# A Note on the Relationship between Different Types of Correction Queries

Cristina Tîrnăucă

Research Group on Mathematical Linguistics, Rovira i Virgili University
Pl. Imperial Tàrraco 1, Tarragona 43005, Spain
cristina.bibire@estudiants.urv.es

22nd of September, 2008

# Outline

1 Introduction

2 Learning with Correction Queries

3 Polynomial Time Learning with Correction Queries

# Motivation

- The adult-child interaction has been the inspiration for Angluin's query learning model [Angluin87], the forerunner of today's active learning field.
- Drawbacks:
  - EQs are both unrealistic and computationally costly
  - MQs are not informative enough (no feedback)
- Correction Queries (CQs) [BecYok04]

# Motivation

- The adult-child interaction has been the inspiration for Angluin's query learning model [Angluin87], the forerunner of today's active learning field.
- Drawbacks:
  - EQs are both unrealistic and computationally costly
  - MQs are not informative enough (no feedback)
- Correction Queries (CQs) [BecYok04]

# Motivation

- The adult-child interaction has been the inspiration for Angluin's query learning model [Angluin87], the forerunner of today's active learning field.
- Drawbacks:
  - EQs are both unrealistic and computationally costly
  - MQs are not informative enough (no feedback)
- Correction Queries (CQs) [BecYok04]

# Motivation

- The adult-child interaction has been the inspiration for Angluin's query learning model [Angluin87], the forerunner of today's active learning field.
- Drawbacks:
  - EQs are both unrealistic and computationally costly
  - MQs are not informative enough (no feedback)
- Correction Queries (CQs) [BecYok04]

# Motivation

- The adult-child interaction has been the inspiration for Angluin's query learning model [Angluin87], the forerunner of today's active learning field.
- Drawbacks:
  - EQs are both unrealistic and computationally costly
  - MQs are not informative enough (no feedback)
- Correction Queries (CQs) [BecYok04]

# Types of CQ

- Prefix Correction Queries (PCQs) [BeDeTi06]

$$C_L(u) = \begin{cases} \min\{v \mid uv \in L\}, & \text{if } u \in \mathit{Pref}(L) \\ \Theta, & \text{otherwise.} \end{cases}$$

- Length Bounded Correction Queries (LBCQs) [Tirn07]

$$C_L^l(u) = \{v \in \Sigma^* \mid uv \in L, |v| \leq l\}.$$

- Edit Distance Correction Queries (EDCQs) [BeHiJaTa07]

$$\textsc{Edc}_L(u) = \begin{cases} \text{Yes}, & \text{if } u \in L \\ v \in L \text{ s.t. } d(u,v) \text{ is minimum}, & \text{otherwise.} \end{cases}$$

# Types of CQ

- Prefix Correction Queries (PCQs) [BeDeTi06]

$$C_L(u) = \begin{cases} \min\{v \mid uv \in L\}, & \text{if } u \in \mathit{Pref}(L) \\ \Theta, & \text{otherwise.} \end{cases}$$

- Length Bounded Correction Queries (LBCQs) [Tirn07]

$$C_L^l(u) = \{v \in \Sigma^* \mid uv \in L, |v| \leq l\}.$$

- Edit Distance Correction Queries (EDCQs) [BeHiJaTa07]

$$\text{EDC}_L(u) = \begin{cases} \text{Yes}, & \text{if } u \in L \\ v \in L \text{ s.t. } d(u,v) \text{ is minimum}, & \text{otherwise.} \end{cases}$$

# Types of CQ

- Prefix Correction Queries (PCQs) [BeDeTi06]

$$C_L(u) = \begin{cases} \min\{v \mid uv \in L\}, & \text{if } u \in \mathit{Pref}(L) \\ \Theta, & \text{otherwise.} \end{cases}$$

- Length Bounded Correction Queries (LBCQs) [Tirn07]

$$C_L^l(u) = \{v \in \Sigma^* \mid uv \in L, |v| \leq l\}.$$

- Edit Distance Correction Queries (EDCQs) [BeHiJaTa07]

$$\textsc{Edc}_L(u) = \begin{cases} \text{Yes}, \\ v \in L \text{ s.t. } d(u,v) \text{ is minimum}, & \text{if } u \in L \\ & \text{otherwise.} \end{cases}$$

# Learning Models

- Concept: indexable class of recursive languages
- FinTxt, FinInf, ConsvTxt, LimTxt, LimInf
- MemQ, EquQ

# Learning Models

- Concept: indexable class of recursive languages
- FinTxt, FinInf, ConsvTxt, LimTxt, LimInf
- MemQ, EquQ

# Learning Models

- Concept: indexable class of recursive languages
- FinTxt, FinInf, ConsvTxt, LimTxt, LimInf
- MemQ, EquQ

# Learning Models

- Concept: indexable class of recursive languages
- FinTxt, FinInf, ConsvTxt, LimTxt, LimInf
- MemQ, EquQ

# Learning Models

- Concept: indexable class of recursive languages
- FinTxt, FinInf, ConsvTxt, LimTxt, LimInf
- MemQ, EquQ

