:500, 10.běh

9.2.2 Procentuální úspěšnost - SpamAssassin 500:500



Obrázek č.37: Procentuální úspěšnost systému, SpamAssassin 500:500, 1.běh



Obrázek č.38: Procentuální úspěšnost systému, SpamAssassin 500:500, 10.běh

Diskuze – SpamAssassin 500:500

Výsledky v tabulkách č.10 a č.11 ukazují na zlepšení úspěšnosti o 0,2% oproti SpamAssassin 1000:1000, dokonce už od prvního běhu. Obálky histogramu se nějak výrazně neliší od předchozího testu. Když se podíváme na tvorbu lymfocytů, tak po třech bězích systému dojde k dramatickému nárůstu lymfocytů po třech bězích systému. U ham lymfocytů z 3397 na 40227 a u spam lymfocytů z 3911 na 23469.

V obrázku č.37 můžeme pozorovat změnu, kde je zachyceno větší množství FalseNegative výskytů vůči FalsePositive. V obrázku č.38 je patrné, že jsou kromě začátku, křivky FalsePositive a FalseNegative na stejné úrovni.

9.3 SpamAssassin – 250:250

Z fáze učení slovník obsahuje 25900 ohodnocených slov, z těchto slov se vytvořilo 2035 ham lymfocytů a 3391 spam lymfocytů.

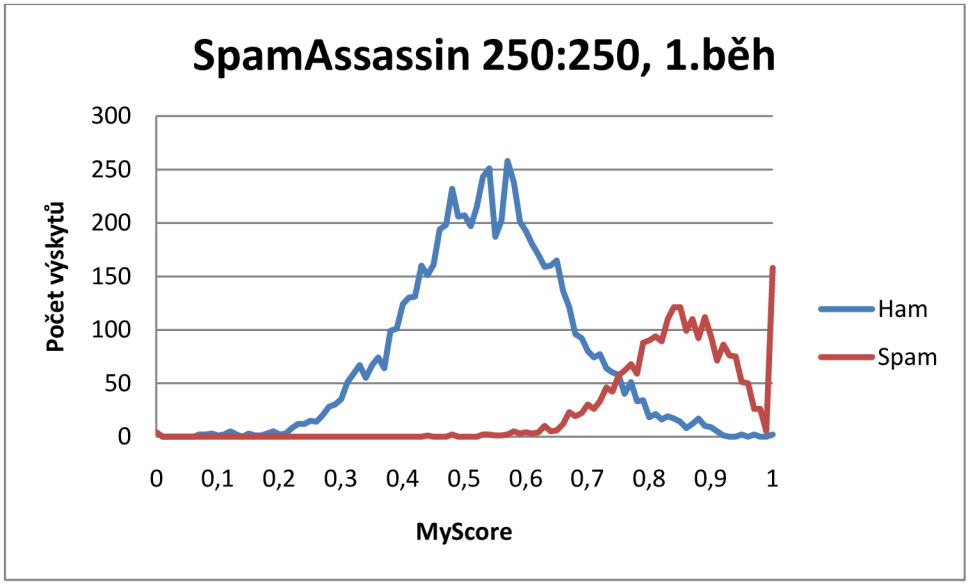
Běh	1.	2.	3.	4.	5.
Ham	6474	6765	6732	6741	6652
Spam	2205	2179	2211	2205	2271
FalsePositive	193	219	187	193	127
FalseNegative	477	186	219	210	299
Accuracy (%)	**92,833**	95,668	95,657	95,689	95,443
Recall (%)	91,952	**90,867**	92,202	91,952	**94,704**
Precision (%)	**82,215**	**92,135**	90,988	91,304	88,366
Miss Rate (%)	8,048	**9,133**	7,798	8,048	5,296
Error (%)	**7,167**	4,332	4,343	4,311	4,557
Ham lymfocytů (na konci běhu)	16766	26945	41592	42293	42370
Spam lymfocytů (na konci běhu)	7135	18215	22131	22377	22581
Čas zpracovaní/email (ms)	6,64	6,89	6,94	7,23	6,89

Tabulka č.12: SpamAssassin 250:250, běh 1.-5.

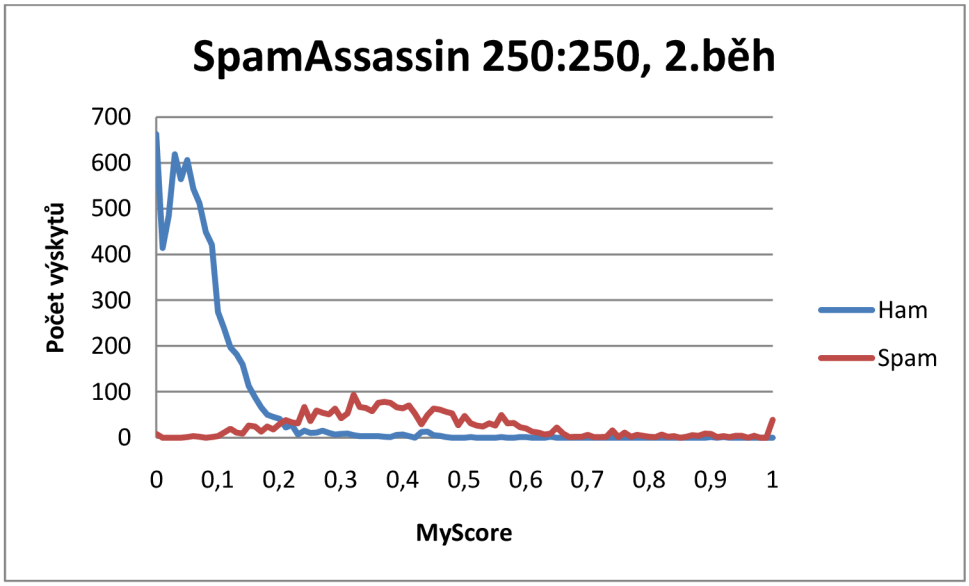
Běh	6.	7.	8.	9.	10.
Ham	6736	6579	6701	6699	6700
Spam	2211	2281	2226	2227	2227
FalsePositive	187	117	172	171	171
FalseNegative	214	372	250	252	251
Accuracy (%)	**95,710**	94,769	95,486	95,475	95,486
Recall (%)	92,202	95,121	92,827	92,869	92,869
Precision (%)	91,175	85,978	89,903	89,835	89,871
Miss Rate (%)	7,798	**4,879**	7,173	7,131	7,131
Error (%)	**4,290**	5,231	4,514	4,525	4,514
Ham lymfocytů (na konci běhu)	48820	48678	49145	49180	49394
Spam lymfocytů (na konci běhu)	30491	30600	32327	32345	32614
Čas zpracovaní/email (ms)	6,76	7,37	7,78	7,72	6,88

Tabulka č.13: SpamAssassin 250:250, běh 6.-10.

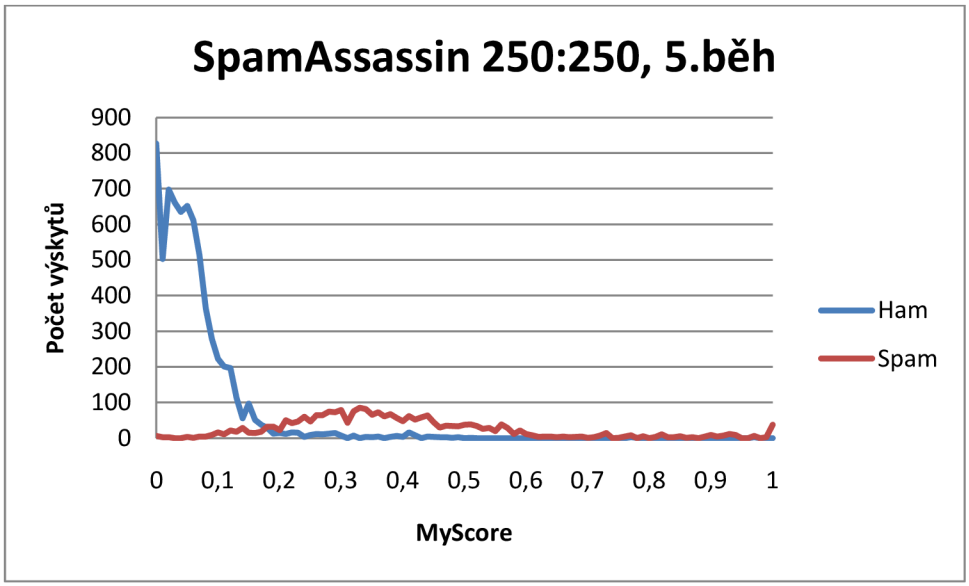
9.3.1 Obálky histogramů – SpamAssassin 250:250



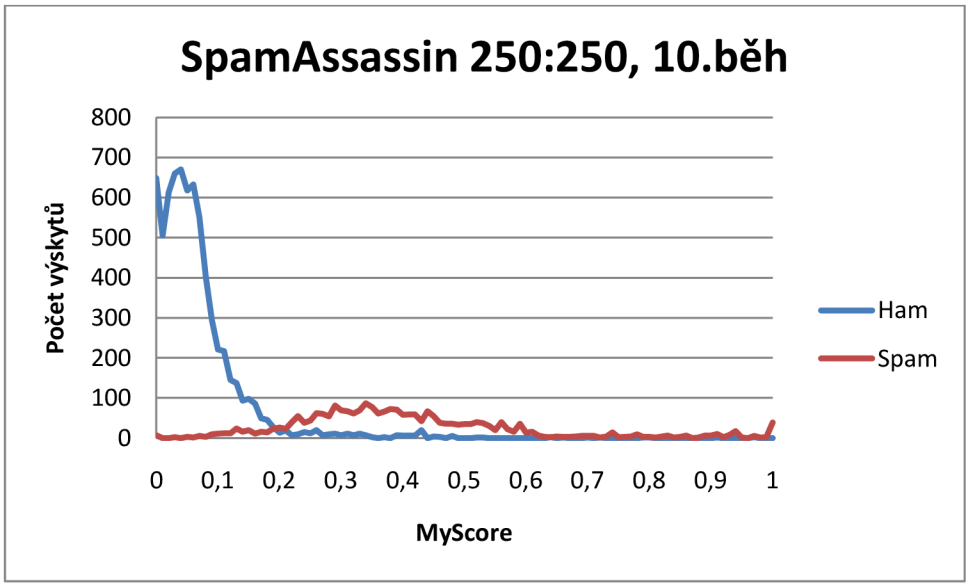
Obrázek č.39: Obálka histogramu - SpamAssassin 250:250, 1.běh



Obrázek č.40: Obálka histogramu - SpamAssassin 250:250, 2.běh

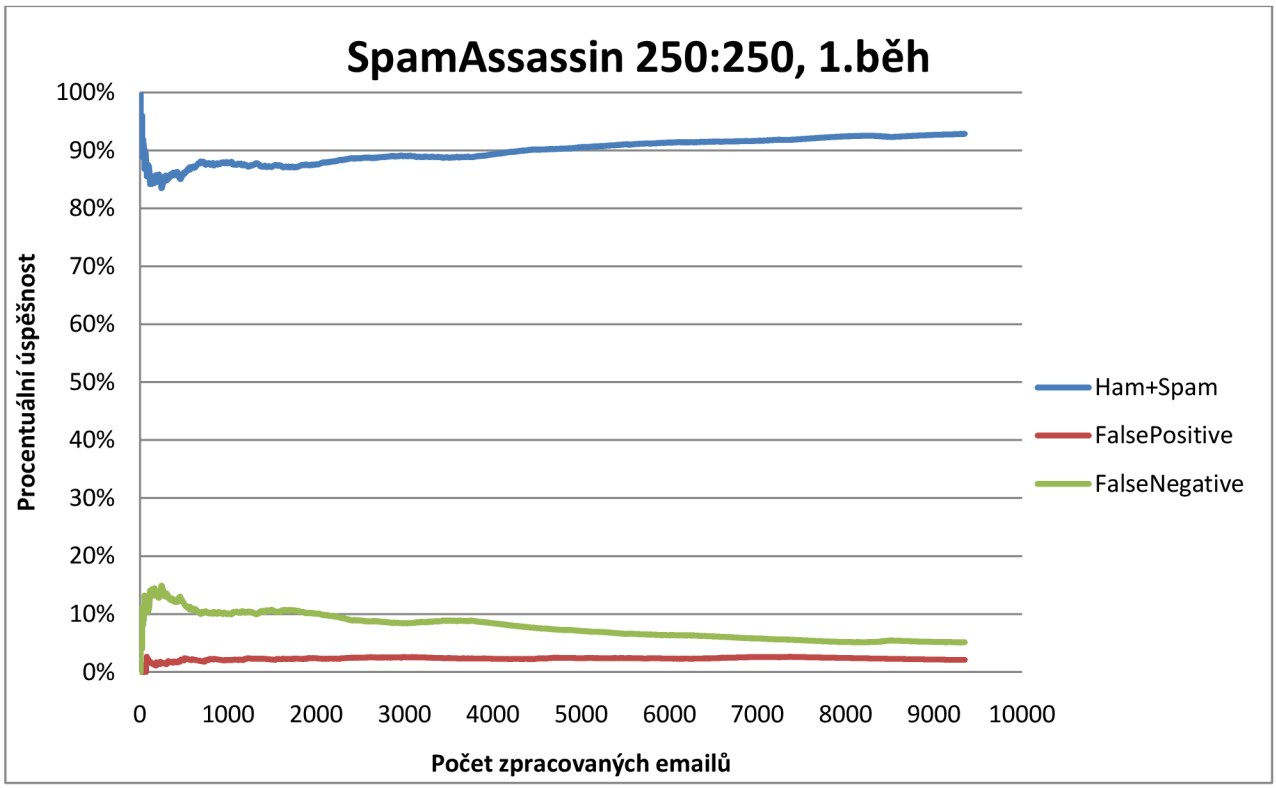


Obrázek č.41: Obálka histogramu - SpamAssassin 250:250, 5.běh

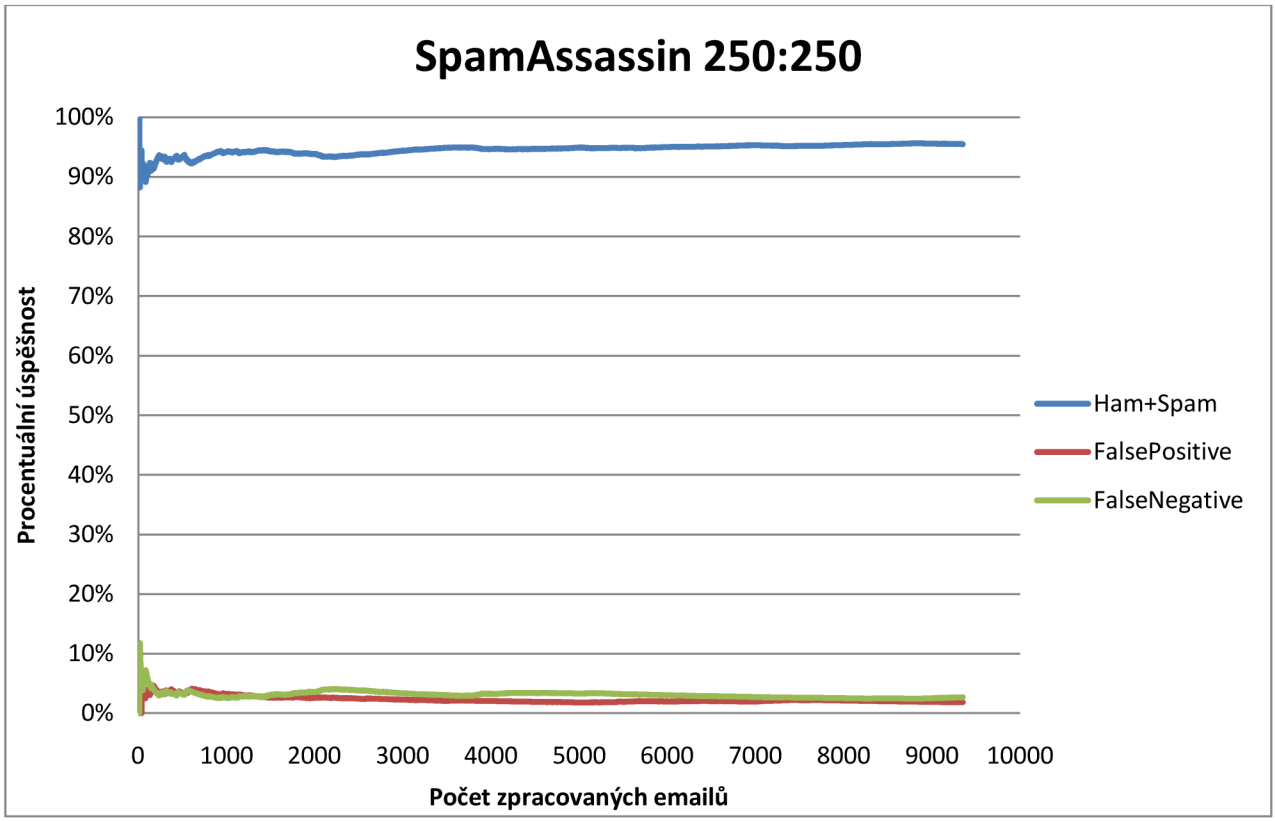


Obrázek č.42: Obálka histogramu - SpamAssassin 250:250, 10.běh

9.3.2 Procentuální úspěšnost - SpamAssassin 250:250



Obrázek č.43: Procentuální úspěšnost systému, SpamAssassin 250:250, 1.běh



Obrázek č.44: Procentuální úspěšnost systému, SpamAssassin 250:250, 10.běh

Diskuze – SpamAssassin 250:250

Z tabulek č.12 a č.12 jde vidět, že výsledky jsou horší oproti předešlým dvěma testům. Pokud nepočítáme první běh, tak ve výsledku je horší úspěšnost o 1%. Pozitivní je, že jsme nezaznamenali žádný dramatický propad úspěšnosti v nějakém běhu, jak tomu bylo v předešlých testech. Plyne to hlavně z poměrně značné redukce fáze učení (redukce z varianty 1000:1000 na 250:250).

