What have fruits got to do with technology? The case of Apple, Blackberry and Orange

Surender Yerva, Zoltan Miklos, Karl Aberer

Distributed Information Systems Lab
EPFL, Switzerland

Sogndal, Norway, WIMS 2011
May 27, 2011

Motivation

- Online Reputation Management
  - Opinion Mining, Sentiment Analysis etc.
  - Blogs, Comments, Surveys, Micro-blogging, Social Media etc.

Motivation

- Online Reputation Management
  - Opinion Mining, Sentiment Analysis etc.
  - Blogs, Comments, Surveys, Micro-blogging, Social Media etc.

  - Preprocessing step essential for Online Reputation Management tasks.

Motivation

- Online Reputation Management
  - Opinion Mining, Sentiment Analysis etc.
  - Blogs, Comments, Surveys, Micro-blogging, Social Media etc.
  - Preprocessing step essential for Online Reputation Management tasks.
- Entity based search (or retrieval) from Twitter streams.

Motivation

- Online Reputation Management
  - Opinion Mining, Sentiment Analysis etc.
  - Blogs, Comments, Surveys, Micro-blogging, Social Media etc.
  - Preprocessing step essential for Online Reputation Management tasks.
- Entity based search (or retrieval) from Twitter streams.
- Goal: To classify a tweet whether it is related to a particular company.

Some Examples

- ".. installed yesterdays update released by apple .."

Some Examples

- ".. installed yesterdays update released by apple .."
- ".. installed yesterdays update released by **apple**.."

Some Examples

- ".. installed yesterdays update released by apple .."
- ".. installed yesterdays update released by **apple**.." (TRUE)

Some Examples

- ".. installed yesterdays update released by apple .."
- ".. installed yesterdays update released by **apple**.." (TRUE)

- ".. the apple juice was bitter :(.."

Some Examples

- ".. installed yesterdays update released by apple .."
- ".. installed yesterdays update released by **apple**.." (TRUE)

- ".. the apple juice was bitter :(.."
- ".. the **apple** juice was bitter :(.."

Some Examples

- ".. installed yesterdays update released by apple .."
- ".. installed yesterdays update released by **apple**.." (TRUE)

- ".. the apple juice was bitter :(.."
- ".. the **apple** juice was bitter :(.." (FALSE)

Some Examples

- ".. installed yesterdays update released by apple .."
- ".. installed yesterdays update released by **apple**.." (TRUE)

- ".. the apple juice was bitter :(.."
- ".. the **apple** juice was bitter :(.." (FALSE)

- ".. it was easy when apples and blackberries were only fruits.."

Some Examples

- ".. installed yesterdays update released by apple .."
- ".. installed yesterdays update released by **apple**.." (TRUE)

- ".. the apple juice was bitter :(.."
- ".. the **apple** juice was bitter :(.." (FALSE)

- ".. it was easy when apples and blackberries were only fruits.."
- ".. it was easy when **apples** and blackberries were only fruits.."

Some Examples

- ".. installed yesterdays update released by apple .."
- ".. installed yesterdays update released by **apple**.." (TRUE)

- ".. the apple juice was bitter :(.."
- ".. the **apple** juice was bitter :(.." (FALSE)

- ".. it was easy when apples and blackberries were only fruits.."
- ".. it was easy when **apples** and blackberries were only fruits.." (TRUE.. FALSE)

Some Examples

- ".. installed yesterdays update released by apple .."
- ".. installed yesterdays update released by **apple**.." (TRUE)

- ".. the apple juice was bitter :(.."
- ".. the **apple** juice was bitter :(.." (FALSE)

- ".. it was easy when apples and blackberries were only fruits.."
- ".. it was easy when **apples** and blackberries were only fruits.." (TRUE.. FALSE)

- ".. dropped my apple, mind you it is not the fruit :("

Some Examples

- ".. installed yesterdays update released by apple .."
- ".. installed yesterdays update released by **apple**.." (TRUE)

- ".. the apple juice was bitter :(.."
- ".. the **apple** juice was bitter :(.." (FALSE)

- ".. it was easy when apples and blackberries were only fruits.."
- ".. it was easy when **apples** and blackberries were only fruits.." (TRUE.. FALSE)

- ".. dropped my apple, mind you it is not the fruit :("
- ".. dropped my **apple**, mind you it is not the fruit"

Some Examples

- ".. installed yesterdays update released by apple .."
- ".. installed yesterdays update released by **apple**.." (TRUE)

- ".. the apple juice was bitter :(.."
- ".. the **apple** juice was bitter :(.." (FALSE)

- ".. it was easy when apples and blackberries were only fruits.."
- ".. it was easy when **apples** and blackberries were only fruits.." (TRUE.. FALSE)

- ".. dropped my apple, mind you it is not the fruit :("
- ".. dropped my **apple**, mind you it is not the fruit" (Tricky)

Content

- Problem Statement & Formalism
- Our Approach
- Techniques
  - Basic Profile based Classifier
  - Relatedness Factor estimation based Classifier
  - Active Stream Learning based Classifier
- Experiments
- Conclusions

Problem Statement

- Tweet Set: $\Gamma = \{T_1, \ldots, T_n\}$, with a company keyword (ex: apple).
- Classify the tweet T_i whether it is related to the company entity("Apple Inc.").

Problem Statement

- Tweet Set: $\Gamma = \{T_1, \ldots, T_n\}$, with a company keyword (ex: apple).
- Classify the tweet T_i whether it is related to the company entity("Apple Inc.").
- Available Company Information:
  - Company Name (ex : apple)
  - Company URL (ex : http://www.apple.com)
  - Domain (ex : Computer Products)

Problem Statement

- Tweet Set: $\Gamma = \{T_1, \ldots, T_n\}$, with a company keyword (ex: apple).
- Classify the tweet T_i whether it is related to the company entity("Apple Inc.").
- Available Company Information:
  - Company Name (ex : apple)
  - Company URL (ex : http://www.apple.com)
  - Domain (ex : Computer Products)
- Examples:
  - "Already missing **Orange** County! Had an AMAZING time in Florida, but glad to be back home." (Orange: www.orange.ch : Telecommunications ?)
  - "Is **Apple** Delaying the Release of iPhone 5? " (Apple: www.apple.com : Computer Products)
  - **"BlackBerry** Messenger updated to version 5.0.2.12" (Blackberry: www.blackberry.com : Mobile company)

Our Approach

- **Tweet Representation**
  - Bag of keywords:(unigrams)
  - Stemmed words(Porter Stemmer), Removal of tweet-specific stop words(RT, smileys, etc.).

$$T_i = set\{wrd_j\}$$

Our Approach

- **Tweet Representation**
  - Bag of keywords:(unigrams)
  - Stemmed words(Porter Stemmer), Removal of tweet-specific stop words(RT, smileys, etc.).

$$T_i = set\{wrd_j\}$$

- **Representation of Company**:

$$P_c = set\{wrd_j : wt_j\}$$

  - Positive Evidence Keywords

$$P_c.Set^+ = \{wrd_j : wt_j \mid wt_j \geq 0\}$$

  - Negative Evidence Keywords

$$P_c.Set^- = \{wrd_j : wt_j \mid wt_j < 0\}$$

  - Auxiliary Information (Relatedness Factor)

Performance Dependencies

Performance Dependencies

- Profile Words (Coverage):
  - Performance depends on quantity of overlap of words between a tweet and profile.
    - Multiple Sources: Training Set, Web Resources, Other sources.
  - Accuracy of the words-weights in a profile.

Performance Dependencies

- Profile Words (Coverage):
  - Performance depends on quantity of overlap of words between a tweet and profile.
    - Multiple Sources: Training Set, Web Resources, Other sources.
  - Accuracy of the words-weights in a profile.
- Word Weights:
  - Based on Training Set
  - Based on quality of the information source.

Basic Profile - 1

- **Homepage Source**:
  - Crawl the homepage until a depth d. Collect keywords. Stemming keywords, Removal of stop-words.
  - Challenges: Need to deal with variety of homepages. Flash-based, Javascript-based etc.
  - Good source for keywords related to the entity, but have to deal with quality of extraction.

Basic Profile - 1

- **Homepage Source**:
  - Crawl the homepage until a depth d. Collect keywords. Stemming keywords, Removal of stop-words.
  - Challenges: Need to deal with variety of homepages. Flash-based, Javascript-based etc.
  - Good source for keywords related to the entity, but have to deal with quality of extraction.

- **Meta-tags Source**:
  - Keywords directly specified in the meta-tags of the html page.
  - Very high quality. But only some percentage of homepages fill these tags.