# Learning Models

- Concept: indexable class of recursive languages
- FinTxt, FinInf, ConsvTxt, LimTxt, LimInf
- MemQ, EquQ

# Learning Models

- Concept: indexable class of recursive languages
- FinTxt, FinInf, ConsvTxt, LimTxt, LimInf
- MemQ, EquQ

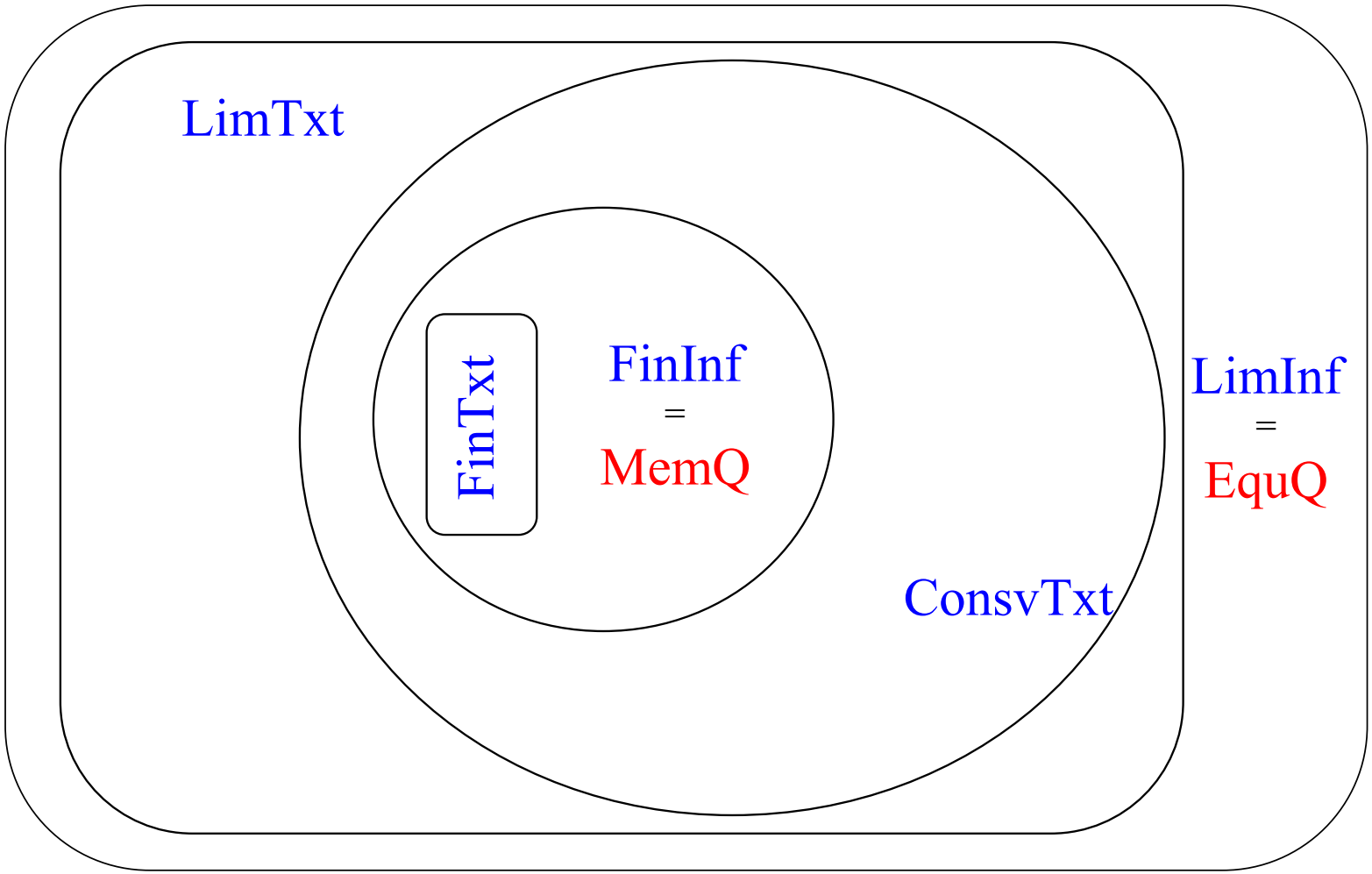
# Learning Models

- Concept: indexable class of recursive languages
- FinTxt, FinInf, ConsvTxt, LimTxt, LimInf
- MemQ, EquQ

# Learning Models

- Concept: indexable class of recursive languages
- FinTxt, FinInf, ConsvTxt, LimTxt, LimInf
- MemQ, EquQ

# Hierarchy [LanZil04]



# Outline

1 Introduction

2 Learning with Correction Queries

3 Polynomial Time Learning with Correction Queries

# Learning with PCQs [TirKob07]

**Definition (PCorQ)**

*all indexable classes $\mathcal{C}$ for which there is a query learner that identifies any language in $\mathcal{C}$ using a finite number of PCQs.*

- MemQ is strictly included in PCorQ,
- PCorQ and ConsvTxt are incomparable,
- PCorQ is strictly included in LimTxt.