Obrázek č.39 ukazuje, jak se posunul práh doprava. Další obálky histogramů se opět posouvají s dalšími běhy doleva. Zajímavých grafy jsou obrázcích č.43 a č.44. U prvního zmíněného je vidět, že systém častěji klasifikoval hamy, jako spamy a procentuální úspěšnost se pohybovala pod 90%. Pozitivní je stále rostoucí úspěšnost systému s dalšími zpracovávanými emaily. Na druhém zmíněném obrázku je patrné, že se systém dokázal úspěšně adaptovat a v podstatě jeho úspěšnost neklesne pod hranici 90%.

9.4 Ling – 1000:1000

Z fáze učení slovník obsahuje 37444 ohodnocených slov, z těchto slov se vytvořilo 6564 ham lymfocytů a 8351 spam lymfocytů. Velikost slovníku po zpracování emailů je 83597 slov.

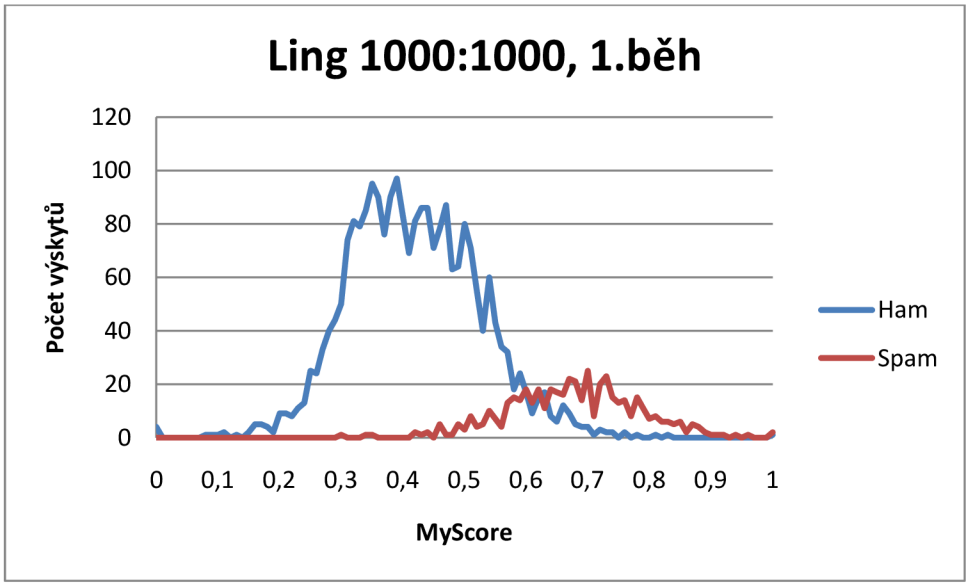
Běh	1.	2.	3.	4.	5.
Ham	2339	2349	2355	2378	2385
Spam	313	461	470	466	466
FalsePositive	168	20	11	15	15
FalseNegative	67	57	51	28	21
Accuracy (%)	**91,860**	97,333	97,852	98,511	98,753
Recall (%)	**65,073**	95,842	97,713	96,881	96,881
Precision (%)	**82,368**	88,996	90,211	94,332	95,688
Miss Rate (%)	**34,927**	4,158	2,287	3,489	3,119
Error (%)	**8,140**	2,667	2,148	1,489	1,247
Ham lymfocytů (na konci běhu)	15717	23281	30627	31553	31780
Spam lymfocytů (na konci běhu)	6718	6861	7135	7557	7799
Čas zpracovaní/email (ms)	11,43	7,85	7,77	7,5	7,82

Tabulka č.14: Ling 1000:1000, běh 1.-5.

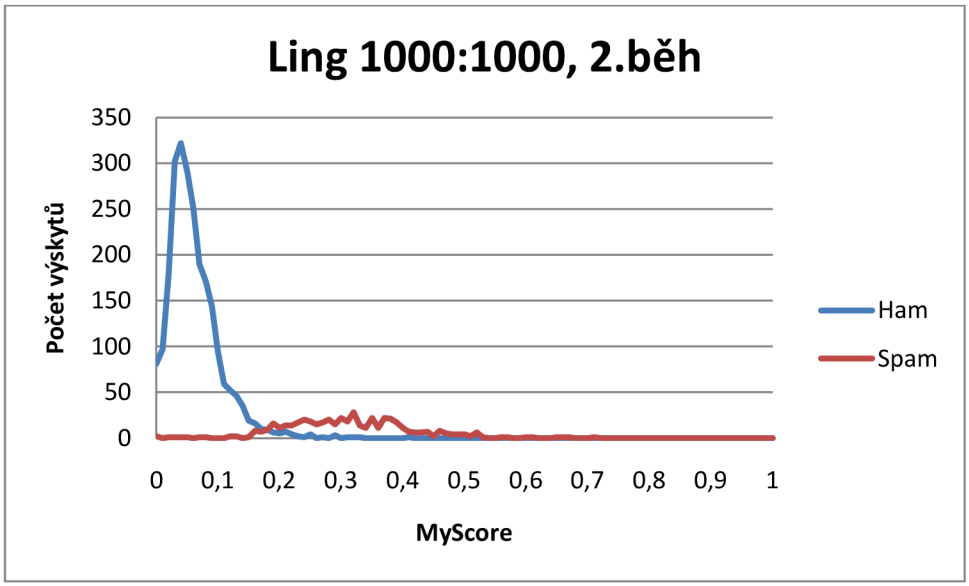
Běh	6.	7.	8.	9.	10.
Ham	2387	2387	2368	2366	2365
Spam	467	467	475	475	475
FalsePositive	14	14	6	6	6
FalseNegative	19	19	38	40	41
Accuracy (%)	**98,857**	**98,857**	98,476	98,407	98,372
Recall (%)	97,089	97,089	**98,753**	**98,753**	**98,753**
Precision (%)	**96,091**	**96,091**	92,593	92,233	92,054
Miss Rate (%)	2,911	2,911	**1,247**	**1,247**	**1,247**
Error (%)	**1,143**	**1,143**	1,524	1,593	1,628
Ham lymfocytů (na konci běhu)	54208	54309	54594	54563	54835
Spam lymfocytů (na konci běhu)	9043	9051	10511	10520	10757
Čas zpracovaní/email (ms)	7,71	7,38	7,38	7.38	7,45

Tabulka č.15: Ling 1000:1000, běh 6.-10.

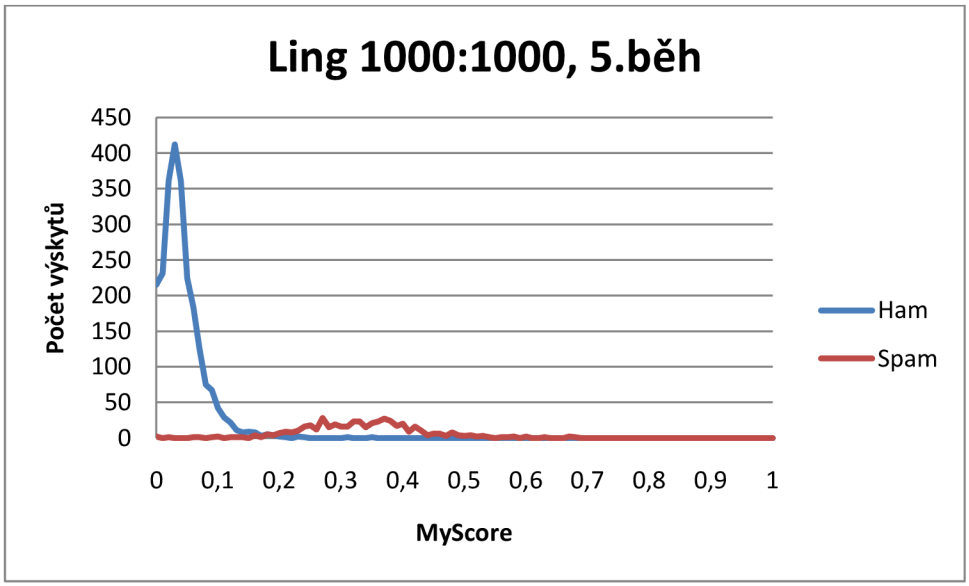
9.4.1 Obálky histogramů – Ling 1000:1000



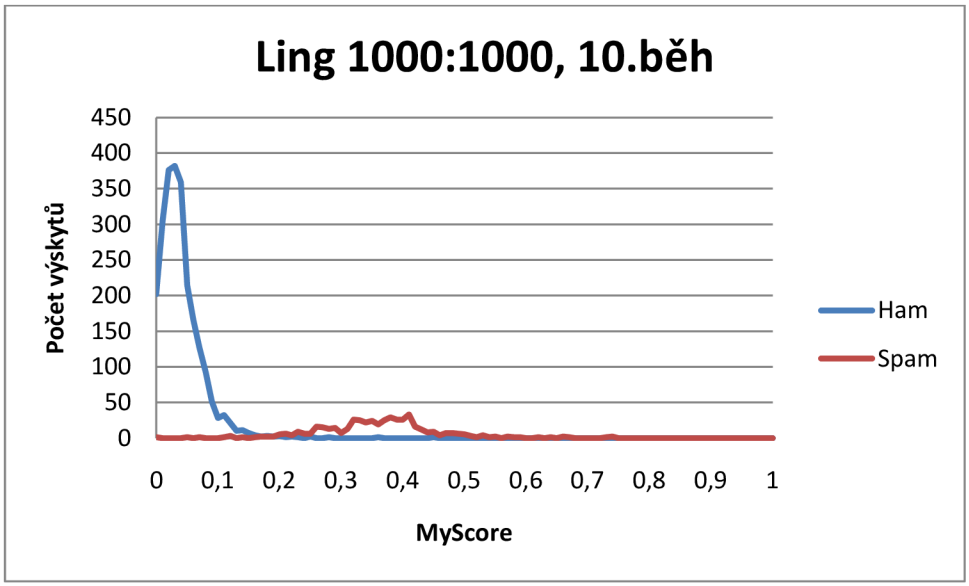
Obrázek č.45: Obálka histogramu - Ling 1000:1000, 1.běh