Basic Profile - 1

- **Homepage Source**:
  - Crawl the homepage until a depth d. Collect keywords. Stemming keywords, Removal of stop-words.
  - Challenges: Need to deal with variety of homepages. Flash-based, Javascript-based etc.
  - Good source for keywords related to the entity, but have to deal with quality of extraction.

- **Meta-tags Source**:
  - Keywords directly specified in the meta-tags of the html page.
  - Very high quality. But only some percentage of homepages fill these tags.

- **Category Source**:
  - Category information of a company, along with wordnet we can identify the keywords which also represent the company.
  - Helps us associate "updates,install" etc. keywords to a software company.

Basic Profile - 2

- **GoogleSet or Common Knowledge Source**:
  - The Google Set keywords provide us with the competitor names, product names of a company.
  - Helps us associate “firefox,explorer,netscape ” keywords with “Opera Browser” Entity

Basic Profile - 2

- **GoogleSet or Common Knowledge Source**:
  - The Google Set keywords provide us with the competitor names, product names of a company.
  - Helps us associate "firefox,explorer,netscape " keywords with "Opera Browser" Entity

- **UserFeedback Positive Source**:
  - A user can list the keywords which he thinks are relevant to the company.
  - Very high quality. (But usually few in number)

Basic Profile - 2

- **GoogleSet or Common Knowledge Source**:
  - The Google Set keywords provide us with the competitor names, product names of a company.
  - Helps us associate "firefox,explorer,netscape " keywords with "Opera Browser" Entity
- **UserFeedback Positive Source**:
  - A user can list the keywords which he thinks are relevant to the company.
  - Very high quality. (But usually few in number)
- UserFeedback Negative Source:
  - Information about alternate entities which has same name as the current entity.
  - Wikipedia Disambiguation pages, User provides us with this set of keywords.

Profiles - Example : "Apple Inc."

HomePage Source : iphone, ipod, mac, safari, ios, iphoto, iwork, leopard, forum, items, employees, itunes, credit, portable, secure, unix, auditing, forums, marketers, browse, dominicana, music, recommend, preview, type, tell, notif, phone, purchase, manuals, updates, fifa, 8GB, 16GB, 32GB,...
Metadata Source : {empty}
Category Source : opera, code, brainchild, movie, telecom, cruncher, trade, cathode-ray, paper, freight, keyboard, dbm, merchandise, disk, language, microprocessor, move, web, monitor, diskett, show, figure, instrument, board, lade, digit, good, shipment, food, cpu, moving-picture, fluid, consign, contraband, electronic, volume, peripherals, crt, resolve, yield, server, micro, magazine, dreck, byproduct, spiritualist, telecommunications, manage, commodity, flick, vehicle, set, creation, procedure, consequence, second, design, result, mobile, home, processor, spin-off, wander, analog, transmission, cargo, expert, record, database, tube, payload, state, estimate, intersect, internet, print, factory, contrast, outcome, machine, deliver, effect, job, output, release, turnout, convert, river,...
GoogleSet Source : itunes, intel, belkin, 512mb, sony, hp, canon, powerpc, mac, apple, iphone, ati, microsoft, ibm,...
UserFeedback Positive Source : ipad, imac, iphone, ipod, itouch, itv, iad, itunes, keynote, safari, leopard, tiger, iwork, android, droid, phone, app, appstore, mac, macintosh
UserFeedback Negative Source : fruit, tree, eat, bite, juice, pineapple, strawberry, drink

Classification Process

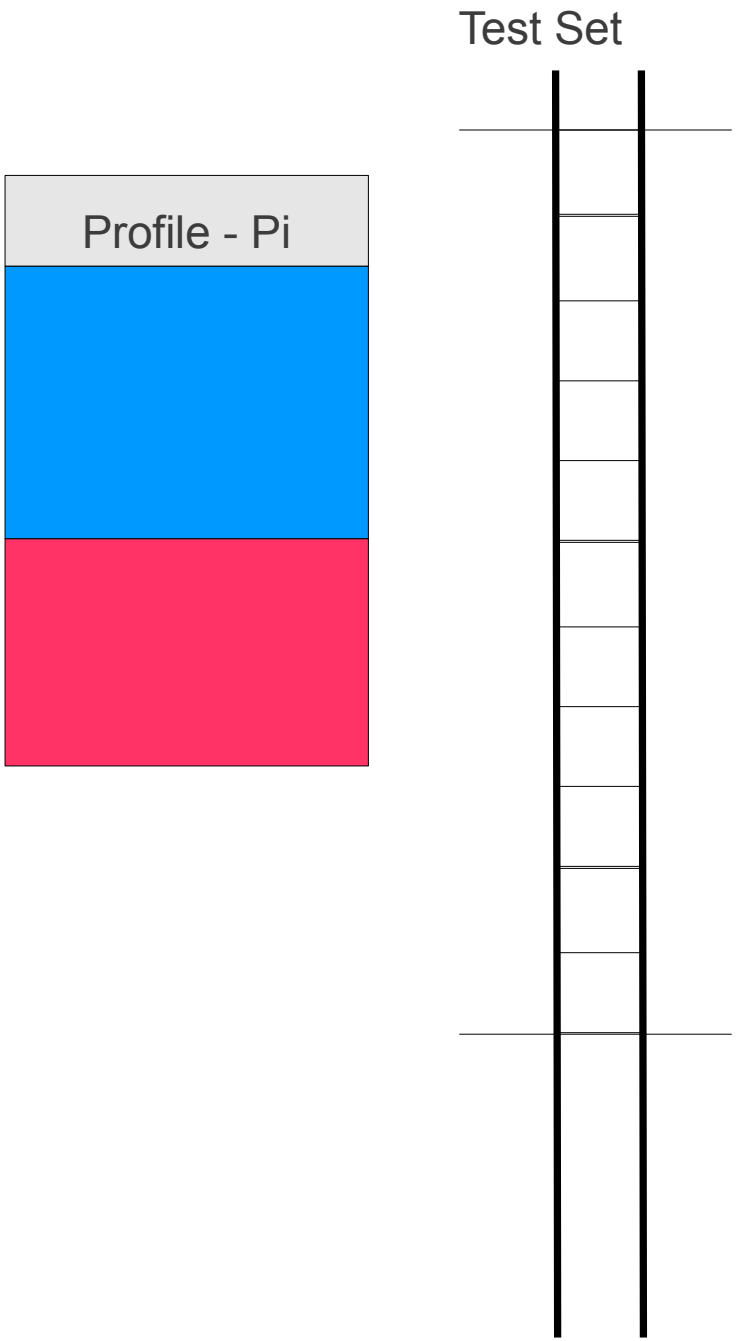
Compute the probabilities $P(C \mid T_i)$ (the tweet belongs to the Company) and $P(\overline{C} \mid T_i)$ (the tweet does not belong to the company)