# Learning with PCQs [TirKob07]

**Definition (PCorQ)**

*all indexable classes $\mathcal{C}$ for which there is a query learner that identifies any language in $\mathcal{C}$ using a finite number of PCQs.*

- MemQ is strictly included in PCorQ,
- PCorQ and ConsvTxt are incomparable,
- PCorQ is strictly included in LimTxt.

# Learning with PCQs [TirKob07]

**Definition (PCorQ)**

*all indexable classes $\mathcal{C}$ for which there is a query learner that identifies any language in $\mathcal{C}$ using a finite number of PCQs.*

- MemQ is strictly included in PCorQ,
- PCorQ and ConsvTxt are incomparable,
- PCorQ is strictly included in LimTxt.

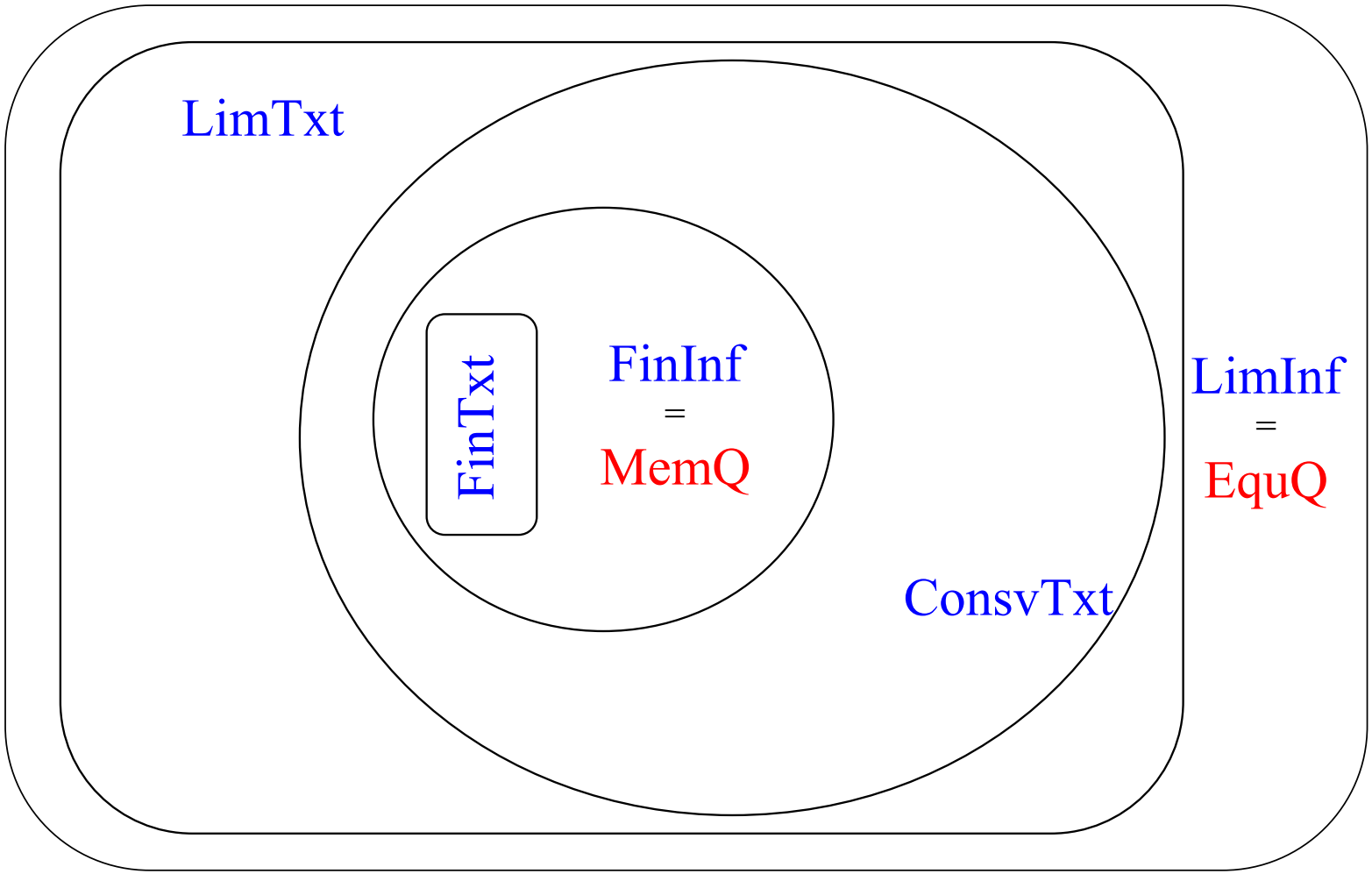
# Learning with PCQs [TirKob07]