Obrázek č.46: Obálka histogramu - Ling 1000:1000, 2.běh

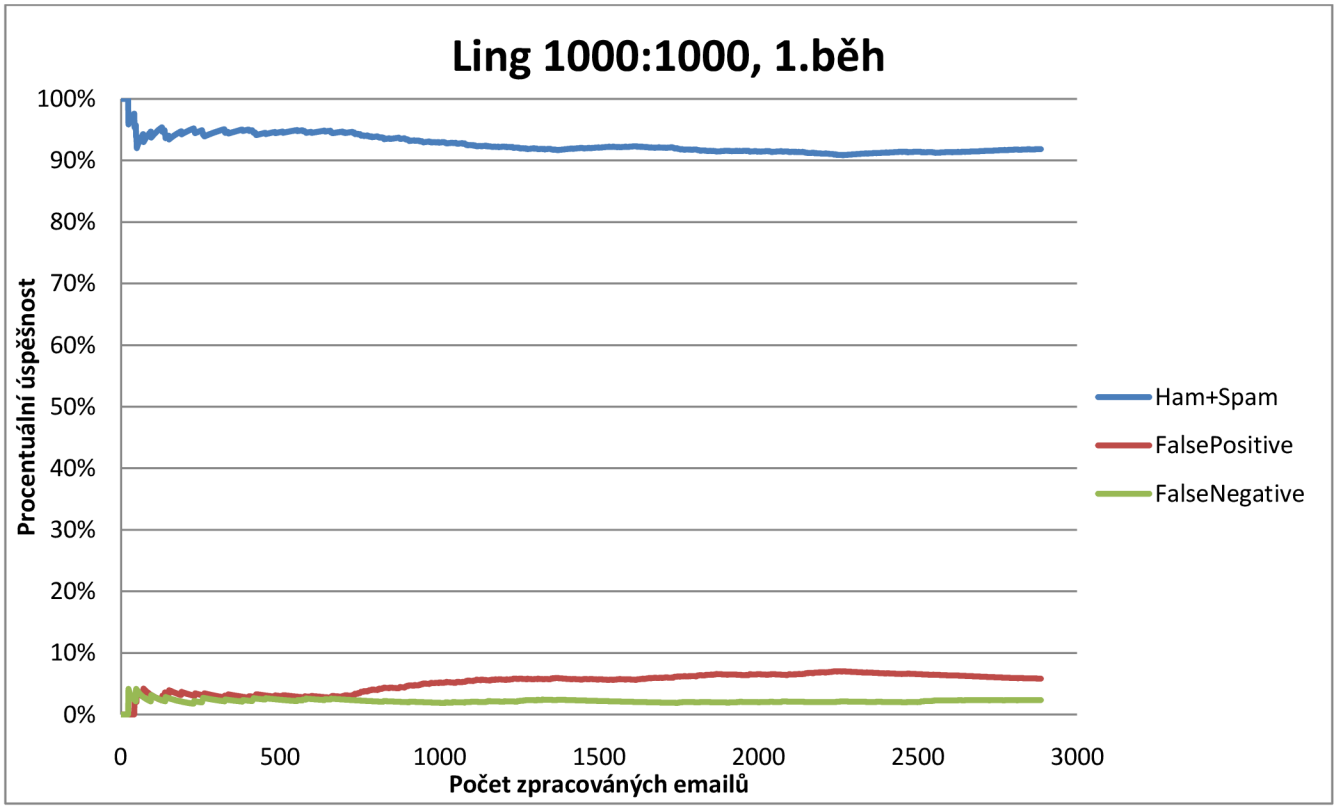


Obrázek č.47: Obálka histogramu - Ling 1000:1000, 5.běh

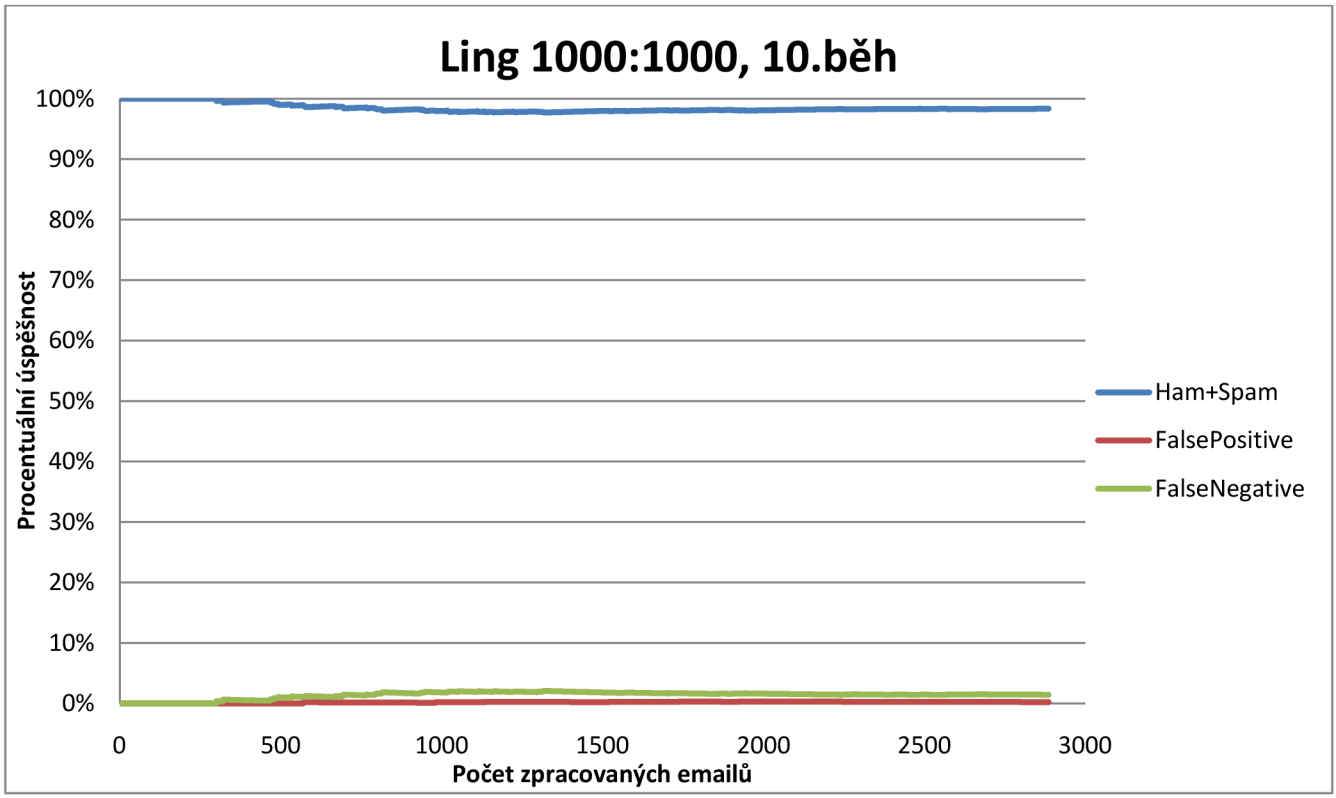


Obrázek č.48: Obálka histogramu - Ling 1000:1000, 10.běh

9.4.2 Procentuální úspěšnost - Ling 1000:1000



Obrázek č.49: Procentuální úspěšnost systému, Ling 1000:1000, 1.běh



Obrázek č.50: Procentuální úspěšnost systému, Ling 1000:1000, 10.běh

Diskuze – Ling 1000:1000

Tento test byl náročnější, jak všechny předchozí, protože trénovací a testovací množina, byli rozdílné. Ne příliš slibné výsledky prvního běhu v tabulce č.14 nejsou žádným překvapením. Za mimořádné, lze považovat výsledek šestého a sedmého běhu v tabulce č.15, kdy se úspěšnost blížila k 99% hranici.

Zajímavé jsou obálky histogramů, kdy od druhého běhu a dál v podstatě žádný email nepřesáhne hranici hodnoty 0,6 u MyScore. Je to dáno také tím, že když se podíváme na poměr ham:spam lymfocytů na konci 10. běhu, tak je to 54835:10727.

Na obrázku č.49 je vidět, že si systém drží úspěšnost nad 90%, zajímavostí tohoto korpusu je to, že chyba FalsePositive je vyšší oproti FalseNegative. U testu číslo 3 to bylo obráceně. V 10. běhu je na obrázku č.50 vidět, jak systém klasifikuje emaily s přesností nad 98%.

9.5 Ling – 500:500

Z fáze učení slovník obsahuje 29045 ohodnocených slov, z těchto slov se vytvořilo 3397 ham lymfocytů a 3911 spam lymfocytů.

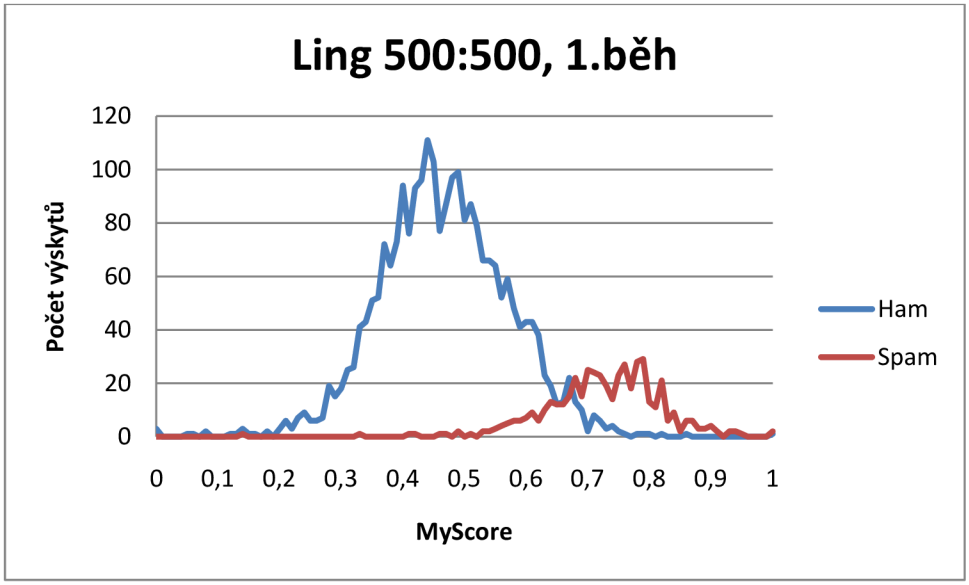
Běh	1.	2.	3.	4.	5.
Ham	2390	2380	2405	2405	2405
Spam	253	405	354	360	379
FalsePositive	228	76	127	121	102
FalseNegative	16	26	1	1	1
Accuracy (%)	**91,548**	96,467	95,566	95,774	96,432
Recall (%)	**52,599**	84,200	73,597	74,844	78,794
Precision (%)	94,052	93,968	99,718	99,723	**99,737**
Miss Rate (%)	**47,401**	15,800	26,403	25,156	21,206
Error (%)	**8,542**	3,533	4,434	4,226	3,568
Ham lymfocytů (na konci běhu)	14464	22529	30670	31712	31879
Spam lymfocytů (na konci běhu)	2005	2246	2485	2744	2907
Čas zpracovaní/email (ms)	11,02	6,97	7,13	7,74	7,50

Tabulka č.16: Ling 500:500, běh 1.-5.

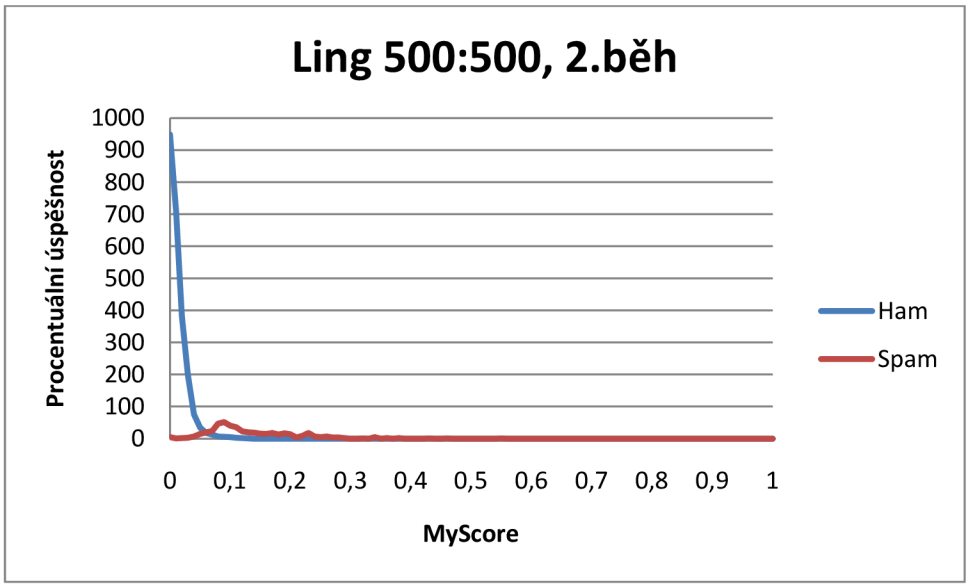
Běh	6.	7.	8.	9.	10.
Ham	2396	2389	2402	2398	2357
Spam	436	447	433	441	470
FalsePositive	45	34	48	40	11
FalseNegative	10	17	4	8	49
Accuracy (%)	98,095	98,233	98,199	**98,337**	97,922
Recall (%)	**90,644**	92,931	90,021	91,684	**97,713**
Precision (%)	97,758	96,336	99,085	98,218	90,559
Miss Rate (%)	9,356	7,069	9,979	8,316	**2,287**
Error (%)	1,905	1,737	1,801	**1,663**	2,078
Ham lymfocytů (na konci běhu)	55097	54891	55291	55172	55252
Spam lymfocytů (na konci běhu)	3769	3874	4864	4953	5273
Čas zpracovaní/email (ms)	7,88	6,99	7,54	7,73	7,41

Tabulka č.17: Obálka histogramu - Ling 500:500, běh 6.-10.

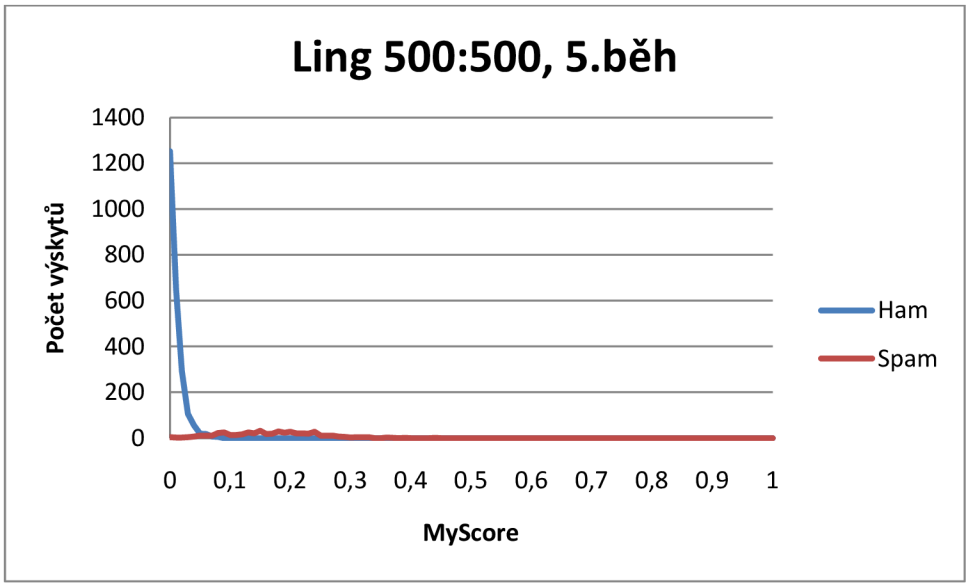
9.5.1 Obálky histogramů – Ling 500:500



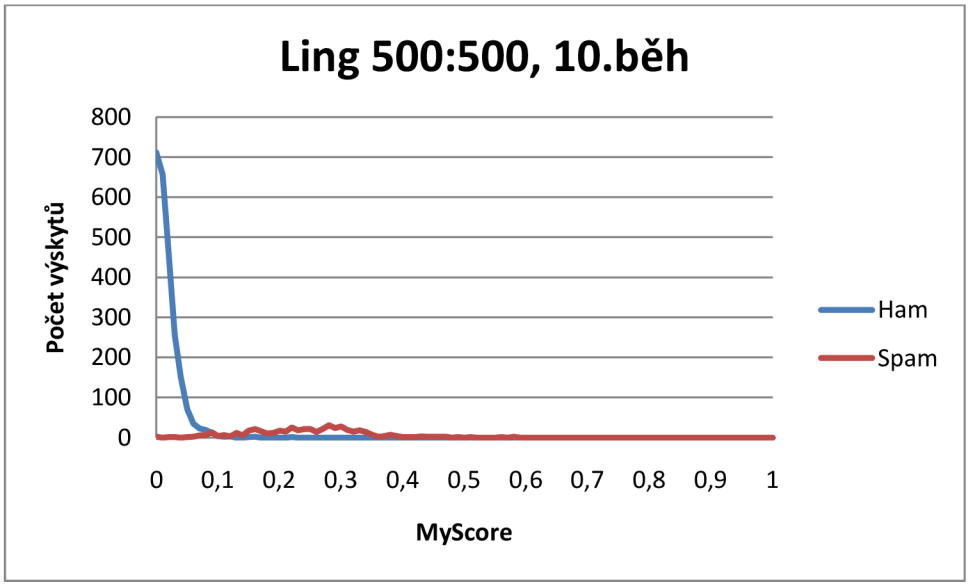
Obrázek č.51: Obálka histogramu - Ling 500:500, 1.běh