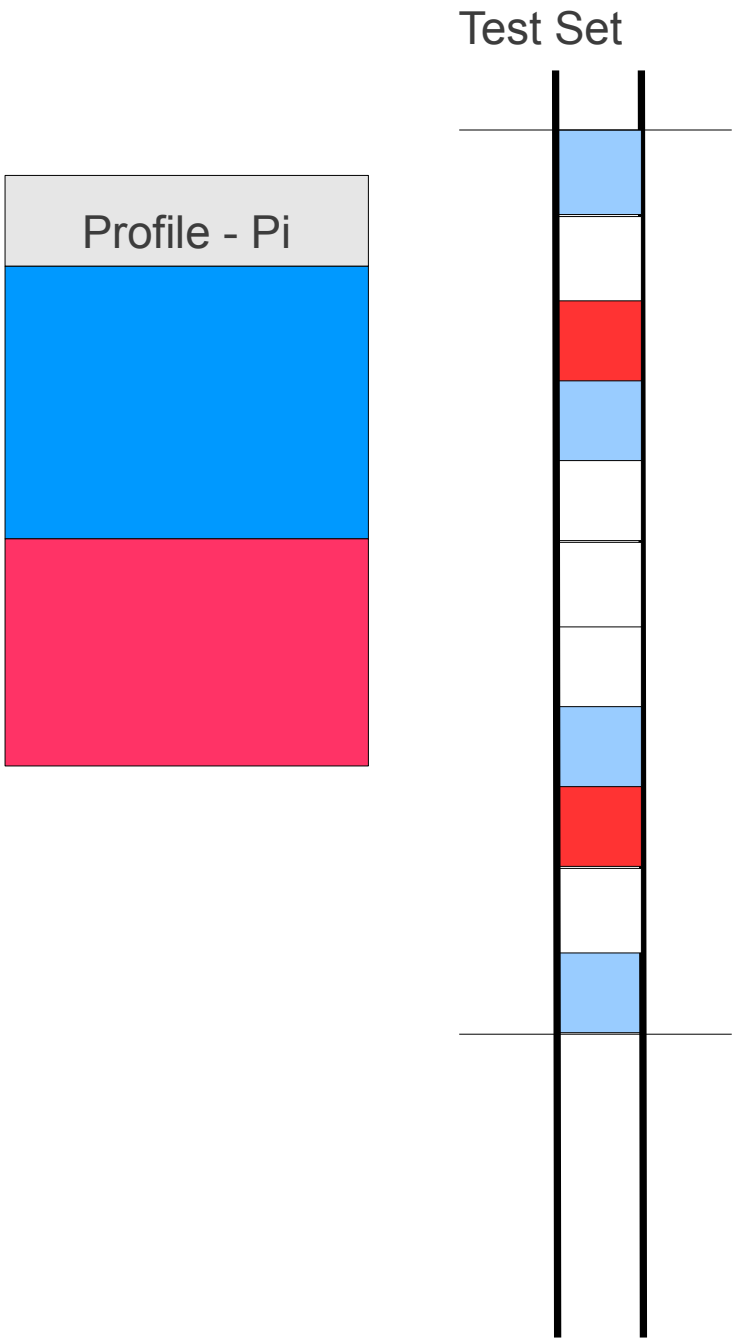
$$\begin{aligned} P(C \mid T_i) &= \frac{P(C) * P(T_i \mid C)}{P(T_i)} \\ &= \frac{P(C) * P(wrd_1^i, \ldots, wrd_n^i \mid C)}{P(T_i)} \\ &= K_1 \prod_{j=1}^{n} P(wrd_j^i \mid C) \end{aligned} \tag{1}$$

Similarly we have,

$$P(\overline{C} \mid T_i) = K_2 \prod_{j=1}^{n} P(wrd_j^i \mid \overline{C}) \tag{2}$$

Depending on which term of (1) and (2) is bigger, the tweet is decided as **belonging** or **not belonging** to the company.


Test Set
Profile - Pi


Test Set
Profile - Pi

Relatedness Factor

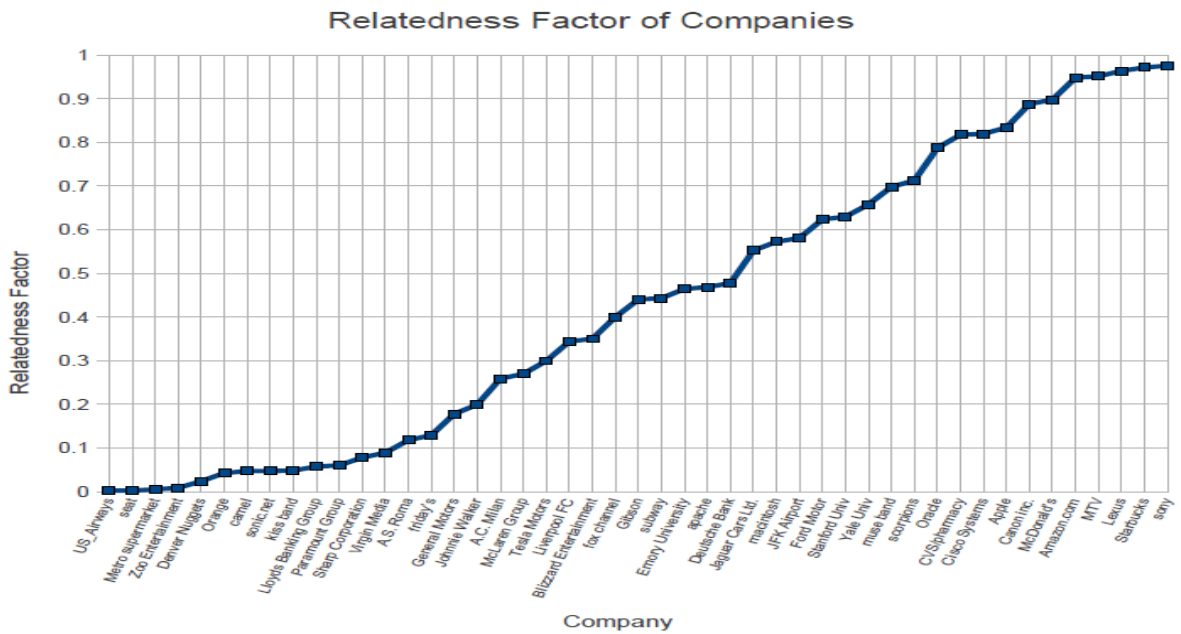
- Observations:
  - Many Tweets may have less overlap with the Basic-Profile of the company $\Rightarrow$ Uncertain Decision.

Relatedness Factor

- Observations:
  - Many Tweets may have less overlap with the Basic-Profile of the company $\Rightarrow$ Uncertain Decision.
  - All Company Names(query term) have different level of ambiguity (*relatedness factor*)

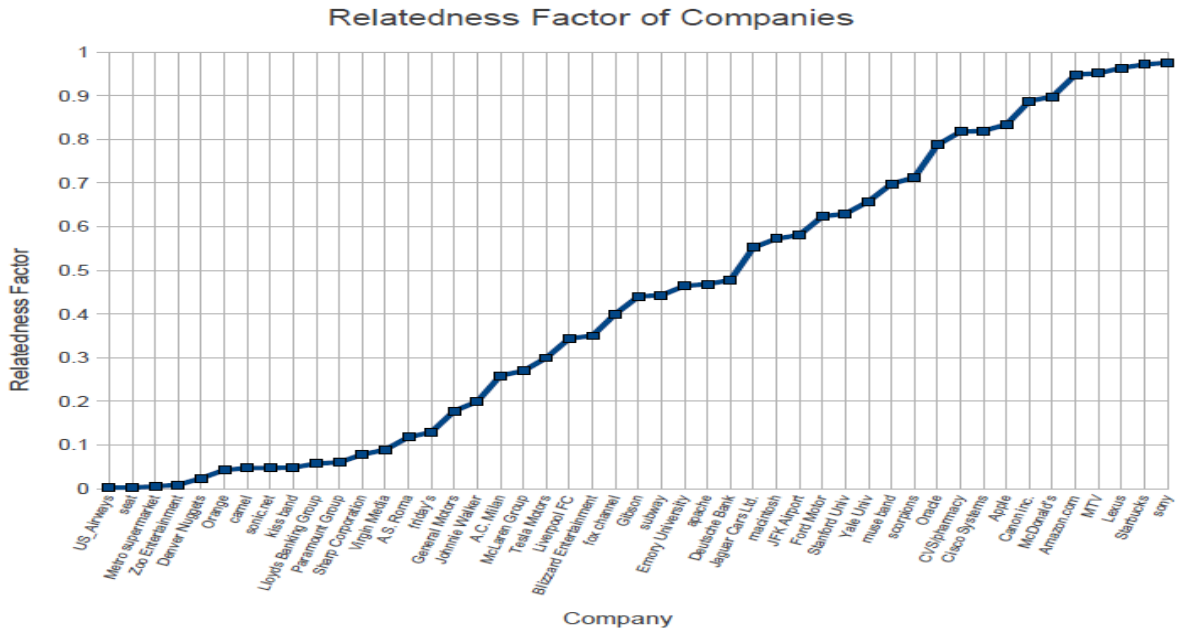
Relatedness Factor

- Observations:
  - Many Tweets may have less overlap with the Basic-Profile of the company $\Rightarrow$ Uncertain Decision.
  - All Company Names(query term) have different level of ambiguity (*relatedness factor*)



Relatedness Factor

- Observations:
  - Many Tweets may have less overlap with the Basic-Profile of the company $\Rightarrow$ Uncertain Decision.
  - All Company Names(query term) have different level of ambiguity (*relatedness factor*)



$$\textbf{Relatedness-Factor} = \frac{\#\ of\ \ tweets\ in\ Training\ Set \in Company}{\#\ of\ \ tweets\ in\ the\ Training\ Set}$$

Relatedness Factor based Classification

- Classification Process:
  - Default Decision:
    - If *relatedness-factor* ≥ 0.5 : Default decision : TRUE
    - Otherwise : Default decision : FALSE

  - ☺ Higher Accuracy. Expected Accuracy = *relatedness-factor*
  - ☹ Can not infer new words for adding to profile.