**Definition (PCorQ)**

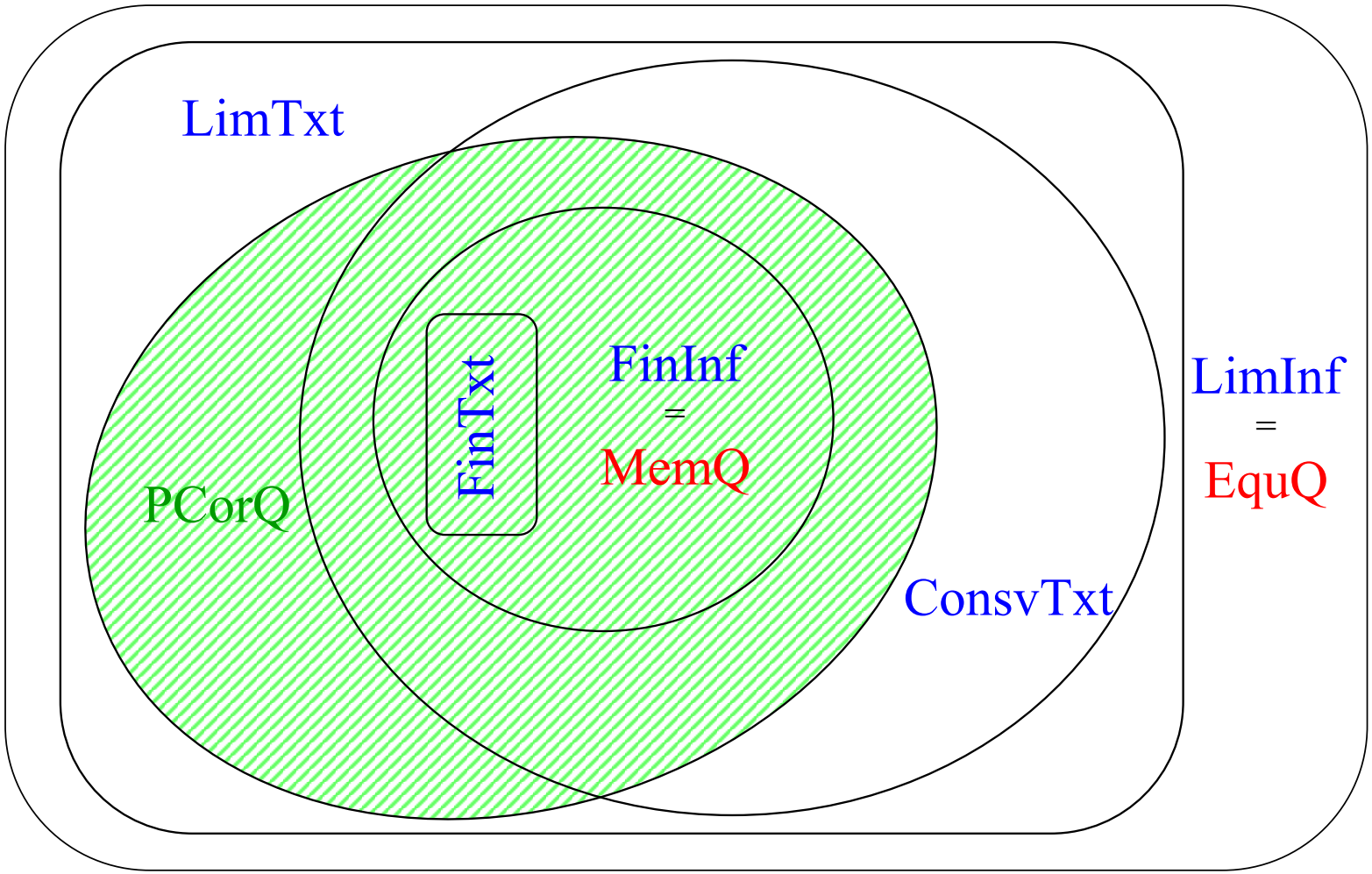
*all indexable classes $\mathcal{C}$ for which there is a query learner that identifies any language in $\mathcal{C}$ using a finite number of PCQs.*

- MemQ is strictly included in PCorQ,
- PCorQ and ConsvTxt are incomparable,
- PCorQ is strictly included in LimTxt.

# Hierarchy [LanZil04]



# Hierarchy [LanZil04, TirKob07]



# Learning with LBCQs

**Definition (IBCorQ)**

*all indexable classes $\mathcal{C}$ for which there is a query learner that identifies any language in $\mathcal{C}$ using a finite number of $l$-bounded CQs.*

**Theorem**

*For any $l \geq 0$, IBCorQ = MemQ.*

**Corollary**

*LBCorQ = MemQ.*

# Learning with LBCQs

**Definition (IBCorQ)**

*all indexable classes $\mathcal{C}$ for which there is a query learner that identifies any language in $\mathcal{C}$ using a finite number of $l$-bounded CQs.*

**Theorem**

*For any $l \geq 0$, $l$BCorQ = MemQ.*

**Corollary**

*LBCorQ = MemQ.*

# Learning with LBCQs

## Definition (IBCorQ)

*all indexable classes $\mathcal{C}$ for which there is a query learner that identifies any language in $\mathcal{C}$ using a finite number of $l$-bounded CQs.*

## Theorem

*For any $l \geq 0$, $l$BCorQ = MemQ.*

## Corollary

*LBCorQ = MemQ.*

# Learning with EDCQs

**Definition (EditCorQ)**

*all indexable classes $\mathcal{C}$ for which there is a query learner that identifies any language in $\mathcal{C}$ using a finite number of EDCQs.*