Obrázek č.52: Obálka histogramu - Ling 500:500, 2.běh

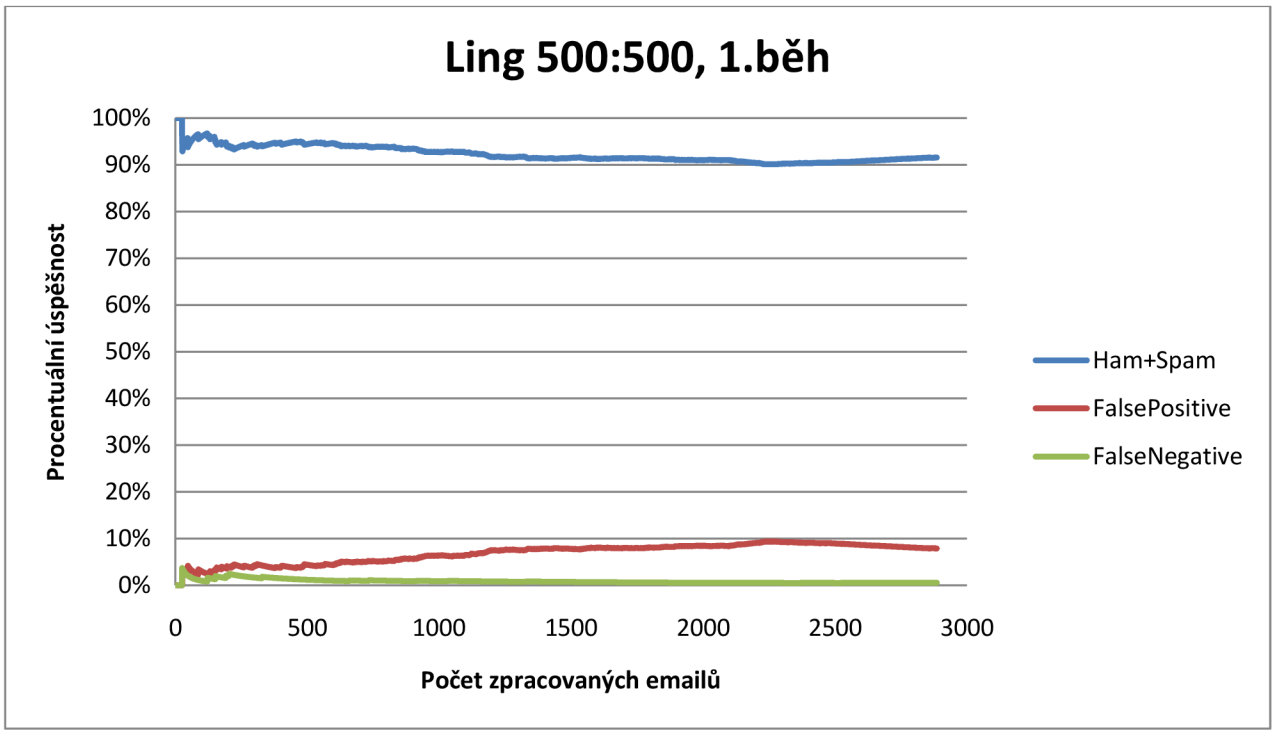


Obrázek č.53: Obálka histogramu - Ling 500:500, 5.běh

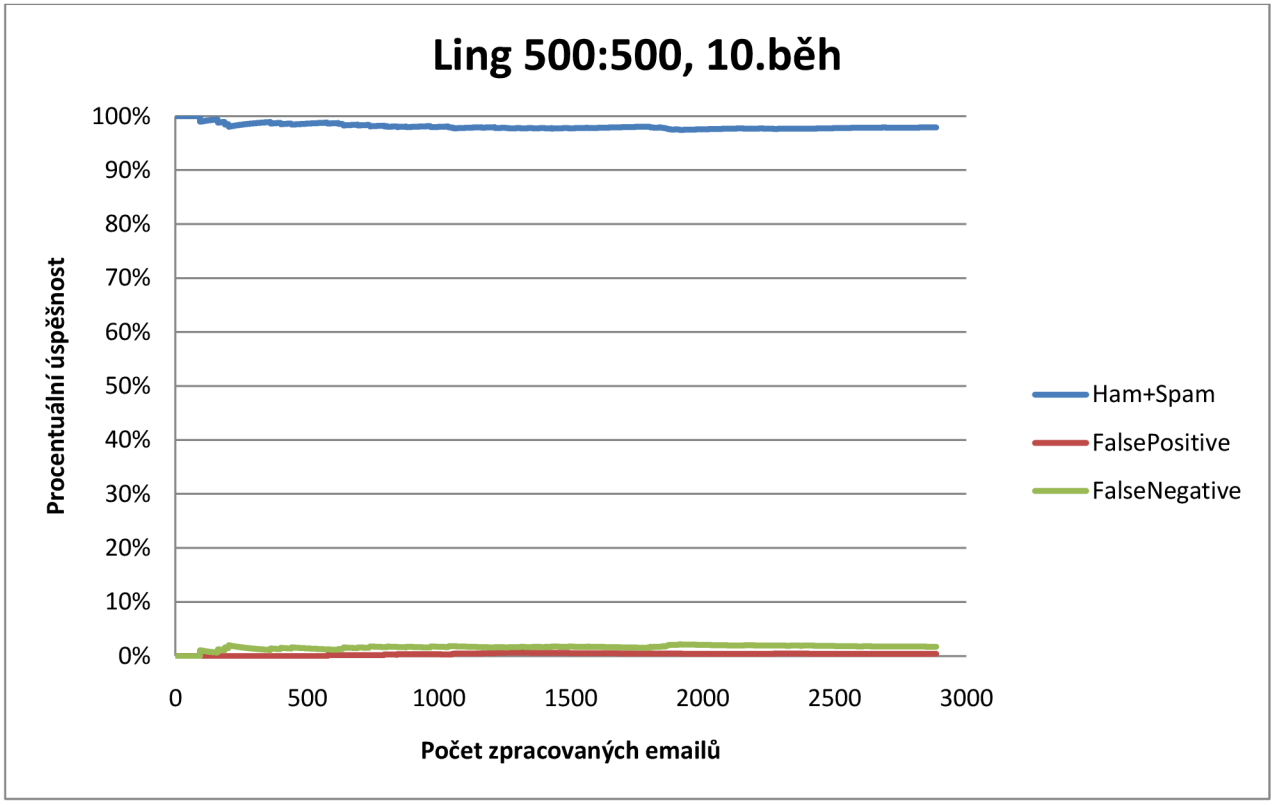


Obrázek č.54: Obálka histogramu - Ling 500:500, 10.běh

9.5.2 Procentuální úspěšnost - Ling 500:500



Obrázek č.55: Procentuální úspěšnost systému, Ling 500:500, 1.běh



Obrázek č.56: Procentuální úspěšnost systému, Ling 500:500, 10.běh

Diskuze – Ling 500:500

I při redukované trénovací množině, byl systém schopen se adaptovat na jiný druh emailů, než na jakém byl natrénován. Systém se také dostal na hranici 98% úspěšnosti. Zajímavostí jsou obálky histogramů, kde došlo k rychlejšímu poklesu práhu.

Na obrázku č.46 je práh cca 0,2 a u č.52 je práh cca 0,06, což je dramatický rozdíl, je to dáno tím, že z fáze učení nemají slova dostatečně vysoké ohodnocení a tedy lymfocyty mají nižší value. Při startu systémů bylo vytvořeno 3911 spam lymfocytů a na konci prvního běhu v systému „žilo“ pouze 2005 spam lymfocytů, tedy pouze polovina. Na konci prvního běhu systému v předešlém testu (č.4) „žilo“ 6718 spam lymfocytů.

Na obrázku č.55 lze oproti obrázku č.49 pozorovat strmější růst chyb typu FalsePositive.

9.6 Ling – 250:250

Z fáze učení slovník obsahuje 25900 ohodnocených slov, z těchto slov se vytvořilo 2035 ham lymfocytů a 3391 spam lymfocytů.

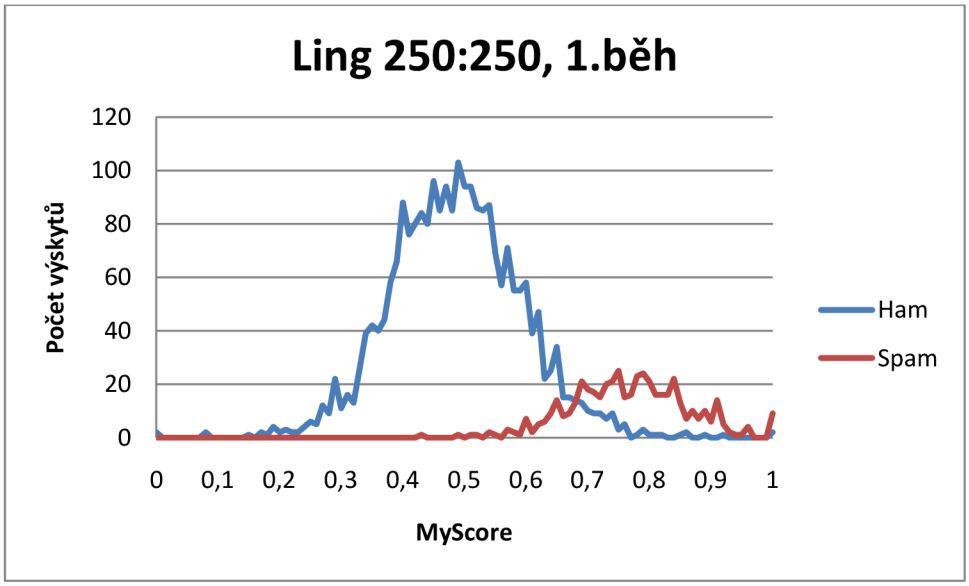
Běh	1.	2.	3.	4.	5.
Ham	2375	2378	2400	2372	2389
Spam	315	422	411	461	460
FalsePositive	166	59	70	20	21
FalseNegative	31	27	6	34	17
Accuracy (%)	**93,176**	97,020	97,368	98,130	98,684
Recall (%)	**65,176**	87,734	85,447	95,842	95,634
Precision (%)	**91,040**	93,987	**98,561**	93,131	96,436
Miss Rate (%)	**34,511**	12,266	14,553	4,158	4,366
Error (%)	**6,824**	2,980	2,632	1,870	1,316
Ham lymfocytů (na konci běhu)	13259	21349	29452	29939	29970
Spam lymfocytů (na konci běhu)	1710	2121	2564	3002	3342
Čas zpracovaní/email (ms)	9,02	7,09	7,09	7,21	7,22

Tabulka č.18: Ling 250:250, běh 1.-5.

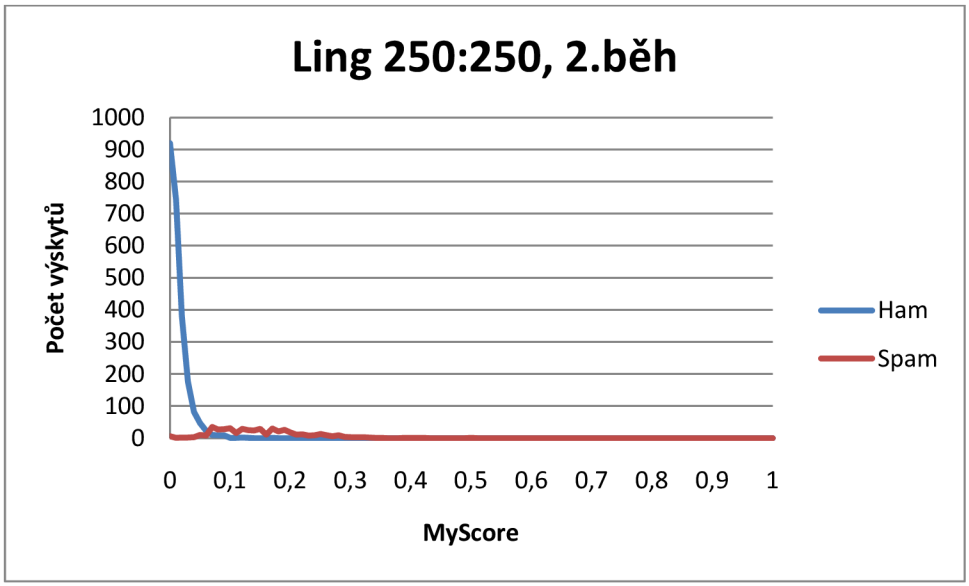
Běh	6.	7.	8.	9.	10.
Ham	2386	2382	2376	2372	2370
Spam	462	472	473	473	474
FalsePositive	19	9	8	8	7
FalseNegative	20	24	29	34	36
Accuracy (%)	98,649	**98,857**	98,718	98,545	98,511
Recall (%)	96,050	98,129	98,337	98,337	**98,545**
Precision (%)	95,851	95,161	94,223	93,294	92,941
Miss Rate (%)	3,950	1,871	**1,663**	**1,663**	1,455
Error (%)	1,351	**1,143**	1,282	1,455	1,489
Ham lymfocytů (na konci běhu)	53329	53216	53658	53630	53783
Spam lymfocytů (na konci běhu)	4703	4824	6108	6183	6423
Čas zpracovaní/email (ms)	7,36	7,24	7,33	7,52	7,29

Tabulka č.19: Ling 250:250, běh 6.-10.

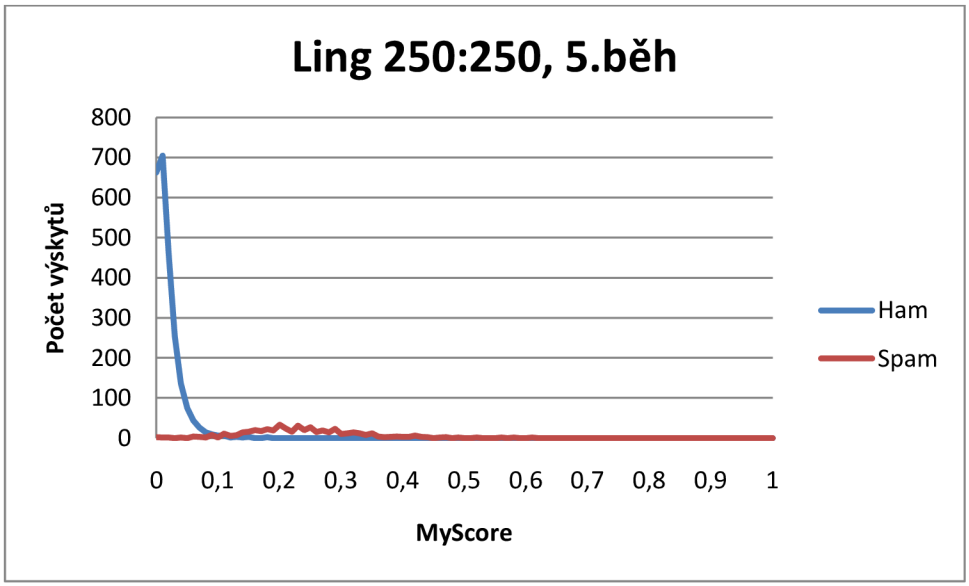
9.6.1 Obálky histogramů – Ling 250:250



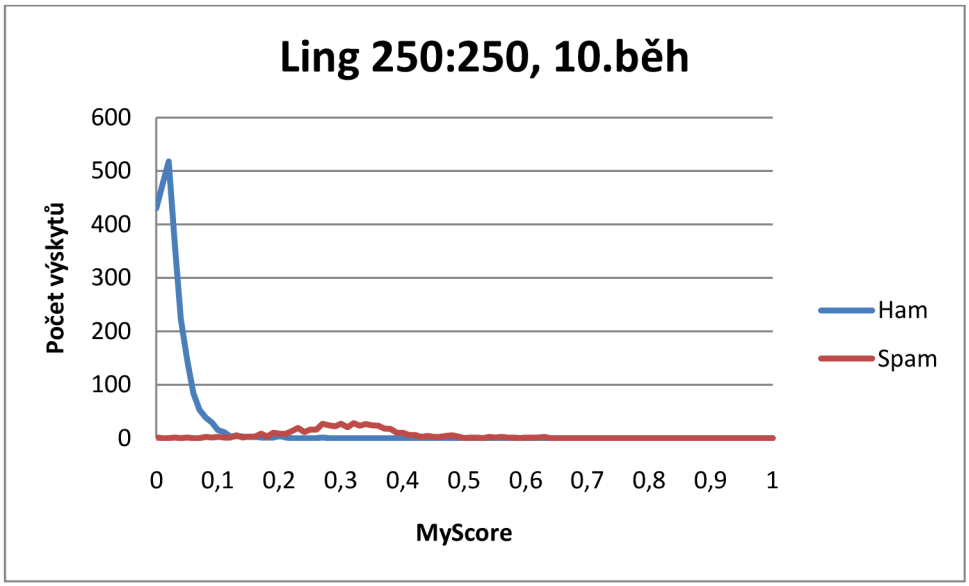
Obrázek č.57: Obálka histogramu - Ling 250:250, 1.běh