Relatedness Factor based Classification

- Classification Process:
  - Default Decision:
    - If *relatedness-factor* ≥ 0.5 : Default decision : TRUE
    - Otherwise : Default decision : FALSE

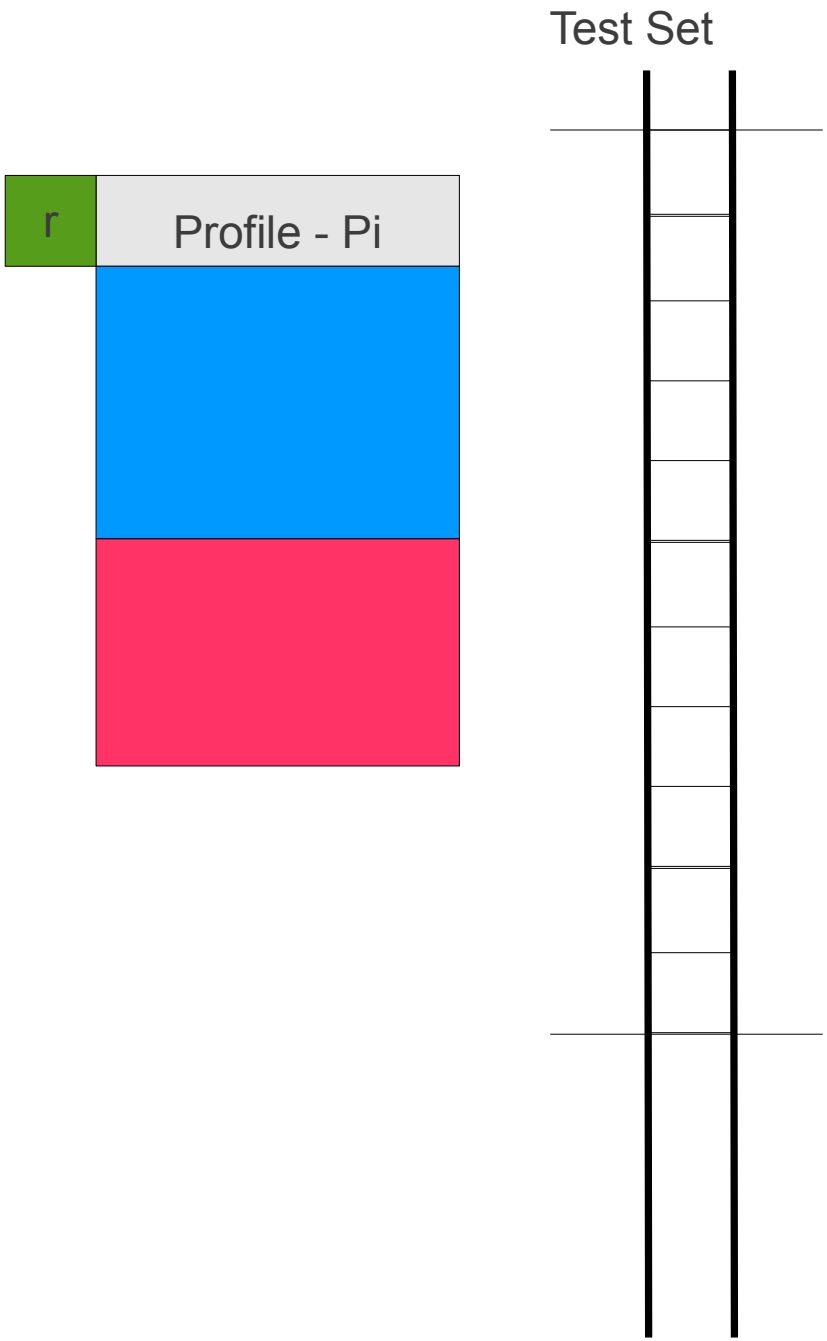
    ☺ Higher Accuracy. Expected Accuracy = *relatedness-factor*

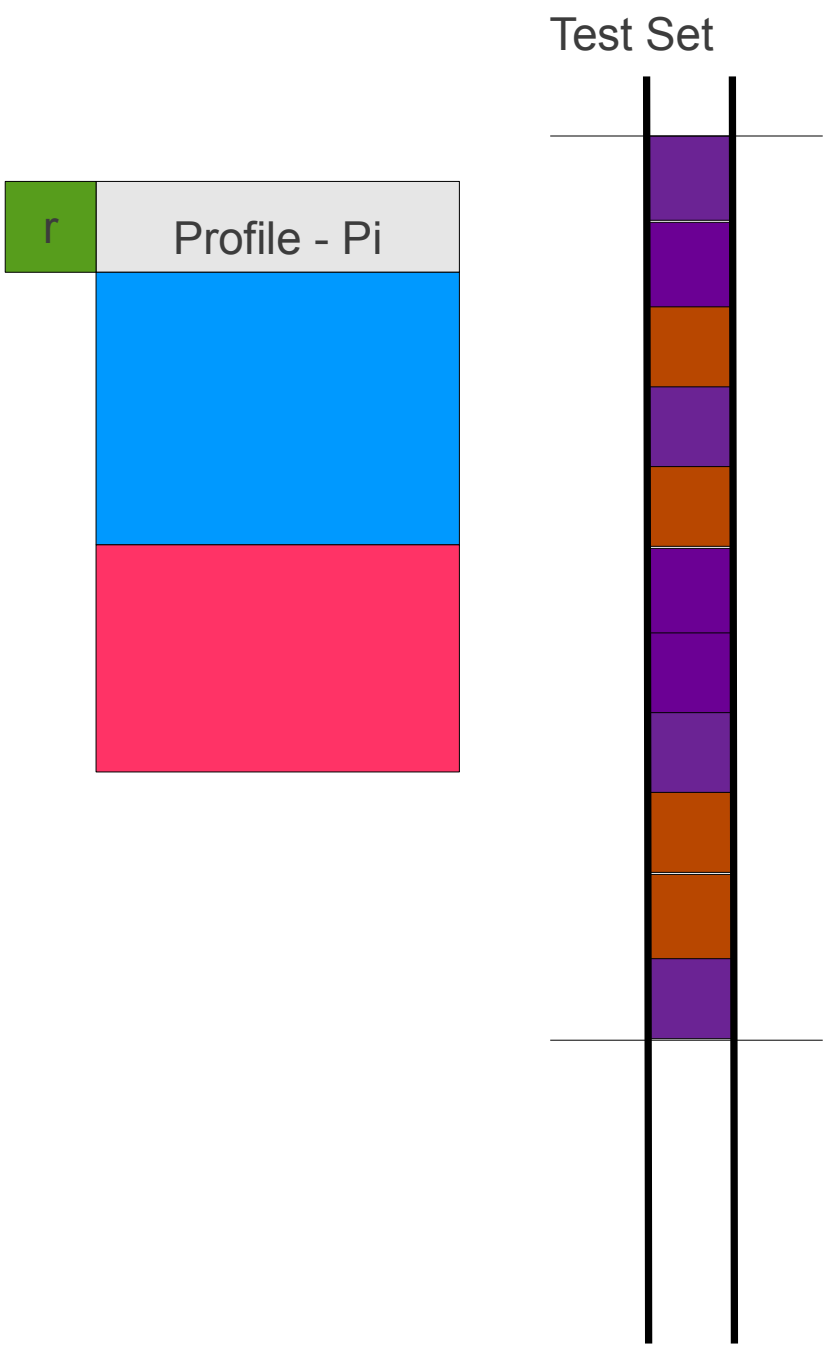
    ☹ Can not infer new words for adding to profile.

  - Random Decision:
    - p = UnifRand(0,1) $\leq$ *relatedness-factor(r)*: Decision : TRUE
    - Otherwise : Decision : FALSE

    ☹ Expected Accuracy $= r^2 + (1 - r)^2$

    ☺ Can infer new words for adding into profile, which should help in improving accuracy.