**Theorem**

*EditCorQ = MemQ.*

# Learning with EDCQs

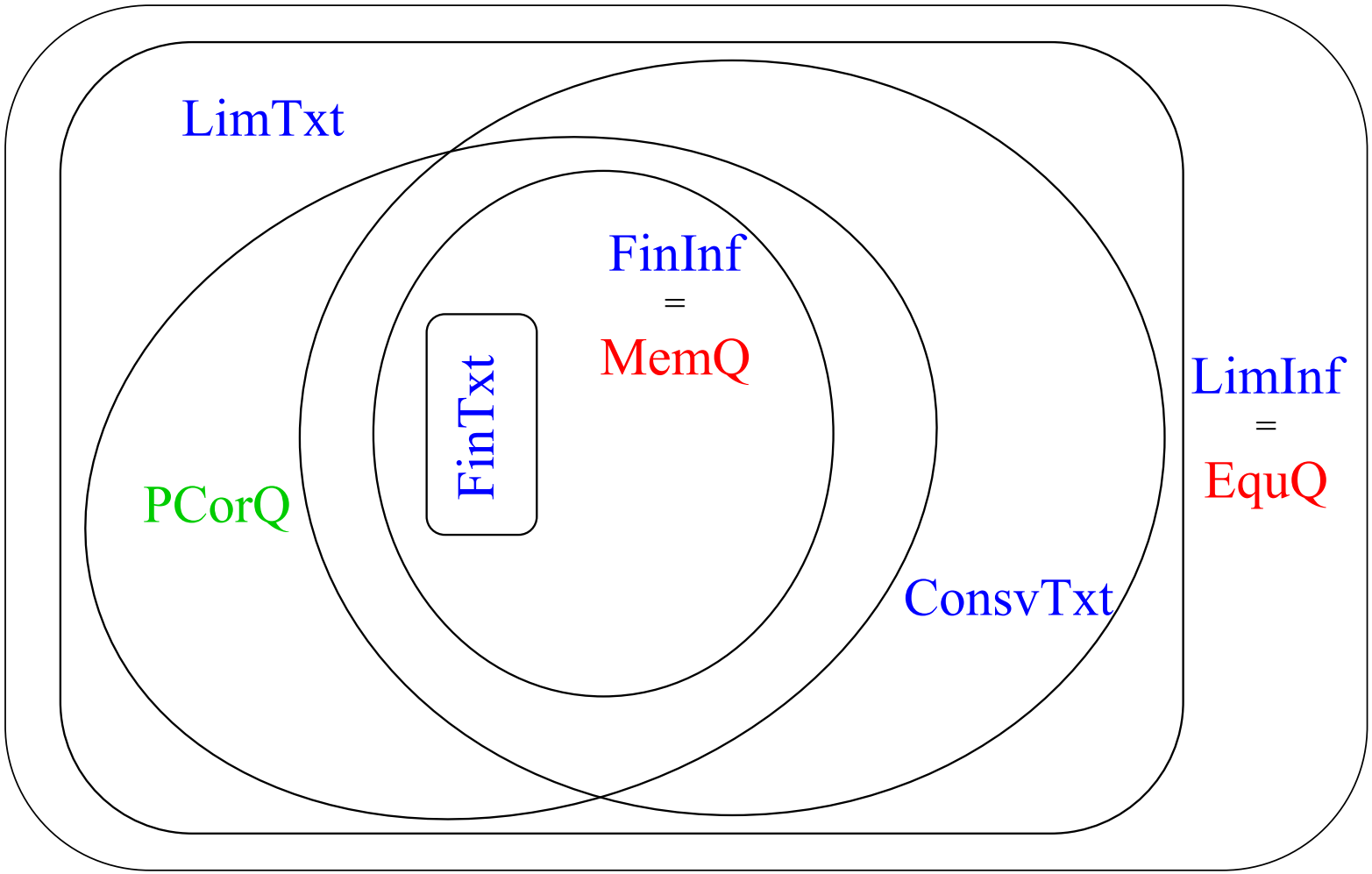
**Definition (EditCorQ)**

*all indexable classes $\mathcal{C}$ for which there is a query learner that identifies any language in $\mathcal{C}$ using a finite number of EDCQs.*