Obrázek č.58: Obálka histogramu - Ling 250:250, 2.běh

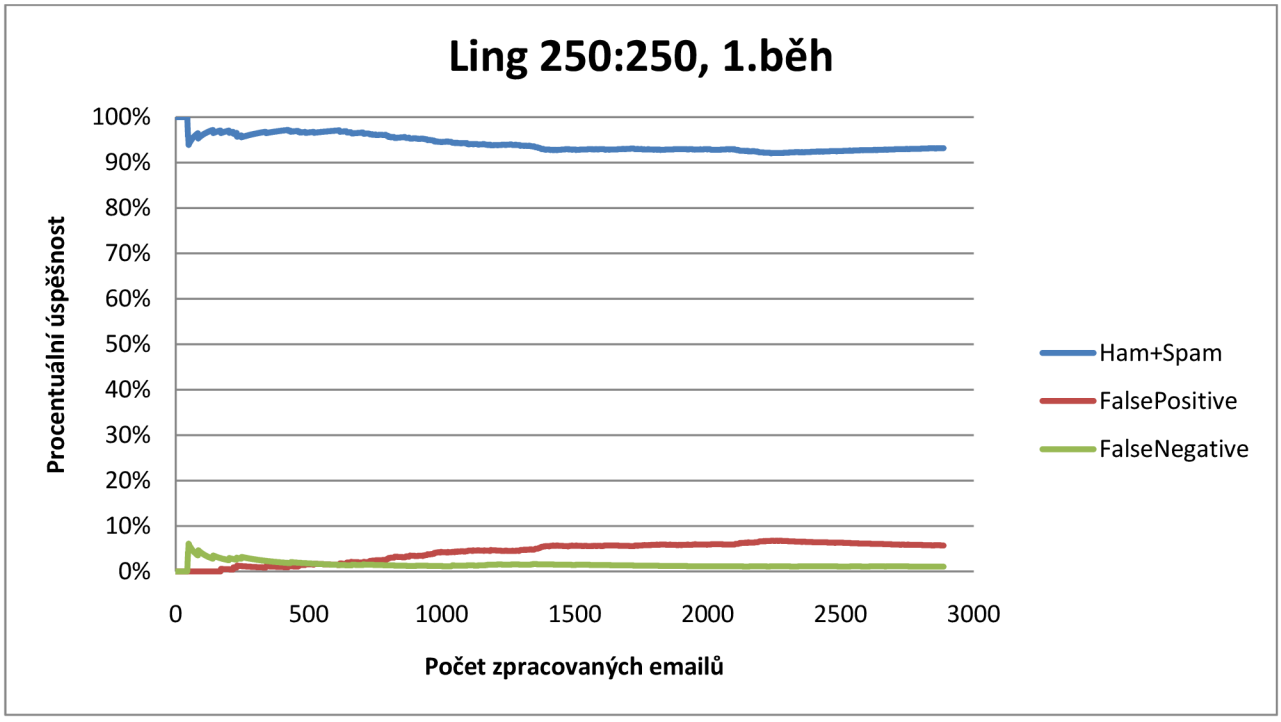


Obrázek č.59: Obálka histogramu - Ling 250:250, 5.běh

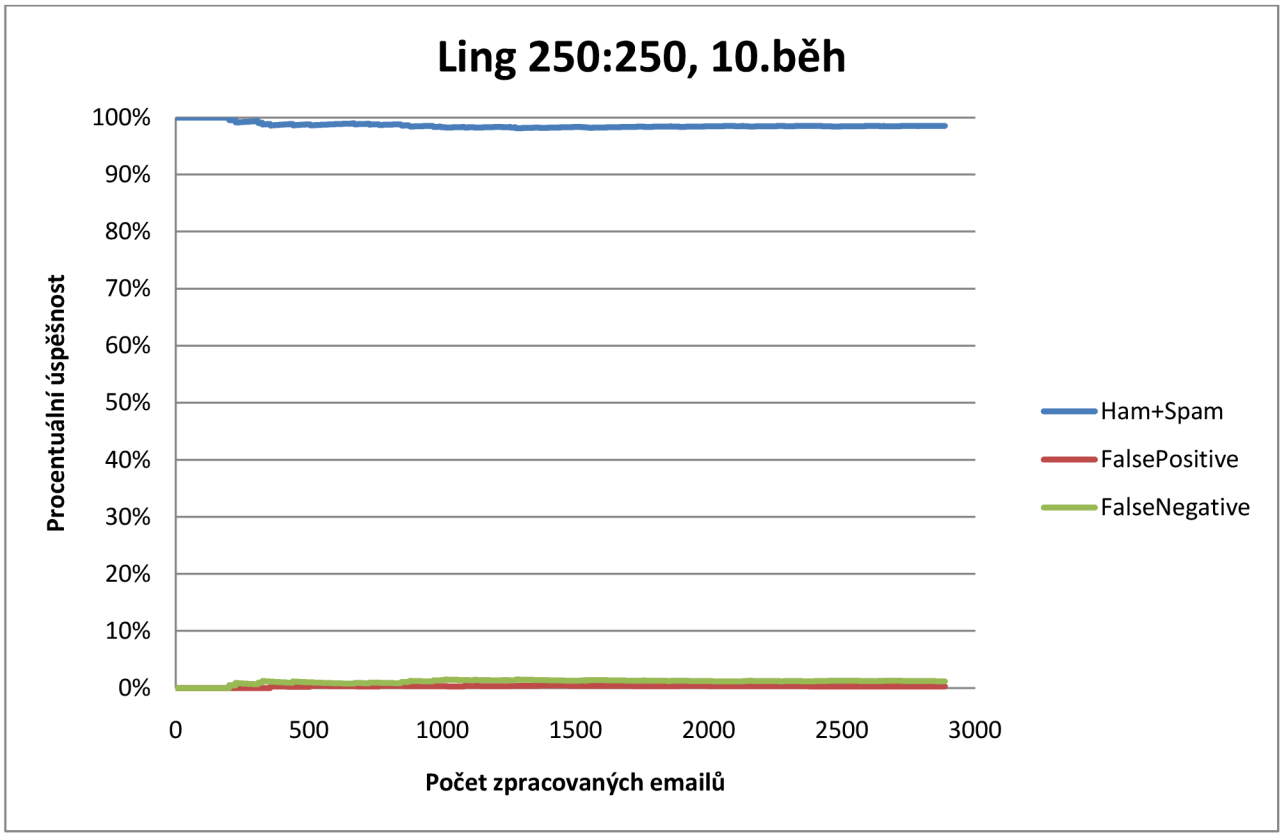


Obrázek č.60: Obálka histogramu - Ling 250:250, 10.běh

9.6.2 Procentuální úspěšnost - Ling 250:250



Obrázek č.61: Procentuální úspěšnost systému, Ling 250:250, 1.běh



Obrázek č.62: Procentuální úspěšnost systému, Ling 250:250, 10.běh

Diskuze – Ling 250:250

Výsledky tohoto testu jsou srovnatelné s předchozím testem č. 5. Důležitým závěrem tohoto testu je fakt, že i při velmi redukované testovací množině 250:250 korpusu SpamAssassin, byl systém se schopen adaptovat na nový druh zpráv. I v tomto testu na tomto korpusu se podařilo překonat hranici 98% úspěšnosti (maximum 98,857%).

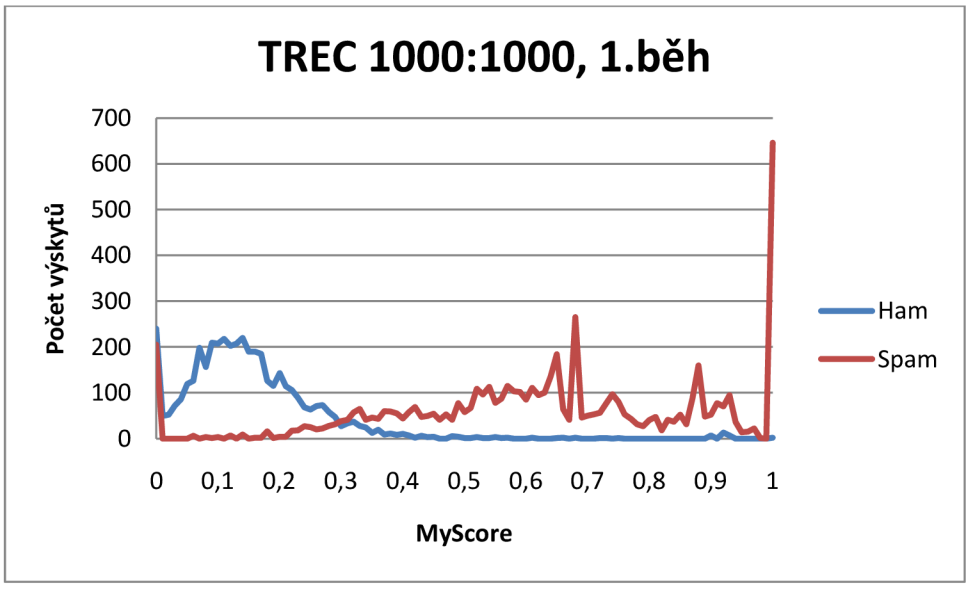
9.7 TREC 1000:1000

Z fáze učení slovník obsahuje 180043 ohodnocených slov, z těchto slov se vytvořilo 14685 ham lymfocytů a 4773 spam lymfocytů. Velikost slovníku po zpracování emailů je 765200 slov. Z důvodu výpočetní náročnosti zpracování tohoto korpusu, bylo provedeno pouze 5 běhů systému.

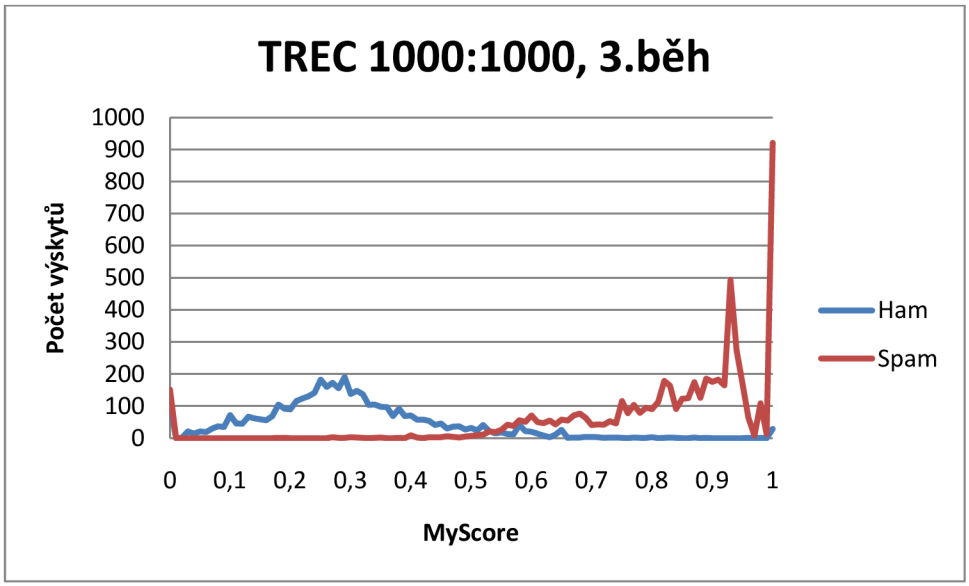
Běh	1.	2.	3.	4.	5.
Ham	3275	3782	3992	3938	4012
Spam	5524	5595	5550	5572	5544
FalsePositive	176	105	150	137	156
FalseNegative	1025	518	308	353	288
Accuracy (%)	**87,990**	93,770	95,420	95,100	**95,560**
Recall (%)	**96,912**	98,158	97,368	**97,600**	97,263
Precision (%)	**84,349**	91,526	94,742	94,042	**95,062**
Miss Rate (%)	**3,088**	1,842	2,632	2,400	**2,737**
Error (%)	**12,010**	6,230	4,580	4,900	**4,440**
Ham lymfocytů (na konci běhu)	18783	29900	66773	177207	nezměřeno
Spam lymfocytů (na konci běhu)	15942	35733	102874	102364	nezměřeno
Čas zpracovaní/email (ms)	12,42	13,26	14,15	28,80	56,31

Tabulka č.20: TREC 1000:1000, běh 1.-5.

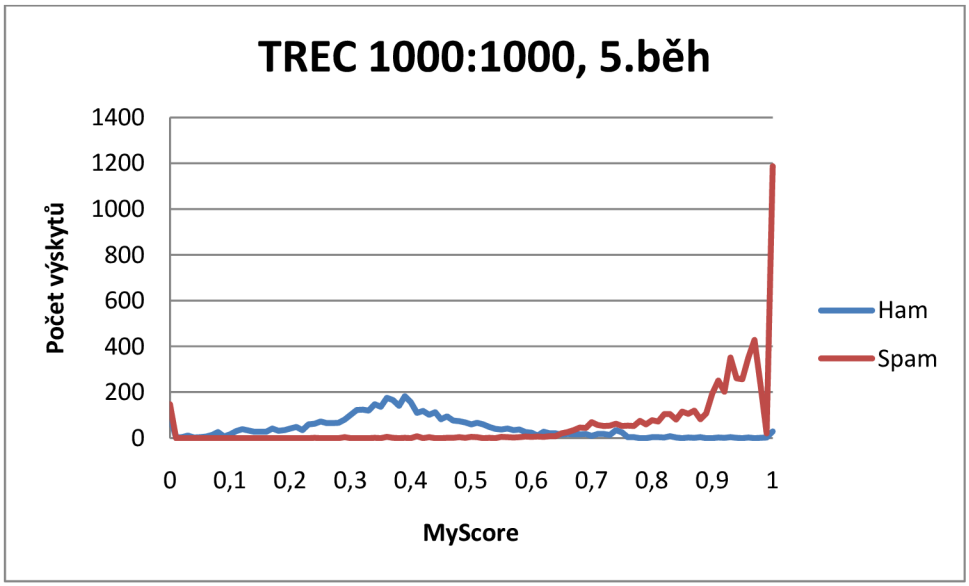
9.7.1 Obálky histogramů – TREC 1000:1000