Test Set
r
Profile - Pi


Test Set
r
Profile - Pi

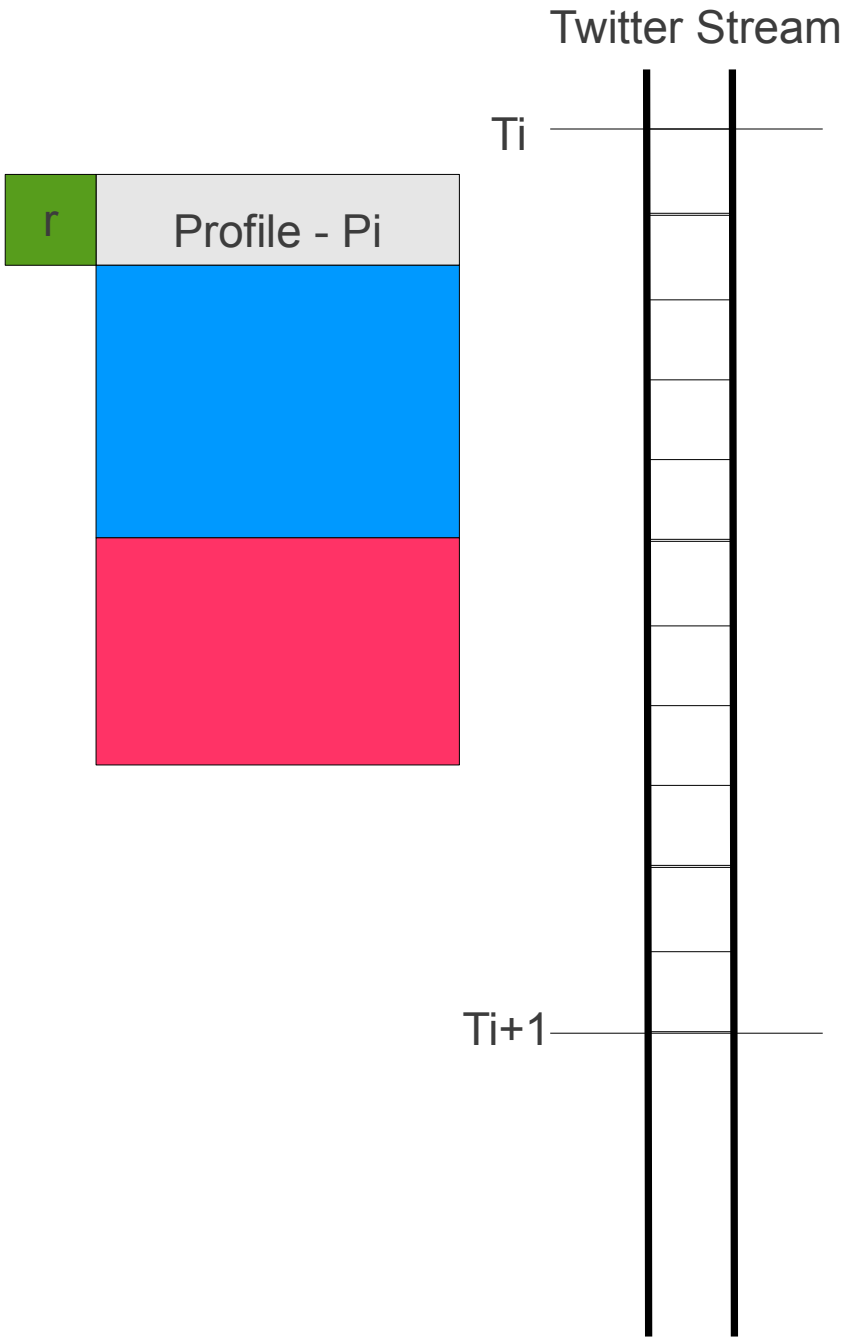
Active Stream based Classifier - 1

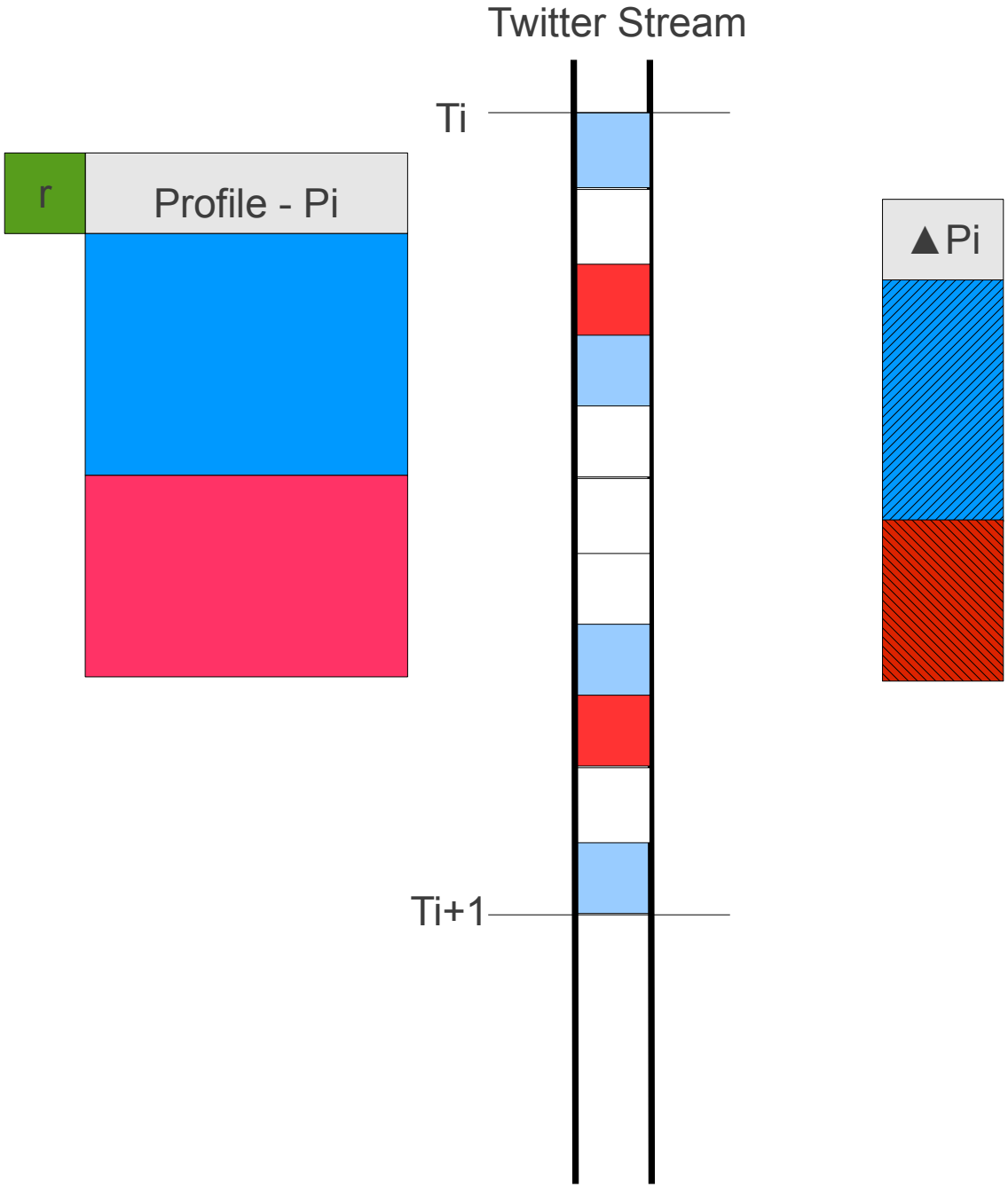
- Observations:
  - Profile contains limited set of words, limiting its overlap with tweets.
  - Impossible to have all words in the profile. Aim at-least for top-k keywords.
  - Power law in words.
  - Significant overlap in topK words in Test Set and words in Live Twitter Stream
  - Augment words into profile based on association.