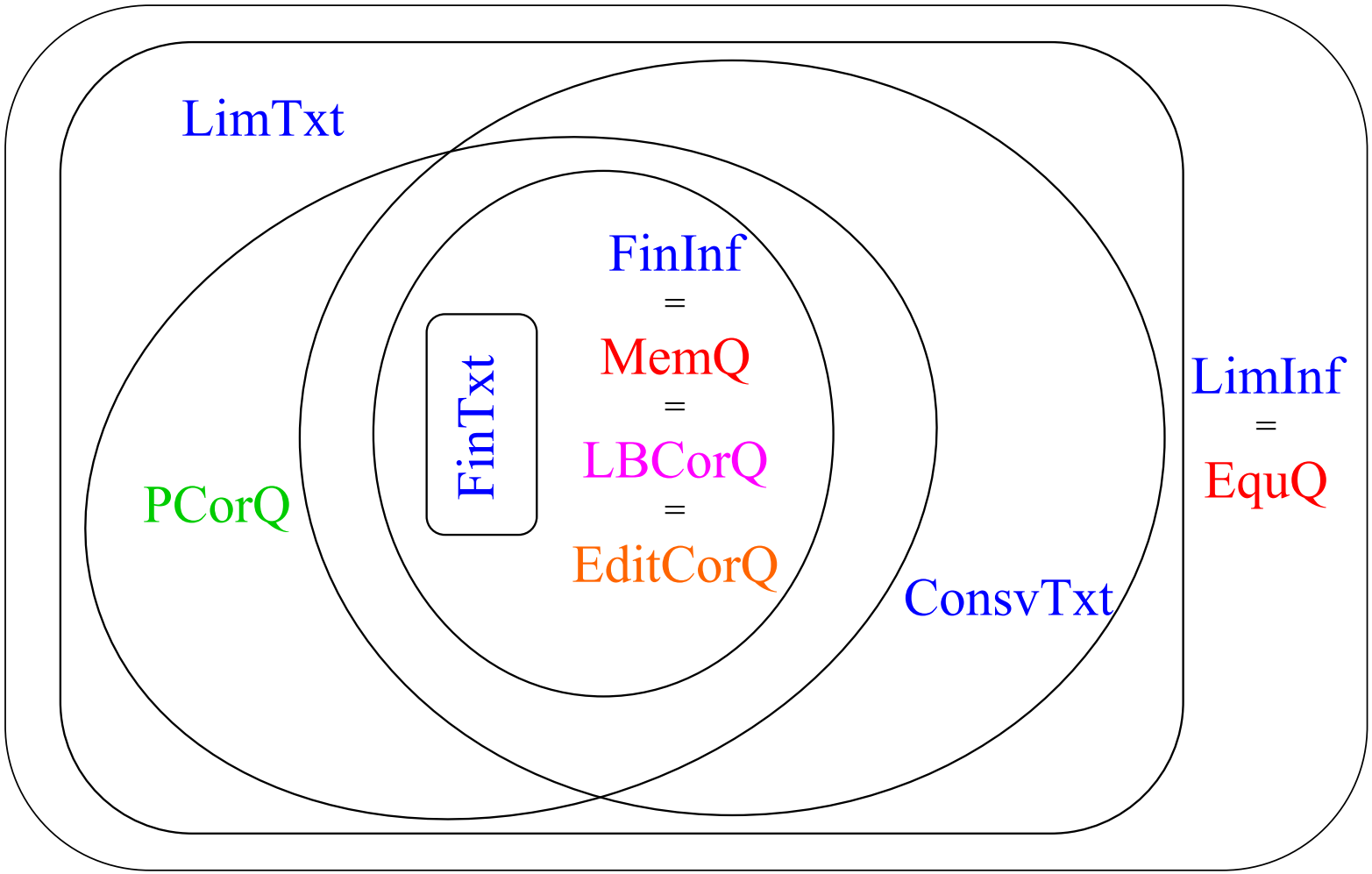
**Theorem**

*EditCorQ = MemQ.*

# The Global Picture



# The Global Picture



# Outline

1 Introduction

2 Learning with Correction Queries

3 Polynomial Time Learning with Correction Queries

# Pol-Time Learning with CQs

| Unrestricted | Polynomial Time |
|---|---|
| $MemQ \subsetneq PCorQ$ | $PolMemQ \subsetneq PolPCorQ$ [1] |
| $PCorQ \subsetneq EquQ$ | $PolPCorQ \not\subseteq PolEquQ$ [2] |
| $MemQ = LBCorQ$ | |
| $MemQ = EditCorQ$ | |

[1] see [TirKnu07]

[2] see [TirKnu07, Angluin90]

# Pol-Time Learning with CQs

| Unrestricted | Polynomial Time |
|---|---|
| $MemQ \subsetneq PCorQ$ | $PolMemQ \subsetneq PolPCorQ$ [1] |
| $PCorQ \subsetneq EquQ$ | $PolPCorQ \not\subseteq PolEquQ$ [2] |
| $MemQ = LBCorQ$ | |
| $MemQ = EditCorQ$ | |

[1] see [TirKnu07]

[2] see [TirKnu07, Angluin90]

# Pol-Time Learning with CQs

| Unrestricted | Polynomial Time |
|---|---|
| $MemQ \subsetneq PCorQ$ | $PolMemQ \subsetneq PolPCorQ$ [1] |
| $PCorQ \subsetneq EquQ$ | $PolPCorQ \not\subseteq PolEquQ$ [2] |
| $MemQ = LBCorQ$ | |
| $MemQ = EditCorQ$ | |

[1] see [TirKnu07]

[2] see [TirKnu07, Angluin90]