Obrázek č.63: Obálka histogramu - TREC 1000:1000, 1.běh

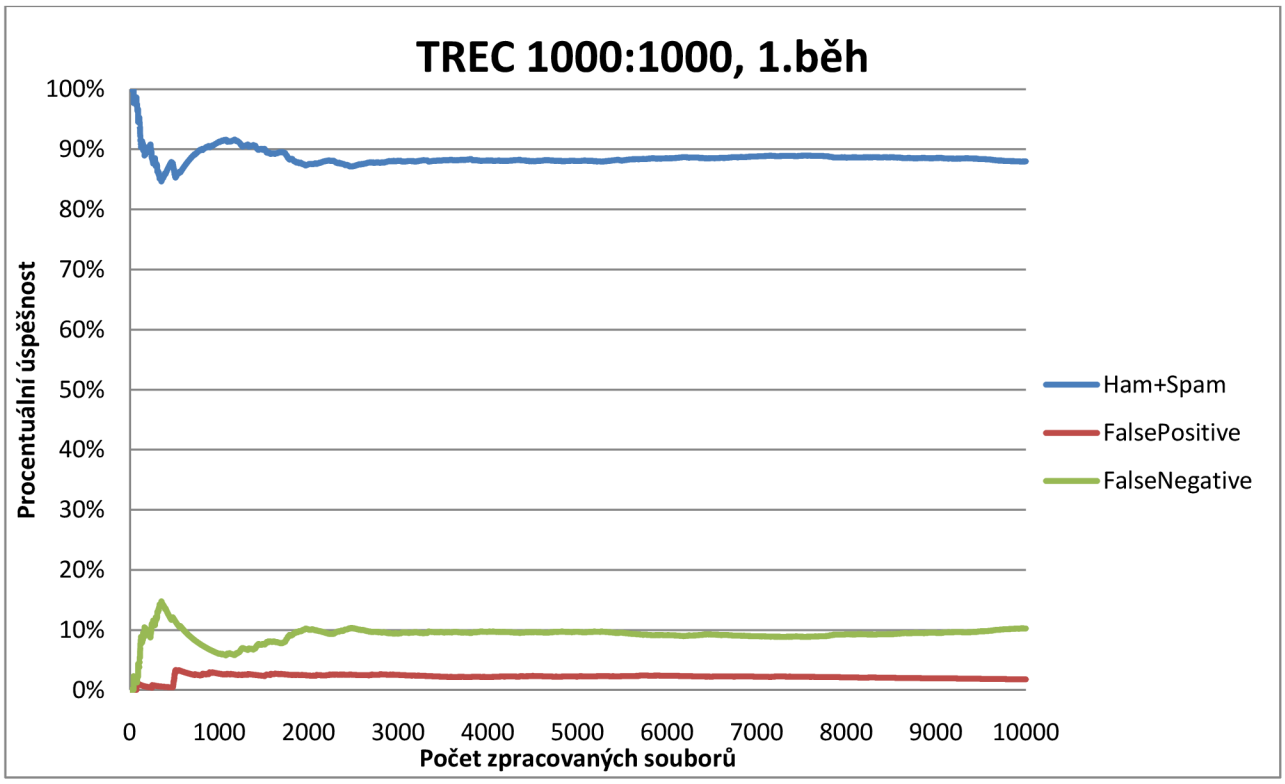


Obrázek č.64: Obálka histogramu - TREC 1000:1000, 3.běh

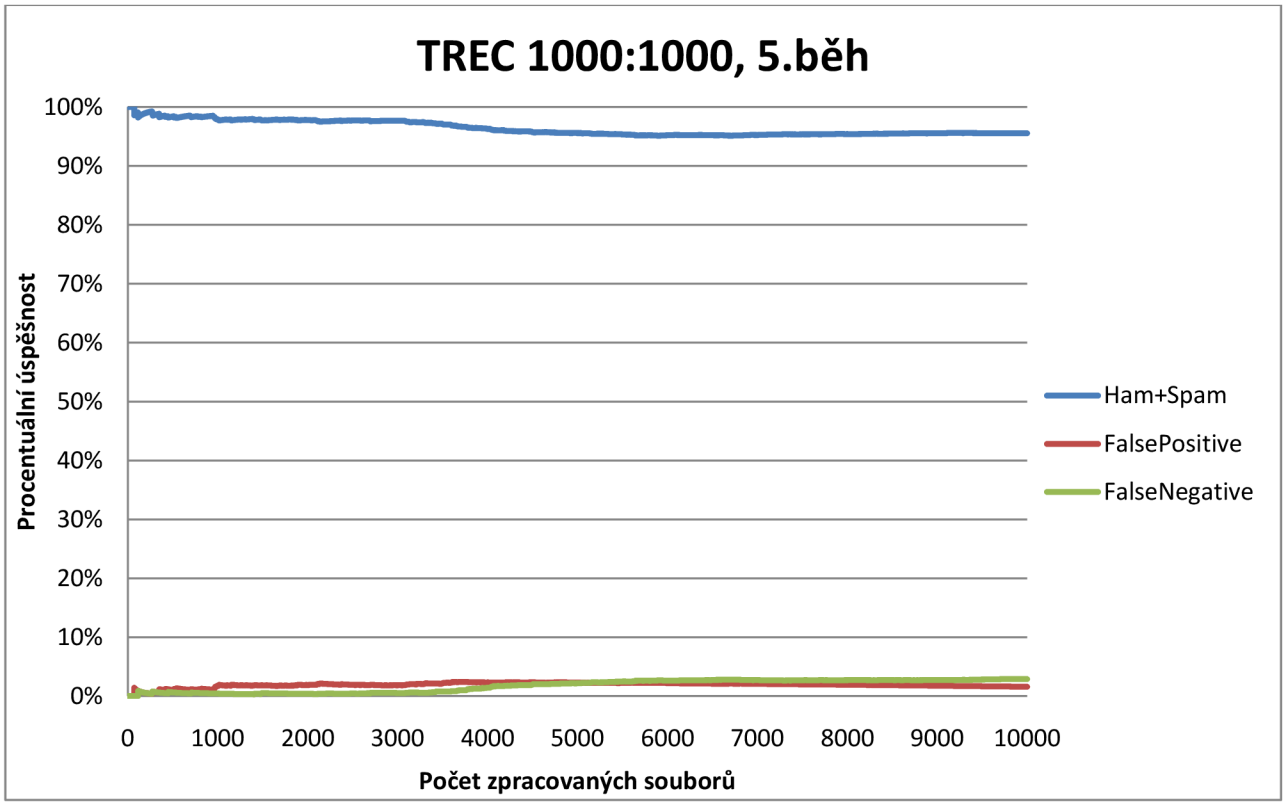


Obrázek č.65: Obálka histogramu - TREC 1000:1000, 5.běh

9.7.2 Procentuální úspěšnost – TREC 1000:1000



Obrázek č.66: Procentuální úspěšnost systému, TREC 1000:1000, 1.běh



Obrázek č.67: Procentuální úspěšnost systému, TREC 1000:1000, 5.běh

Diskuze – TREC 1000:1000

V tomto případě se jednalo také o velmi zajímavý test systému. Oproti minulým testům byl vysoký obsah spamu (57%) a emaily obsahovaly i více textu oproti předešlým testům. To mělo za následek veliký počet lymfocytů (přes 250 000 lymfocytů) a slovník dosáhl velikosti 765 200 ohodnocených slov.

I u tohoto korpusu se podařilo překonat hranici 95% úspěšnosti (po dvou bězích). Přestože bylo „žijících" veliké množství lymfocytů, dokázal systém zpracovat průměrně jeden email pod 100 ms. Podle obálek histogramů lze vypozorovat, že v testovaném korpusu se vyskytovalo cca 200 emailů, které byly stále jednoznačně identifikovány jako ham (MyScore = 0) a těchto 200 emailů/spamů (2%) je zachyceno i na obrázku č.66, jako křivka FalsePositive. Na obrázku můžeme vidět, že zpočátku byl počet výskytů FalseNegative výraznější, ale podle výsledků v tabulce č.20 a také podle obrázku č.67 je zřejmé, že systém tuto chybu v dalších bězích redukoval.

9.8 Výsledky jiných systémů

V článku [Pei-yu] používali korpus CCERT a CASA, tyto korpusy jsem bohužel nemohl použít pro přímou komparaci výsledků z důvodu, že uvedené korpusy obsahují emaily v čínském jazyce a můj systém pracuje s anglickým jazykem. V článku porovnávají výsledky naivního Bayesovského klasifikátoru s jejich upravenou verzí tohoto klasifikátoru. Jejich trénovací množina obsahovala 2000 náhodně vybraných emailů z uvedených korpusů, proto jsem zvolil k porovnání výsledky, kde jsem měl stejný počet trénovacích emailů. Testování prováděli na 400 emailech (300 ham, 100 spam). Uvedené hodnoty MyScore v následující tabulce č.21 jsou vytvořeny aritmetickým průměrem ze všech deseti běhů.

	Naïve Bayesian	Improved filtering algorithm	MyScore 1000:1000 SpamAssassin	MyScore 1000:1000 Ling
Accuracy	94,25	96,79	95,89	97,73
Recall	95,92	97,89	94,40	94,28
Precision	96,4	97,83	90,16	92,07
Miss rate	4,08	2,11	5,59	5,75
Error	5,75	3,21	4,11	2,27

Tabulka č.21: Porovnání výsledků mého systému s výsledky z [Pei-yu]. Zdroj: [Pei-yu]

V článku [Ruan] pracují s metodou „posuvného okna" a s korpusem Ling. V tabulce jsou zaznamenány výsledky z M1 (klasifikační kritérium – Hamming bez mutace), M2 (Hamming s mutací) a M3 (SVM – Support Vector Machines, viz. kap. 6.7.6).

Methods	Accuracy (%)	Precision (%)	Recall (%)	Miss rate (%)
M1	86,09	56,44	77,31	12,16
M2	90,89	77,02	64,3103	3,82
M3	97,04	97,57	84,29	0,42

Tabulka č.22: Naměřené výsledky s posuvným oknem o velikosti 3. Zdroj: [Ruan]

Methods	Accuracy (%)	Precision (%)	Recall (%)	Miss rate (%)
M1	87,04	59,10	77,96	11,16
M2	92,26	83,68	66,19	2,55
M3	97,65	97,10	88,48	0,53

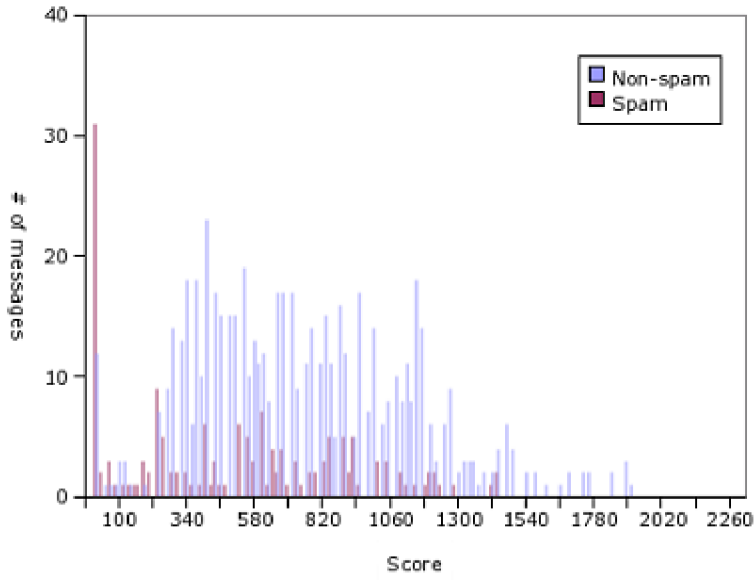
Tabulka č.23: Naměřené výsledky s posuvným oknem o velikosti 5. Zdroj: [Ruan]

V práci [Oda1] jsou ukázány / implementovány 3 rovnice pro výpočet skóre.

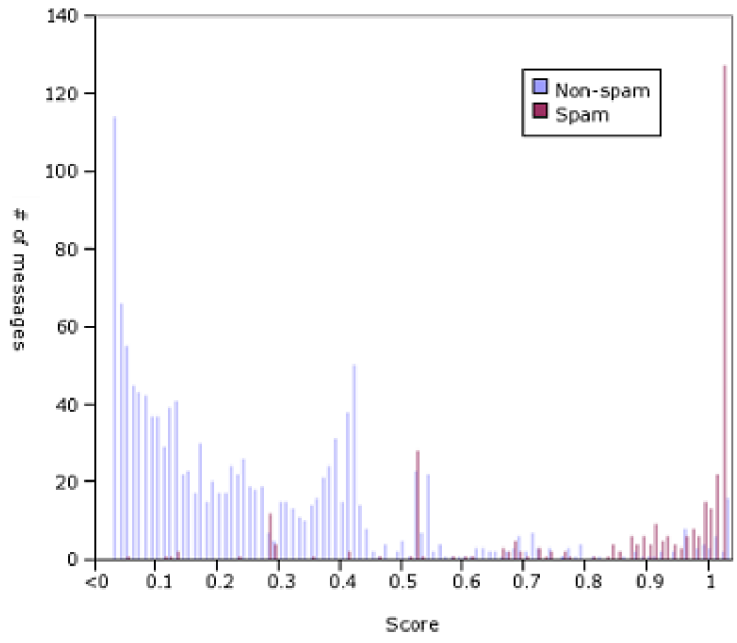
$$Streight\ sum = \sum_{matching\ lymphocytes} spam\_matched \tag{29}$$

$$Weighted\ average = \frac{\sum_{matching\ lymphocytes} spam\_matched}{\sum_{matching\ lymphocytes} msg\_matched} \tag{30}$$

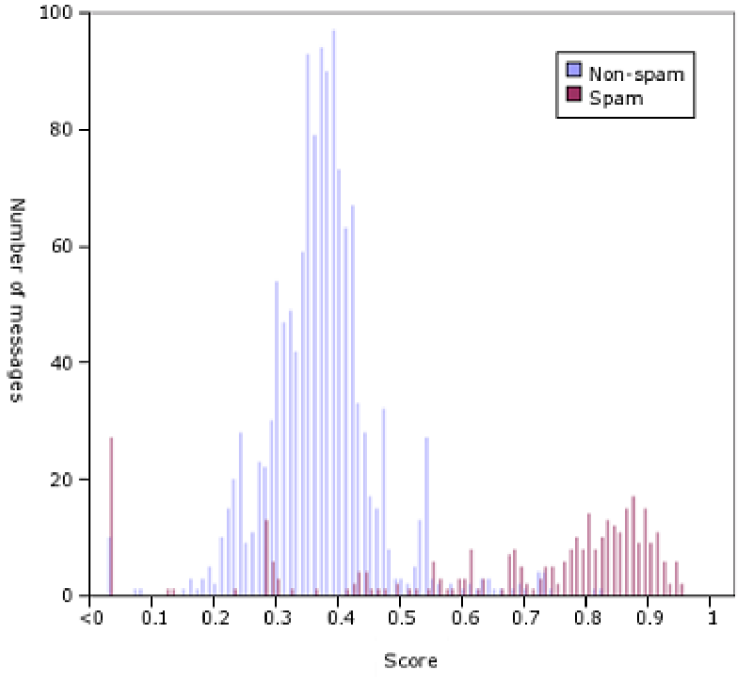
Třetí je výpočet Bayesovského skóre, jak jsme si uvedli v rovnici [Zapletal] v kapitole 6.6.



Obrázek č.68: Distribuční funkce výpočtu skóre podle Streight Sum. Zdroj [Oda1]



Obrázek č.69: Distribuční funkce výpočtu skóre podle Bayes. Zdroj [Oda1]



Obrázek č.70: Distribuční funkce výpočtu skóre podle Weighted Avarage. Zdroj [Oda1]

Scoring System	Thresold	Percent Error	Standard Defiation of Threshold
Streight Sum	3808	20,11	772,62
Bayes	0,62	7,08	0,12
Weighted Average	0,55	4,96	0,01

Tabulka č.24: Průměrné hodnoty práhu pro jednotlivé klasifikátory. Zdroj: [Oda1]

Autorka v závěru práce udává, že průměrně její systém dostahuje úspěšnosti 93,6% s 1,1% FalsePositive.

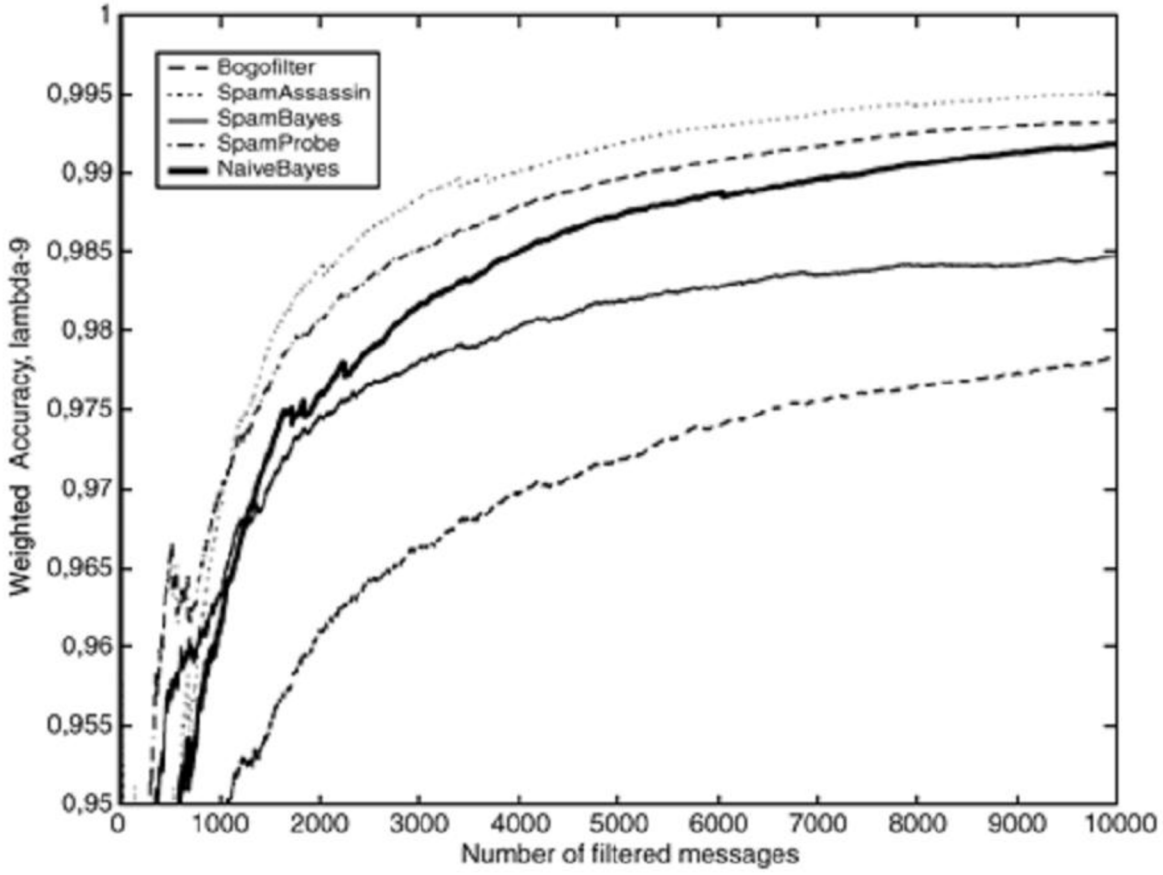
Další srovnání je s diplomovou prací [Neu2009] z roku 2009. Autor měl testovací množinu 100 emailů (20 ham, 80 spam). Spamový filtr průměrně rozpoznal 46,625 spamů z 80 a průměrná doba zpracování jedné zprávy je 28,501 sekund.