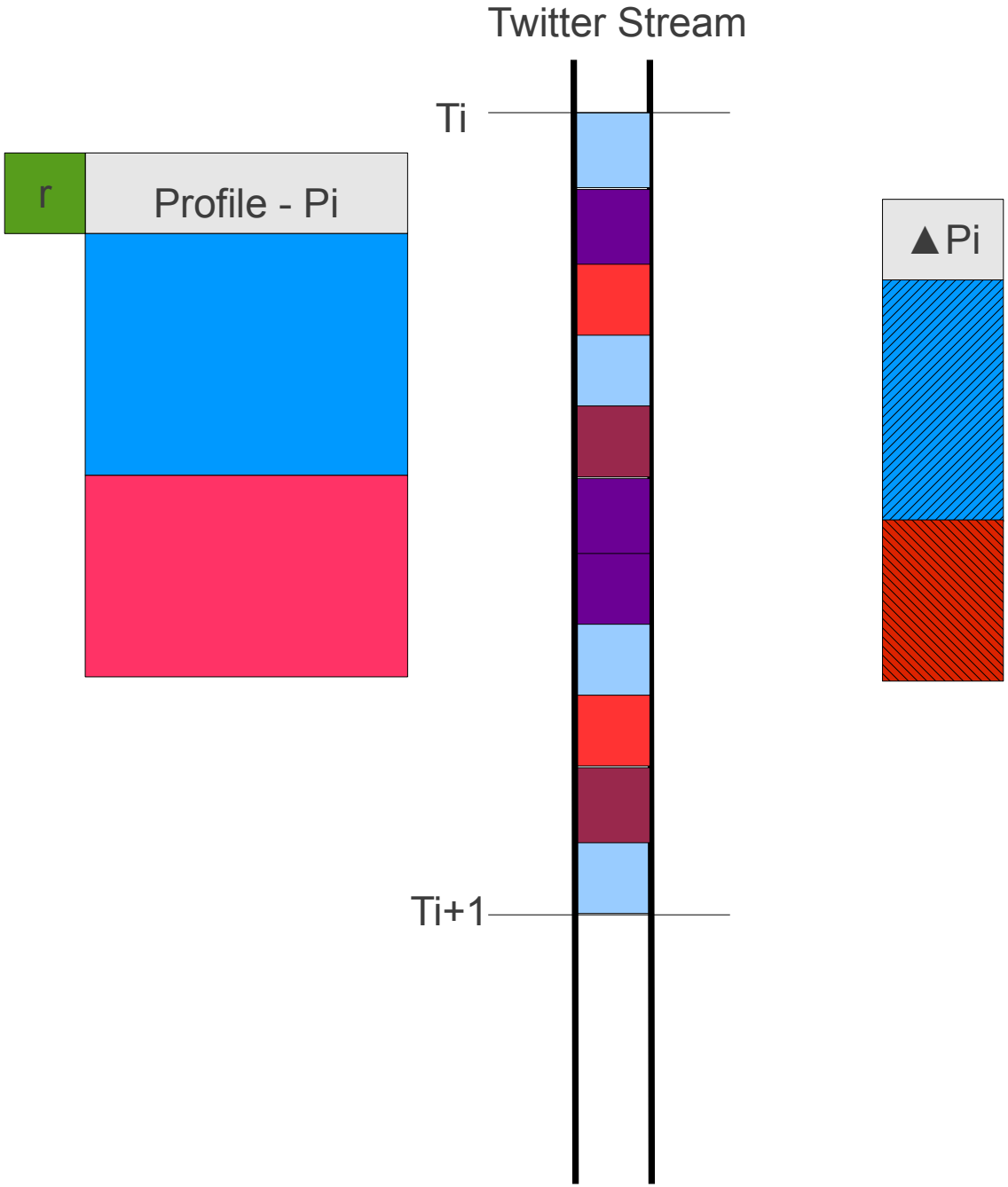
Active Stream based Classifier - 1

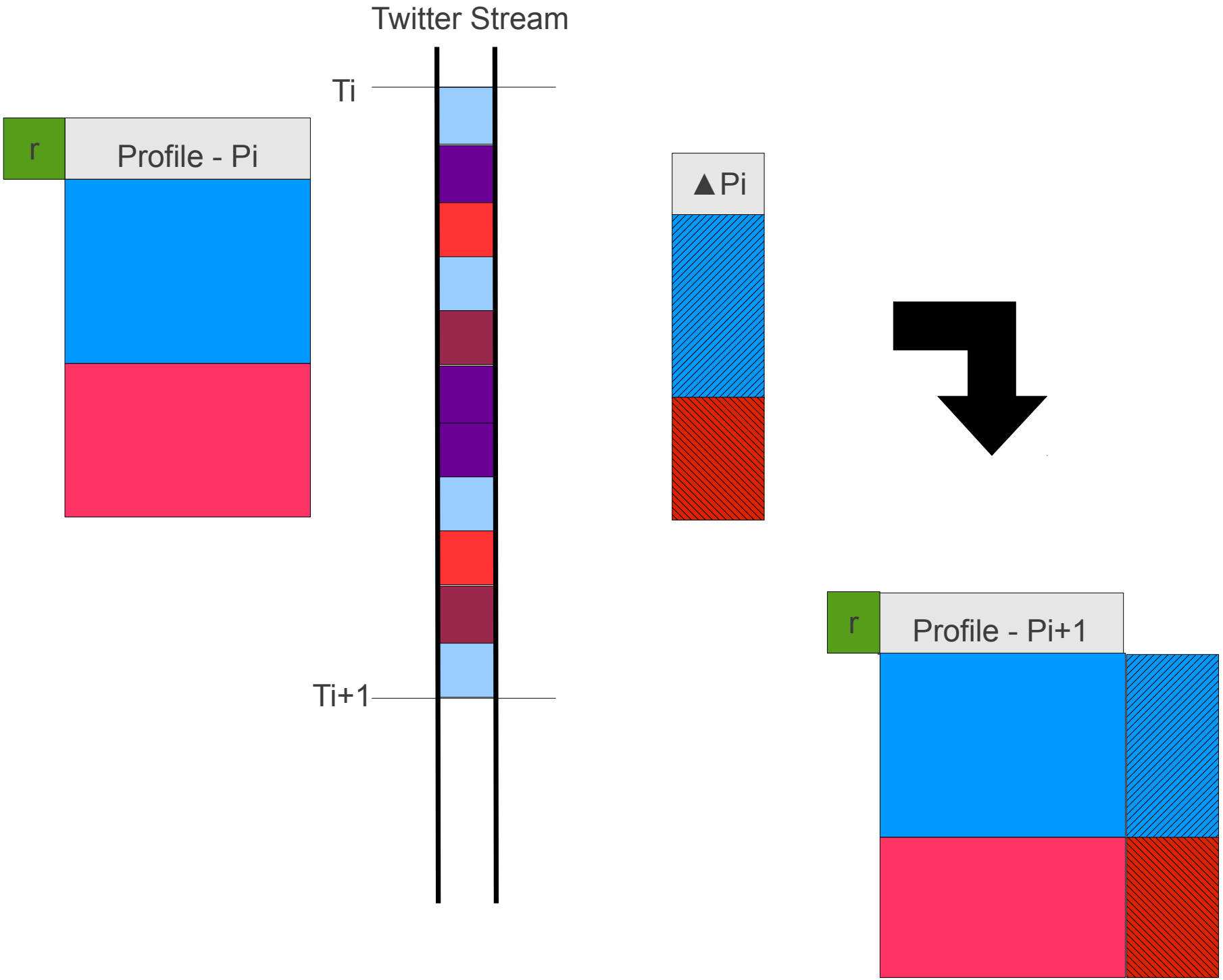
- Observations:
  - Profile contains limited set of words, limiting its overlap with tweets.
  - Impossible to have all words in the profile. Aim at-least for top-k keywords.
  - Power law in words.
  - Significant overlap in topK words in Test Set and words in Live Twitter Stream
  - Augment words into profile based on association.

- Quality Control:
  - Keep track of frequency of the new words one observes.
  - The weights of the newly identified words should be proportional to the quality of the words, that made the new words as possible candidates, and on the frequency of the word occurrence.


Twitter Stream
Ti
r
Profile - Pi
Ti+1


Twitter Stream
Ti
r
Profile - Pi
▲Pi
Ti+1


Twitter Stream
Ti
Ti+1
r
Profile - Pi
▲Pi


Twitter Stream
Ti
Ti+1
r
Profile - Pi
▲Pi
r
Profile - Pi+1

Active Stream based Classifier - 2

Input : Basic Profile: $P_0.Set^+$, $P_0.Set^-$
Twitter Stream: $\Gamma = \{T_1, \ldots, T_n\}$
R : *Relatedness* factor of company

Init : Active Tweet Sets: $P_\triangle.Set^+ = \{\}$, $P_\triangle.Set^- = \{\}$
for all $T_i \in \Gamma$ **do**
    $score$ = SCORE(T_i, $P_0.Set^+$)+SCORE(T_i, $P_0.Set^-$)
    if $score > 0$ **then**
        $P_\triangle.Set^+$.add(T_i,score)
    else if $score < 0$ **then**
        $P_\triangle.Set^-$.add(T_i,score)
    else
        if $Math.radom(0, 1) <$ *Relatedness* factor **then**
            $P_\triangle.Set^+$.add(T_i,*Relatedness*)
        else
            $P_\triangle.Set^-$.add(T_i,*Relatedness*)
        end if
    end if
end for
{ $P_\triangle.Set^+$,$P_\triangle.Set^-$ } = WordFreqAnalysis($P_\triangle.Set^+$, $P_\triangle.Set^-$)
Add Top-K keywords or all words above Threshold from $P_\triangle.Set^+$ to $P_0.Set^+$
Add Top-K keywords or all words above Threshold from $P_\triangle.Set^-$ to $P_0.Set^-$
return $P_0.Set^+$, $P_0.Set^-$

Experiments - Setup

Dataset

- WePS - 3 Dataset (available at http://nlp.uned.es/weps/weps-3/data)
- 50 Companies, about 500 Tweets per company.