# Pol-Time Learning with CQs

| Unrestricted | Polynomial Time |
|---|---|
| $MemQ \subsetneq PCorQ$ | $PolMemQ \subsetneq PolPCorQ$ [1] |
| $PCorQ \subsetneq EquQ$ | $PolPCorQ \not\subseteq PolEquQ$ [2] |
| $MemQ = LBCorQ$ | |
| $MemQ = EditCorQ$ | |

[1] see [TirKnu07]

[2] see [TirKnu07, Angluin90]

# Pol-Time Learning with CQs

| Unrestricted | Polynomial Time |
|---|---|
| $MemQ \subsetneq PCorQ$ | $PolMemQ \subsetneq PolPCorQ$ [1] |
| $PCorQ \subsetneq EquQ$ | $PolPCorQ \not\subseteq PolEquQ$ [2] |
| $MemQ = LBCorQ$ | |
| $MemQ = EditCorQ$ | |

[1] see [TirKnu07]

[2] see [TirKnu07, Angluin90]

# Pol-Time Learning with CQs

| Unrestricted | Polynomial Time |
|---|---|
| $MemQ \subsetneq PCorQ$ | $PolMemQ \subsetneq PolPCorQ$ [1] |
| $PCorQ \subsetneq EquQ$ | $PolPCorQ \not\subseteq PolEquQ$ [2] |
| $MemQ = LBCorQ$ | |
| $MemQ = EditCorQ$ | |

[1] see [TirKnu07]

[2] see [TirKnu07, Angluin90]

# Pol-Time Learning with CQs

| Unrestricted | Polynomial Time |
|---|---|
| $MemQ \subsetneq PCorQ$ | $PolMemQ \subsetneq PolPCorQ$ [1] |
| $PCorQ \subsetneq EquQ$ | $PolPCorQ \not\subseteq PolEquQ$ [2] |
| $MemQ = LBCorQ$ | |
| $MemQ = EditCorQ$ | |

[1] see [TirKnu07]

[2] see [TirKnu07, Angluin90]

# Pol-Time Learning with CQs

| Unrestricted | Polynomial Time |
|---|---|
| $MemQ \subsetneq PCorQ$ | $PolMemQ \subsetneq PolPCorQ$ [1] |
| $PCorQ \subsetneq EquQ$ | $PolPCorQ \not\subseteq PolEquQ$ [2] |
| $MemQ = LBCorQ$ | |
| $MemQ = EditCorQ$ | |

[1] see [TirKnu07]

[2] see [TirKnu07, Angluin90]

# Pol-Time Learning with CQs

| Unrestricted | Polynomial Time |
|---|---|
| $MemQ \subsetneq PCorQ$ | $PolMemQ \subsetneq PolPCorQ$ [1] |
| $PCorQ \subsetneq EquQ$ | $PolPCorQ \not\subseteq PolEquQ$ [2] |
| $MemQ = LBCorQ$ | $PolMemQ \; ? \; PolLBCorQ$ |
| $MemQ = EditCorQ$ | |

[1] see [TirKnu07]

[2] see [TirKnu07, Angluin90]

# Pol-Time Learning with LBCQs

**Proposition**

*Pol(l-1)BCorQ* $=$ *PolIBCorQ for any l* $\geq 1$.

**Remark**

*Pol0BCorQ* $=$ *PolMemQ for any l* $\geq 1$.

**Corollary**

*PolLBCorQ* $=$ *PolMemQ for any l* $\geq 1$.

# Pol-Time Learning with LBCQs

**Proposition**

*Pol(l-1)BCorQ* $=$ *PollBCorQ for any l* $\geq 1$.

**Remark**

*Pol0BCorQ* $=$ *PolMemQ for any l* $\geq 1$.

**Corollary**

*PolLBCorQ* $=$ *PolMemQ for any l* $\geq 1$.

# Pol-Time Learning with LBCQs

**Proposition**

*Pol(l-1)BCorQ* $=$ *PollBCorQ for any l* $\geq 1$.

**Remark**

*Pol0BCorQ* $=$ *PolMemQ for any l* $\geq 1$.

**Corollary**

*PolLBCorQ* $=$ *PolMemQ for any l* $\geq 1$.

# Pol-Time Learning with CQs

| Unrestricted | Polynomial Time |
|---|---|
| $MemQ \subsetneq PCorQ$ | $PolMemQ \subsetneq PolPCorQ$ |
| $PCorQ \subsetneq EquQ$ | $PolPCorQ \not\subseteq PolEquQ$ |
| $MemQ = LBCorQ$ | $PolMemQ \; ? \; PolLBCorQ$ |
| $MemQ = EditCorQ$ | |

# Pol-Time Learning with CQs

| Unrestricted | Polynomial Time |
|---|---|
| $MemQ \subsetneq PCorQ$ | $PolMemQ \subsetneq PolPCorQ$ |
| $PCorQ \subsetneq EquQ$ | $PolPCorQ \not\subseteq PolEquQ$ |
| $MemQ = LBCorQ$ | $PolMemQ = PolLBCorQ$ |
| $MemQ = EditCorQ$ | |