Za nejdůležitější výsledky k porovnání můžeme považovat článek [Aguero], ve kterém se srovnávají výsledky anti-spamových filtrů (Bogofilter, SpamAssassin, SpamBayes, SpamProbe, NaiveBayes). Testování probíhalo na korpusu TREC 2005, z něhož bylo vybráno 10 000 emailů, kde 57% byly spamy. Na obrázku č.71 můžeme vidět, jak se „umístili" jednotlivé filtry na základě Weighted Accuracy (W_{acc}), které se vypočítá podle následující rovnice (31):

$$W_{acc} = \frac{\lambda t_n + t_p}{\lambda(t_n + f_p) + (t_p + f_n)} \tag{31}$$

Rovnice (31): Výpočet Weighted Accuracy.

Kde λt_n je váha legitimní zprávy, t_n je počet korektně identifikovaných legitimních zpráv (ham), t_n je počet korektně identifikovaných spamů, f_p je FalsePositive a f_n je FalseNegative. V textu/testech používali $\lambda = 9$.

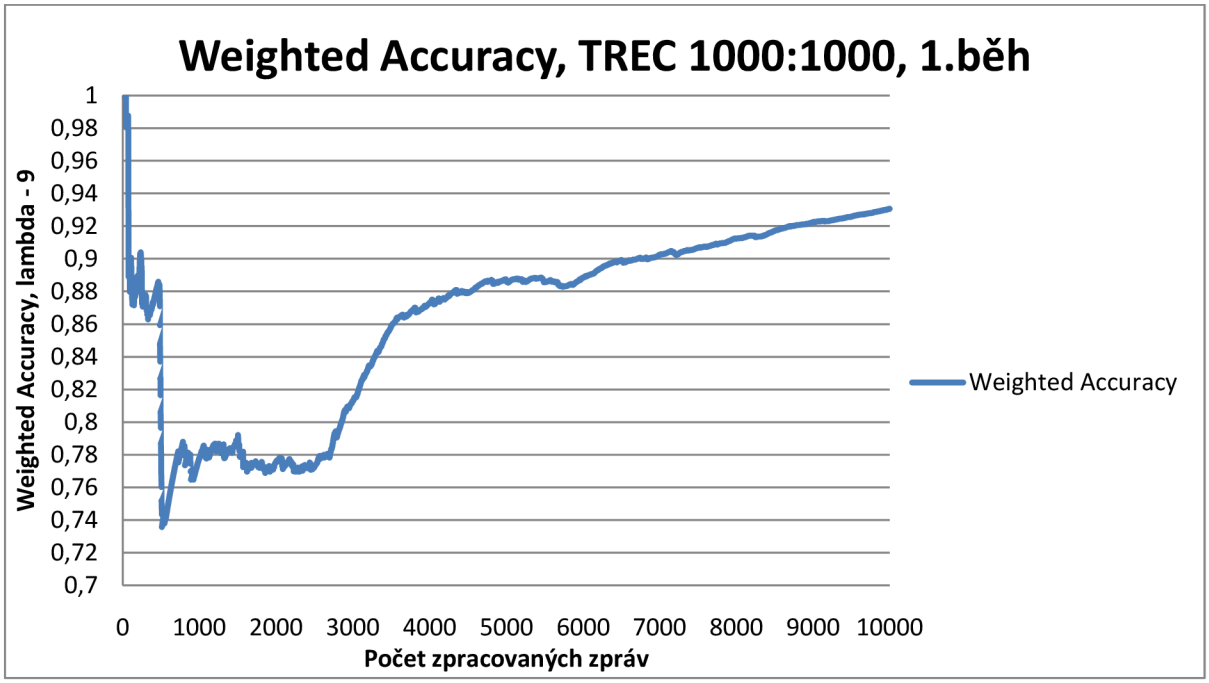


Obrázek č.71: Výsledky testů spam-filtrů na TREC 2005. Zdroj [Aguero]

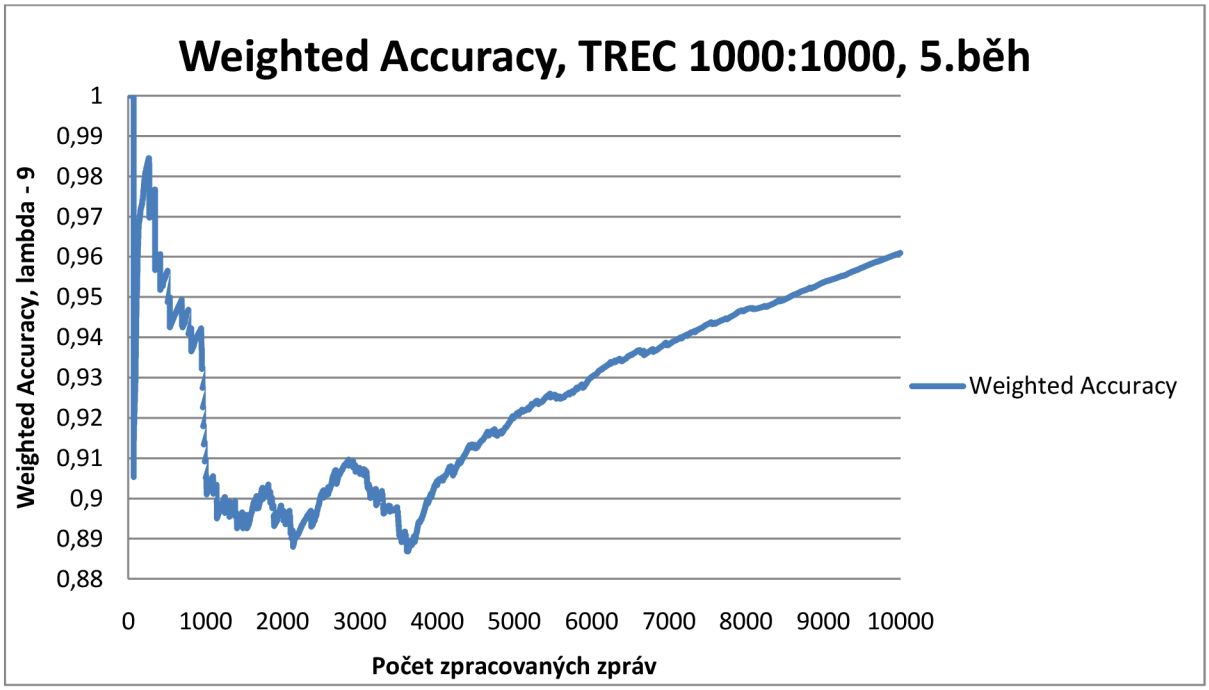
Následující tabulka č.25 ukazuje naměřené W_{acc} při testování korpusu TREC (TREC 1000:1000).

Běh	1.	2.	3.	4.	5.
Wacc	0,931	0,964	0,962	0,963	0,961

Tabulka č.25: Naměřené hodnoty W_{acc} v 7. testu.



Obrázek č.72: Weighted Accuracy, TREC 1000:1000, 1.běh.



Obrázek č.73: Weighted Accuracy, TREC 1000:1000, 1.běh.

9.9 Diskuze k získaným výsledkům

Korpus SpamAssassin:

Průměrná úspěšnost (Accuracy) se pohybuje kolem 96%, což lze považovat za celkem slušný výsledek v porovnání s jinými umělými imunitními systémy. U histogramů můžeme v prvních bězích vidět, že se vypočtený práh pohyboval v intervalu <0.6, 0.7>. Práh je vyšší oproti dalším běhům z toho důvodu, že počet lymfocytů ham a spam jsou zhruba stejně velké. Po prvním běhu jde vidět velký počet (více jak 100) jednoznačně určených emailů (MyScore = 1) jako spam. U další běhů dochází ke snižování práhu z důvodu update slovníku o další korektní/ham slova a také z důvodu, že ve většině emailů je počet korektních slov je větší, jak spamových slov a tím pádem je i větší množina ham lymfocytů oproti spam lymfocytům.

Korpus Ling:

Výsledky testů ukázaly, že systém je schopen se adaptovat na nový druh zpráv i u velmi redukované varianty 250:250 a dokonce po pár bězích se dostat na úspěšnost 98% (v nejlepším případě 98,857%). Když si zanalyzujeme variantu 1000:1000, můžeme v prvním běhu vidět na histogramu, že polovina množiny spam překrývá množinu ham, to vede k většímu množství FalseNegative a FalsePositive výskytu. Naštěstí systém se dokázal adaptovat a v druhém běhu se obě množiny překrývají minimálně. Zajímavé u tohoto korpusu je, že podle histogramu neobsahuje jednoznačné spamy (MyScore = 1).

Grafy průběhu systému jsou u tohoto korpusu zajímavější oproti předchozímu. Při 1. běhu systému jde jasně vidět narůstající křivka FalsePositive. Při pohledu na 10. průběh, jde vidět, že systém pracoval, až na pár chybně identifikovaných spamů, téměř bezchybně.

Výsledky Recall, Precision, Miss Rate nejdou jednoznačně zhodnotit. U variant 1000:1000 a 250:250 lze považovat hodnoty Recall za dobré (v nejlepším případě 98,753) a ostatní za průměrné. U varianty 500:500 dosáhlo Precision úctihodných 99,737%. Samozřejmě, že tyto hodnoty jsou velice ovlivněny aktuálním vypočteným prahem a proto se výsledky (procenta) „přesýpají“ tam a zpět, proto za univerzální porovnávání považuji celkovou úspěšnost (Accuracy).

Korpus TREC:

Díky tomuto korpusu bylo možné přímo porovnat vytvořený systém oproti jiným systémům. Dodržel jsem počet zpracovaných emailů i procentuální poměr spamu. Porovnání se provádělo podle W_{acc} , kde porovnávané systémy dosahovaly po prvním běhu $W_{acc} > 0,975$ a náš implementovaný systém dosahoval v prvním běhu $W_{acc} = 0,931$ a v dalších bězích $W_{acc} > 0,96$.

Zajímavé bylo pozorovat průměrné časy zpracování jednoho emailu, kde se čas zpracování byl 12ms, resp. 29 při cca 30 000 žijících lymfocytech, respektive cca. 270 000 žijících lymfocytech.

Obecné závěry:

1. Dosažené výsledky zejména v parametrech přesnosti (Accuracy) a Recall jsou srovnatelné se standardními publikovanými technikami.
2. Systém má automatické nastavování práhu pro použitý klasifikátor.
3. Za významný přínos lze považovat rychlost zpracování emailu. To je důsledkem uložení lymfocytů (kde detektor je unigram/slovo) do stromu. Následné prohledávání se blíží lineární časové složitosti.
4. Některé algoritmy jsou zpracováné plně paralelně (např. výpočet optimálního práhu), jiné jsou zpracované semi-paralelně, tedy je použit arbitr, který řídí činnost jednotlivých vláken (např. předzpracování emailů, fáze učení).
5. Cíleně nebyly použité další techniky, které by dále zvyšovaly úspěšnost detekce spamů, jako je Blacklisting a další, které by bylo jednoduché přidat do případné on-line verze.

10 Možnosti rozšíření

V této práci jsem použil jako receptory lymfocytů unigramy, kdy při průchodu stromu, kde jsou lymfocyty uloženy, on-line vzniká regulární výraz. Bylo by užitečné rozšířit tuto techniku o bigramy, trigramy a zabudovat fenomén kontextu pro tělo zprávy.

Paramtery `msg_matched` a `spam_matched` by bylo také možné definovat i jiným způsobem. V první fázi učení by se analyzovaly pouze spam soubory a ze zlomku `spam_matched/msg_matched` pro každý lymfocyt by vyplývala pravděpodobnost/síla lymfocytu specifikovat spam.

Další možností by bylo kombinovat několik klasifikátorů napr. SVM a Bayesův filtr. Je však třeba mít stále na paměti, že v on-line systémech je důležitá i doba trvání klasifikace.

11 Závěr

V této práci jsem se zabýval umělými imunitními systémy pro detekci spamů. Než jsem se pustil do samotného návrhu systému, musel jsem nejprve nastudovat, jak funguje lidský imunitní systém. Dále jsme si popsali základy umělého imunitního systému a jaké jeho části odpovídají biologickému imunitnímu systému. Popsal jsem základní algoritmy používané v umělých imunitních systémech.

Nastudoval jsem dnes běžně používané techniky, které se v boji proti spamu používají a jaké druhy spamu převažují. Popsal jsem i několik druhů klasifikátorů, které je možné použít k identifikování, zda testovaný email je vyžádaný a nebo se jedná o spam. Mezi uvedenými klasifikátory byly uvedeny Bayesovský filtr či SVM (Support Vector Machines), tak i „exotičtější" klasifikátory inspirované paradigmatem kolonie mravenců (AntColony – kap. 6.8).

Cílem nebylo vytvořit systém, který by byl nasazen on-line, nýbrž systém, který na bázi umělého imunitního systému a za pomoci vhodných heuristik bude efektivně analyzovat databáze emailů a rozhodovat, zda se jedná o nevyžádanou poštu, či nikoliv.

Podařilo se vytvořit systém, který obsahuje několik původních technik. Za inovativní lze považovat nahrazení Hammingovy vzdálenosti za reverzní haxorovací funkci (kap. 7.3), pro detekci pozměněných slov. Uložení lymfocytů do stromové struktury, vedlo k vytvoření vyhledávacího algoritmu s lineární časovou složitosti O(N). Dále jsem navrhl nový vzorec pro výpočet Score rovnice č.18, na základě které se rozhoduje, zda je email vyžádaný, či nikoliv. Vytvořený systém je autonomní, tedy nevyžaduje ke své činnosti nastavení rozhodovacího práhu od uživatele, protože sám v procesu testování/zpracování zpráv vypočítá a adaptuje hodnotu práhu.

Systém byl otestován na třech korpusech emailů (SpamAssassin, Ling, TREC 2005) mezi nimiž nebyla definována žádná korelace. Přesonost (Accuracy) klasifikace pro korpus SpamAssassin pohybovala kolem 96%, pro Ling byla překonána hranice přesnosti 98% (viz. kap. 9) a pro TREC 2005 byla dosažena přesnost 95,5%. Pro korpusy SpamAssassin a Ling bylo provedeno nepřímé porovnání s jinými metodami/systémy (nebyli použité stejné korpusy zpráv) a bylo zjištěno, že pro aspekty přesnost (Accuracy) a Recall je systém srovnatelný, dokonce v některých případech dosahoval lepších výsledků.

U korpusu TREC 2005 bylo provedeno přímé porovnání s běžně používanými systémy (SpamAssassin, SpamBayes, SpamProbe,..) pomocí metriky Weighted Accuracy (W_{acc}). Běžně používané systémy dosahovali $W_{acc} > 0{,}975$. Vytvořený systém dosahoval $W_{acc} = 0{,}962$.

Díky použití pseudo-paralelnímu zpracování jednotlivých úloh, dosahuje systém průměrného zpracování jednoho emailu za 6-8 ms, při velikosti slovníku do 100 000 ohodnocených slov, při velikosti slovníku do 1 000 000 je průměrné zpracování jednoho emailu 12-50ms.

Při rozšíření stávajícího systému na on-line verzi a zabudování blacklistů, graylistů a dalších pravidel by bylo možné jeho nasazení např. jako předřazeného hrubého filtru poštovního serveru. Určitě však může sloužit jako testovací platforma pro testování nových filtračních technik.