Experiments - Setup

Dataset

- WePS - 3 Dataset (available at http://nlp.uned.es/weps/weps-3/data)
- 50 Companies, about 500 Tweets per company.

Metric:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

Experiments - Setup

Dataset

- WePS - 3 Dataset (available at http://nlp.uned.es/weps/weps-3/data)
- 50 Companies, about 500 Tweets per company.

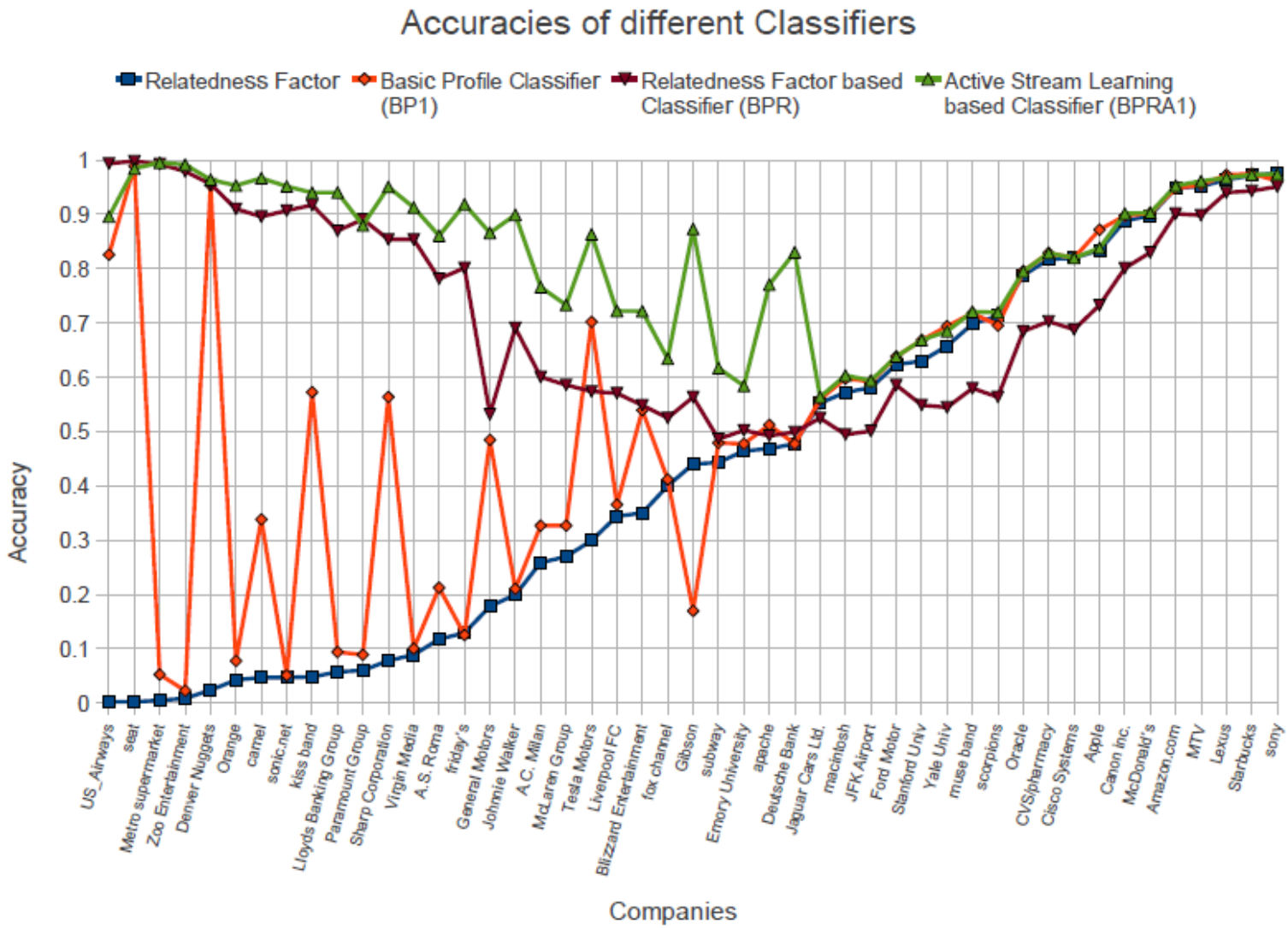
Metric:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

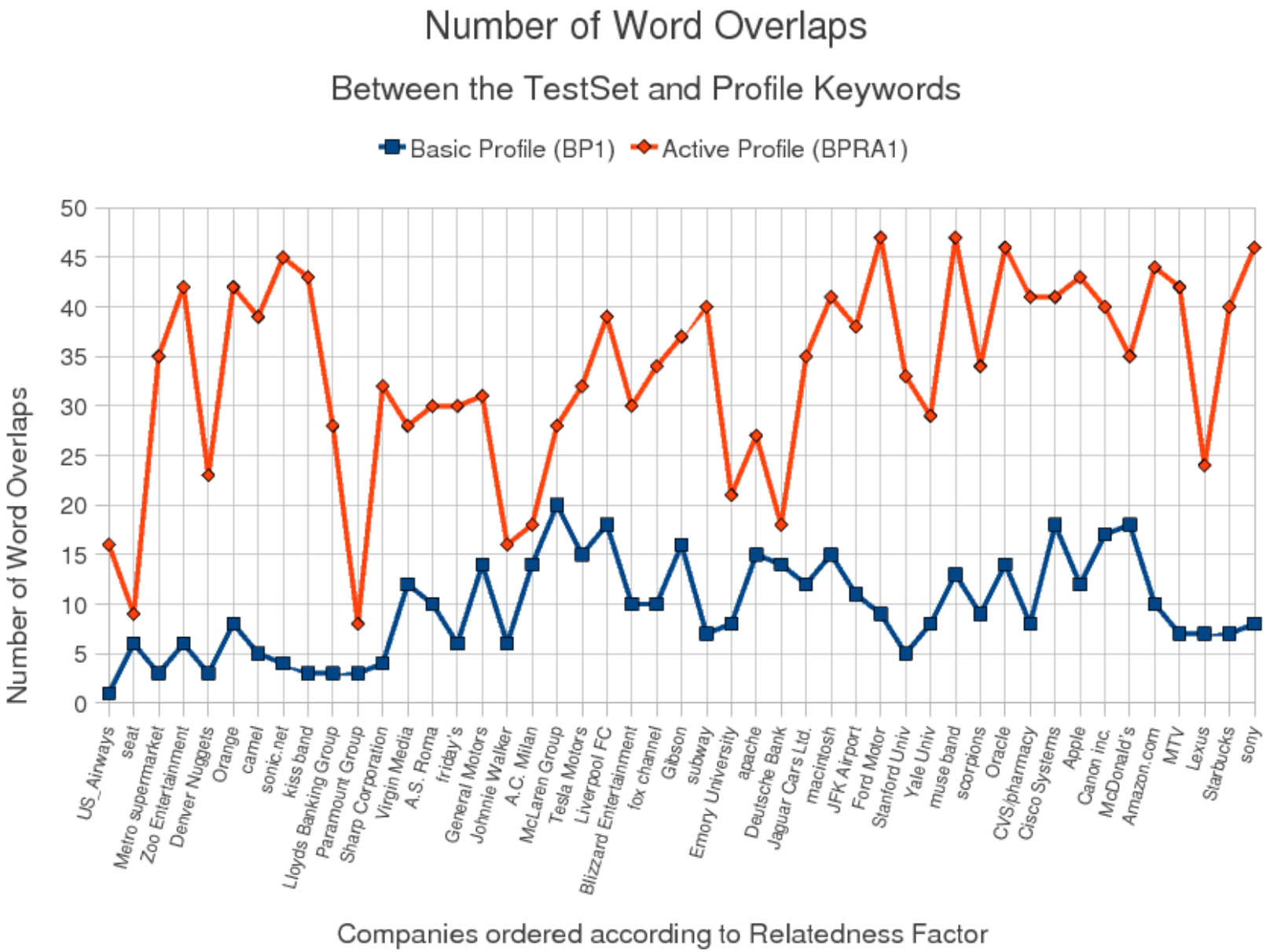
Experiments - I

- Comparison of classification accuracy of different classifiers:
  - Basic Profile Based Classifier (BP)
  - Relatedness Factor based Classifier (R)
  - Active Stream based Classifier (BP-R-A)

Performance of Different Classifiers



Top-K words overlap



Experiment II: Impact of Starting Profile - I

Basic Profiles (BP-n)

- Basic Profile Classifier using all sources (BP-1)
- Basic Profile Classifier using high quality sources (BP-2)

Experiment II: Impact of Starting Profile - I

Basic Profiles (BP-n)

- Basic Profile Classifier using all sources (BP-1)
- Basic Profile Classifier using high quality sources (BP-2)

Relatedness Factor based Classifier (BPR)

Experiment II: Impact of Starting Profile - I

Basic Profiles (BP-n)

- Basic Profile Classifier using all sources (BP-1)
- Basic Profile Classifier using high quality sources (BP-2)

Relatedness Factor based Classifier (BPR)

Active Learning based Profiles (BP-R-An)

- Active Learning Classifier starting with empty basic profile (BP-R-A0)
- Active Learning Classifier starting with low quality BP-0 (BP-R-A1)
- Active Learning Classifier starting with high quality BP-1 (BP-R-A2)

Impact of Starting Profile - 2

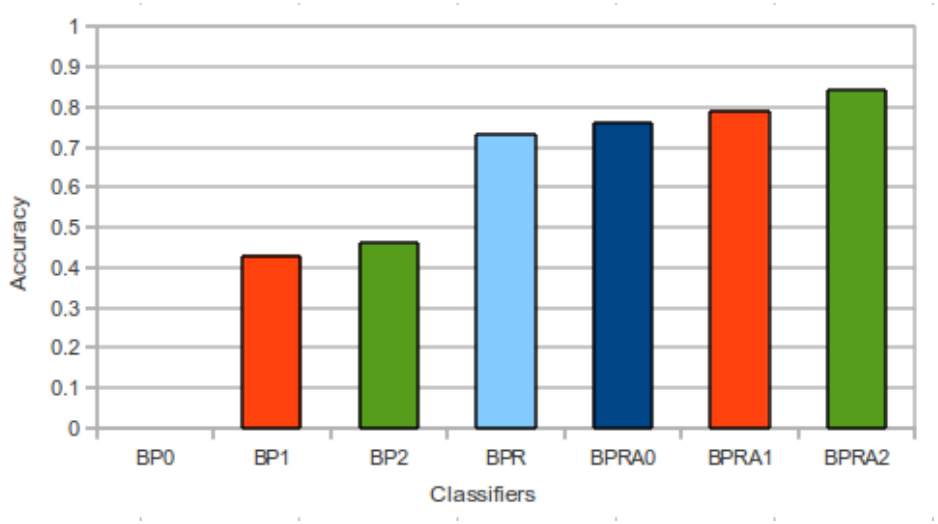


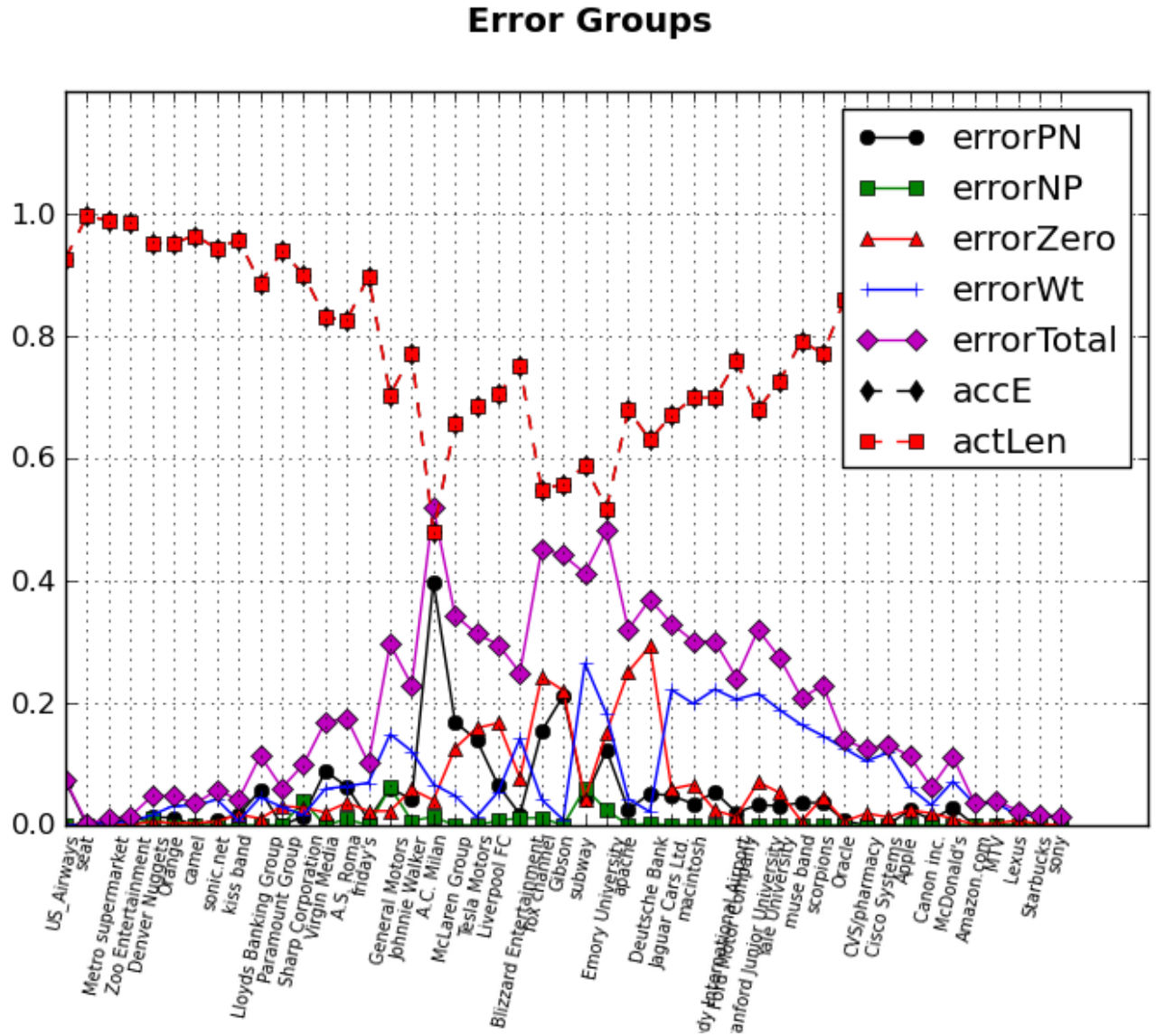
Table: Average Accuracy of Different Classifiers

Classifier	Average Accuracy
Basic Profile using all sources (BP1)	0.43
Basic Profile using only high quality sources (BP2)	0.46
Relatedness factor based classifier (BPR)	0.73
Active Profile constructed using the empty Basic Profile (BPRA0)	0.76
Active Profile constructed using normal quality Basic Profile-BP1 (BPRA1)	0.79
Active Profile constructed using high quality Basic Profile-BP2 (BPRA2)	0.84

Error Sources

- **errorZero** : Missing Words. When the profile does not contain the Tweet words.
- **errorPN** and **errorNP** : Positive evidence words wrongly put in negative profile and vice-versa.
- **errorWeight**: Wrong estimation of weight of a word.

Error Classes Distribution



Error Sources - Control

- **errorZero** : By inspecting the active streams for longer time windows.
- **errorPN**, **errorNP** and **errorWeight**: Adding only those words which have higher confidence. Tight trade-off between recall and accuracy.

Conclusions

- Classification of Tweet message w.r.t. a Company Entity.

Conclusions

- Classification of Tweet message w.r.t. a Company Entity.
- Techniques:
  - Basic Profile based Classification.
  - Relatedness Factor based Classification.
  - Active Learning based Classification.

Conclusions

- Classification of Tweet message w.r.t. a Company Entity.
- Techniques:
  - Basic Profile based Classification.
  - Relatedness Factor based Classification.
  - Active Learning based Classification.
- Future Work:
  - Error Analysis.
  - Trade-offs between Accuracy, Recall, and User Involvement.

Thank You !!
Questions/Comments/Discussion !!