Literatura

[url-neu] Neuwirth, D.: Umělé imunitní systémy [online]. FIT VUT Brno: 2007, [cit. 2010-10-23]. Dostupné na URL: <http://www.neuwirth.name/AIS/bis.html>

[Castro] De Castro, L. N., Timmis, J.: *Artificial Immune Systems: A New Computational Intelligence Approach*. 2002

[Tschudin] Tschudin, CH., Meyer, T., Yamamoto, L.: *Artificial Immune Systems*. University of Basel: November 24, 2009

[bc-neu] Neuwirth, D.: *Umělé imunitní výpočetní systémy*. Brno: 2007.

[Motol] Kolektiv autorů: *NORMÁLNÍ IMUNITNÍ SYSTÉM.* Ústav imunologie 2. LF a FN Motol, Praha, [cit. 2010-10-23]. Dostupné na URL: <http://www.tigis.cz/Knihy/imuno/normalni_imunitni_system.htm>

[Dunkova] Duňková, J.: *Dendrické buňky*. Přírodovědecká fakulta, Masarykova univerzita, 2007, [cit 2010-10-23]. Dostupné na URL: <http://theses.cz/id/kd7zw1/>

[url-nk] *NK buňky*, [cit 2010-10-23]. Dostupné na URL: http://www.biobran.cz/NK.html

[url-skin] *Imunitní systém kůže.* [cit 2010-10-23]. Dostupné na URL: <http://www.eucerin.cz/skin/immunesystem_of_the_skin.asp>

[Troegel] Trögl, J.:*Člověk a mikroorganismy*. [cit 2010-10-23]. Dostupné na URL: *<fzp.ujep.cz/~trogl/1MIKR9Clovek.ppt>*

[Dzubak] Džubák, J.: *Co je to hoax*. [cit 2010-10-24]. Dostupné na URL: http://www.hoax.cz/hoax/co-je-to-hoax

[url-scam] *Spam a Scam glosář*, [cit. 2010-09-26]. Dostupné na URL: <http://www.spamfighter.com/Lang_CS/FAQ_Glossary.asp>

[Bayliss] Bayliss, C. B.: *General Description od Electronic Mail*. The University of Birmingham, aktualizováno 2000-10-18 [cit. 2010-10-25]. Dostupné na URL: <http://www.email.bham.ac.uk/intro_gen.shtml>

[Matousek] Matoušek, P.: *Síťové aplikace a správa sítí – 4. Poštovní služby*. FIT VUT Brno 2008, [cit. 2010-10-25]

[Faltynek] Faltýnek, L.: *Linux, viry a spam*. 4.srpen 2005, [cit. 2010-10-26]. Dostupné na URL: <http://www.linuxexpres.cz/business/linux-viry-a-spam>

[Macura] Macura, L.: *Co je to SPAM a jak s ním nakládat*. Ústav informačních technologií, Obchodně podnikatelská fakulta v Karviné, Slezská univerzita v Opavě, aktualizováno 2006-12-04, [cit. 2010-10-26]. Dostupné na URL: <http://uit.opf.slu.cz/howto/co-je-to-spam-a-jak-s-nim-nakladat/>

[Zapletal] Zapletal, L.: *Bayesův filtr*. 5. Květen 2005, [cit. 2010-10-26]. Dostupné na URL: <http://www.linuxexpres.cz/modules/marwel/index.php?show=001053000006>

[Oda1] Oda, T., White, T.: *Immunity for spam: an analysis of an artificial immune system for junk email detection*. Carleton University, Ottawa ON, Canada. [cit. 2010-10-28]

[Oda2] Oda, T., White, T.: *Developing an Immunity to Spam*. Carleton University, Ottawa ON, Canada. [cit. 2010-10-28]

[Sim] Sim, Kwee-Bo., Lee, Dong-Wook.: *Modeling of Positive Selection for the Development of a Computer Immune System and a Self-Recognition Algorithm*. International Journal of Control, Automation, and Systems Vol. 1, No. 4, December 2003

[Canova] Canova, A., Freschi, F., Repetto, M.: *Hybrid method coupling AIS and zeroth order deterministic search*. COMPEL: The International Journal for Computation and Mathematics in Electrical and Electronic Engineering, Vol. 24 Iss: 3, str.784 – 795

[Khorski] Khorski, A.: *An Overview of Content-Based Spam Filtering Techniques*. Department of Computer Science, Djillali Liabes University, Bel Abbes, Algeria, 2007

[Abi] Abi-Haidar, A., Rocha, L. M.: *Adaptive Spam Detection Inspired by the Immune System*. Department of Informatics, Indiana University, Bloomington, USA., Instituto Gulbenkian de Ciencia, Oeiras, Portugal: 2008, Artificial Life XI

[Zboril] Zbořil, F., Zbořil, F ml.: *Základy umělé inteligence IZU, studijní opora.* FIT VUT, Brno: 2006

[Dobrovolny] Dobrovolný, P.: *Klasifikace obrazu - algoritmy řízené klasifikace*. Institute of Geography, Masaryk University, Brno, [cit. 2010-12-23]. Dostupné na URL: <http://www.geogr.muni.cz/archiv/vyuka/DPZ_CVICENI/Texty/DZO_07_klasifikace_1.pdf>

[Dolansky] Dolanský, T.: *Klasifikátory a metody hodnocení klasifikace snímku.* Univerzita J.E.Purkyně, Ústí nad Labem, 2006. [cit. 2010-12-23]. Dostupné na URL: <http://*gis.fzp.ujep.cz/files/pr04b_klasifikatory.pdf*>

[Langham] Langhammer, J.: *Řízená klasifikace, Spektrální indexy, Část 3.* Přírodovědecká fakulta, Univerzita Karlova v Praze. [cit. 2010-12-23]. Dostupné na URL: <http://www.natur.cuni.cz/~langhamr/lectures/vtfg2/prednasky/dpz_3/DPZ-3.ppt>

[pingdom] Pingdom.com: *Internet 2009 in numbers*. January 22nd 2010. [cit. 2010-12-29]. Dostupné na URL: <http://royal.pingdom.com/2010/01/22/internet-2009-in-numbers/>

[gorum1]: Guromors.com: *Global Spam Statistics*. April 29th 2010. [cit. 2010-12-29]. Dostupné na URL: <http://gorumors.com/crunchies/global-spam-statistics-april-2010/>

[gorum2] Guromors.com: *Countries With The Most Number Of Hacked Computers*. February 16th 2010. [cit. 2010-12-29]. Dostupné na URL: <http://gorumors.com/crunchies/top-zombie-producing-nations/>

[dmz] DMZGlobal.com: *Spam Statistics Map*. [cit. 2010-12-29]. Dostupné na URL: <http://www.dmzglobal.com/spam-statistics.htm>

[spamhaus] Spamhaus.org: *The 10 Worst Spam Countries*. [cit. 2010-12-29]. Dostupné na URL: <http://www.spamhaus.org/statistics/countries.lasso>

[El-Alfy] El-Alfy, E.-S. M.: *Discovering Classification Rules for Email Spam Filtering with an Ant Colony Optimalization Algorithm*. IEEE Congress on Evolutionary Computation: 2009, str. 1778-1782. ISBN 978-1-4244-2959-2.

[Krcmar] Krčmář, P.: *SpamAssassin: Braňte se proti spamům!*. 25.9.2003. [cit. 10.3.2011]. Dostupné na URL: <http://www.root.cz/clanky/spamassassin-brante-se-proti-spamum/>

[Seraf] Sarafijanovic, S., Hernandez, L., Naefen, R., Le Boudec, J-Y.: *AntispamLab – A Tool for Realistic Evaluation of Email Spam Filters.* CAES 2007-Fourth Conference on Email and Anti-Spam. 2-3 August 2007. Mountain View, California USA.

[url-MS] *Microsoft Visual Studio 2010.* [cit. 3.4.2011]. Dostupné na URL: <http://www.microsoft.com/cze/msdn/vstudio/2010/> .

[Pei-yu] Pei-yu, L., Li-wei, Z., Zhen-fang, Z.: *Research on E-mail Filtering Based On Improved Bayesian*. Journal of Computers, Vol. 4, str.271 – 275, March 2009.

[Ruan] Ruan, G., Tan, Y.: *Intelligent Detect Approaches for Spam*. The State Key Laboratory of Machine Perception, Peking University, China: 2007.

[Neu2009] Neuwirth, D.: *Realizace spamového filtru na bázi umělého imunitního systému,* diplomová práce, Brno, FIT VUT v Brně, 2009.

[Aguero] Agüero, P. D., Moreira, J. C., Liberatori, M., Bonadero, J. C., Tulli, J. C.: *IMPROVING THE PERFORMANCE OF ANTI-SPAM FILTERS USING OUT-OF-VOCABULARY STATISTICS.* Ingeniare, Revista chilena de ingeniería, vol. 10 No.3. str. 386-392. August 6, 2009.

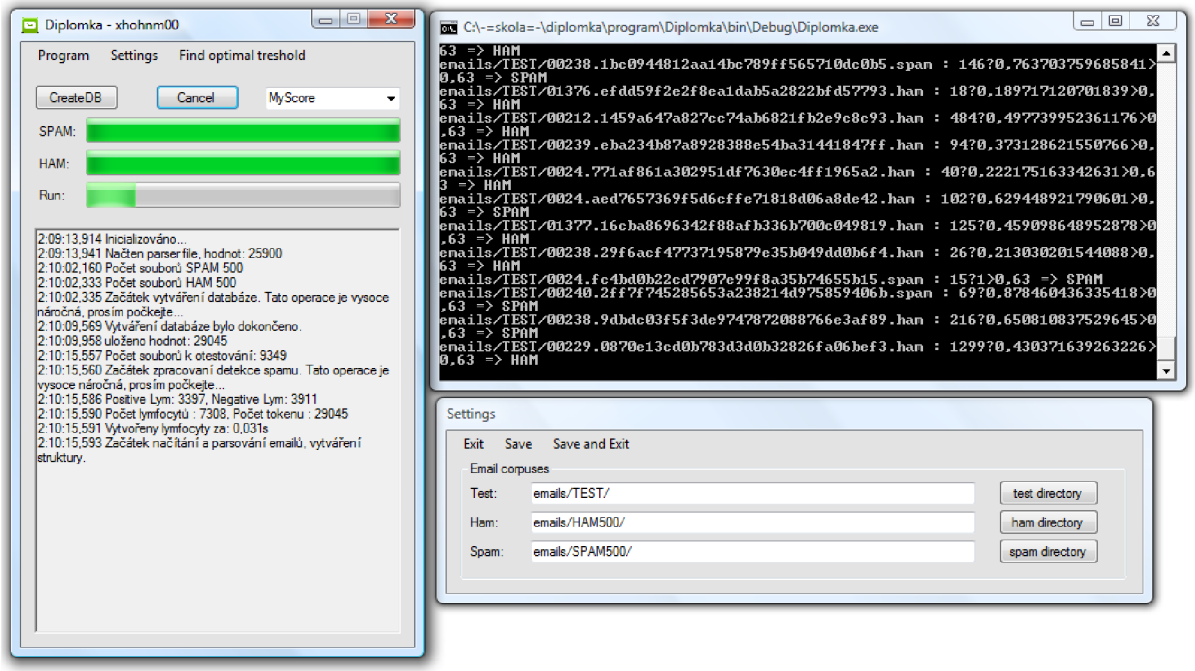
Seznam příloh

Příloha č.1: Ukázka XML – slova

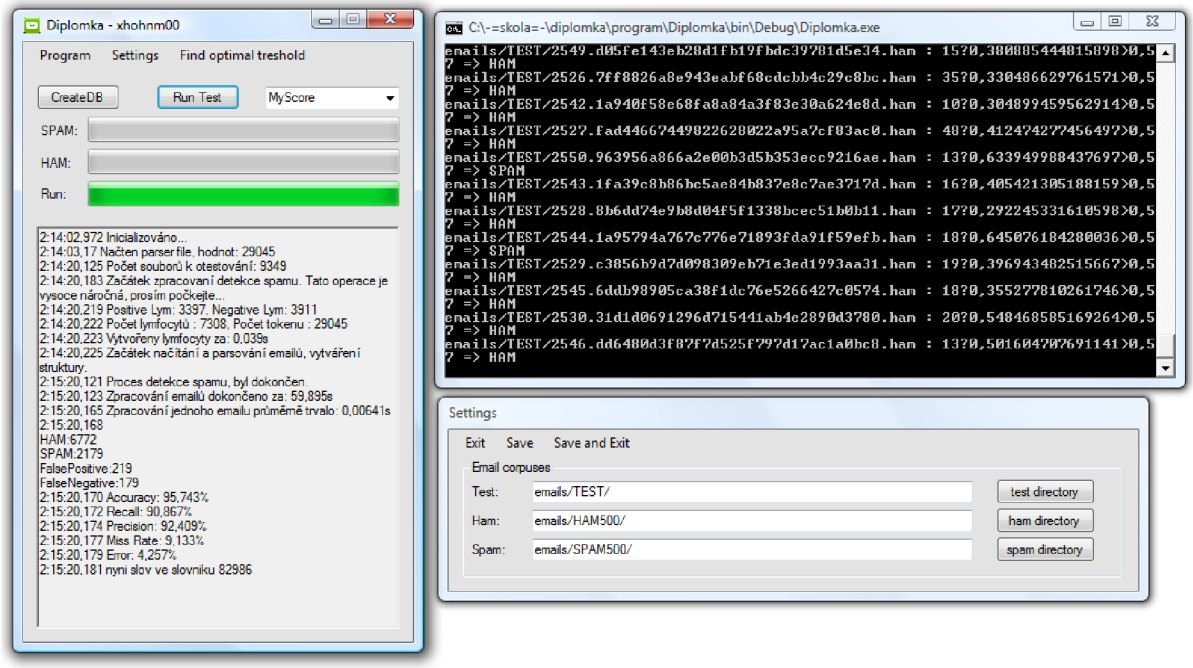
```
<Parser>
    <Words>
        <Word key="they" value="576" />
        <Word key="that" value="498" />
        <Word key="but" value="352" />
        <Word key="it" value="292" />
        <Word key="fork" value="278" />
        <Word key="there" value="256" />
        <Word key="wrote" value="254" />
        <Word key="said" value="184" />
        <Word key="some" value="172" />
        <Word key="think" value="170" />
        .......
        <Word key="click" value="-374" />
        <Word key="from" value="-410" />
        <Word key="content" value="-424" />
        <Word key="here" value="-428" />
        <Word key="with" value="-470" />
        <Word key="business" value="-518" />
        <Word key="please" value="-520" />
        <Word key="we" value="-530" />
        <Word key="mail" value="-552" />
        <Word key="email" value="-634" />
        <Word key="money" value="-644" />
        <Word key="our" value="-700" />
        <Word key="free" value="-728" />
        <Word key="will" value="-744" />
        <Word key="for" value="-1318" />
        <Word key="this" value="-1392" />
    </Words>
</Parser>
```

Příloha č.2: Ukázka aplikace

Po natrénování 500:500 a při zpracování korpusu SpamAssassin



Načtení slovníku z vytvořeného XML souboru a výsledek po 1.běhu.



Příloha č.3: Ovládání aplikace

Při spuštění aplikace se ze souboru `Settings.xml` načte uložená konfigurace, pokud soubor neexistuje, tak se použije defaultní nastavení. Dále se načte `Parser.xml`, který reprezentuje slovník ohodnocených slov.

V příloze č.2 jde vidět tlačítko `CreateDB`, které provede první fázi učení a vytvoří `Parser.xml`. Pomocí tlačítka `Run Test` se spustí testování souborů. Pomocí combo boxu je možné si vyzkoušet výpočet pomocí Balanced Value metody (pouze experiment). Výpočet je možné přerušit. Tlačítko `Run Test` se po kliknutí změní na `Cancel`.

V záložce *Settings* je možné se upravovat zdroj dat emailů. Ve složce *emails/* je předpřipraveno několik testů, které jsem využíval k naměření hodnot. Jak bylo řečeno již v práci, tak po každém běhu systému se uloží obálka histogramu (`histogramReal.csv`) a procentuální úspěšnost (`stats.csv`). Jelikož se jedná o soubory kompatibilní s kancelářským programem *MS Office Excel*, je jednoduché z nich vytvořit grafy.