# Pol-Time Learning with CQs

| Unrestricted | Polynomial Time |
|---|---|
| $MemQ \subsetneq PCorQ$ | $PolMemQ \subsetneq PolPCorQ$ |
| $PCorQ \subsetneq EquQ$ | $PolPCorQ \not\subseteq PolEquQ$ |
| $MemQ = LBCorQ$ | $PolMemQ = PolLBCorQ$ |
| $MemQ = EditCorQ$ | $PolMemQ \; ? \; PolEditCorQ$ |

# Pol-Time Learning with EDCQs

**Proposition**

*PolMemQ* $\subsetneq$ *PolEditCorQ.*

**Proof.**

Balls of strings [BeHiJaTa07]. □

# Pol-Time Learning with EDCQs

**Proposition**

*PolMemQ* $\subsetneq$ *PolEditCorQ.*

**Proof.**

Balls of strings [BeHiJaTa07]. □

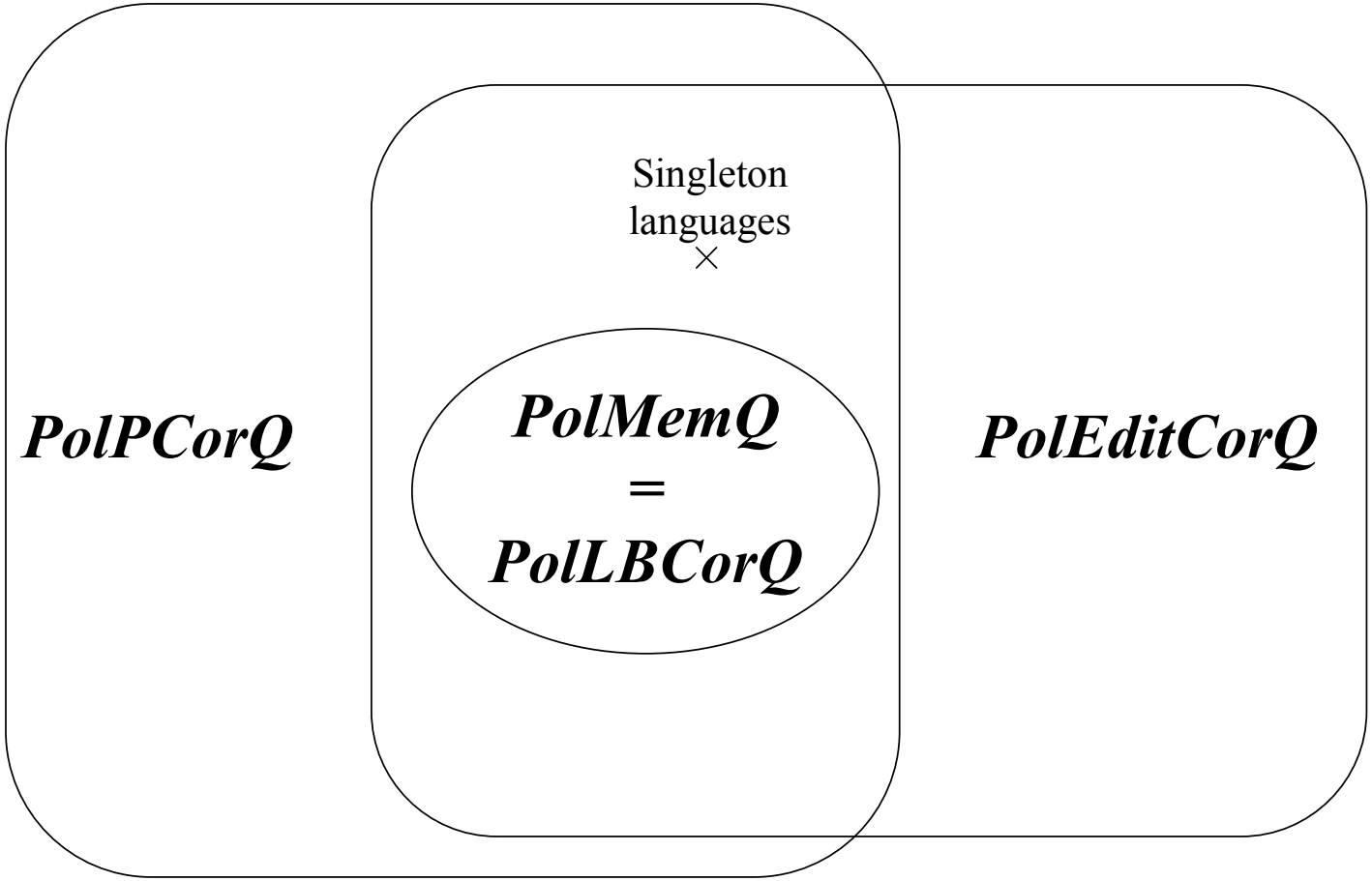
# Pol-Time Learning with CQs

| Unrestricted | Polynomial Time |
|---|---|
| $MemQ \subsetneq PCorQ$ | $PolMemQ \subsetneq PolPCorQ$ |
| $PCorQ \subsetneq EquQ$ | $PolPCorQ \not\subseteq PolEquQ$ |
| $MemQ = LBCorQ$ | $PolMemQ = PolLBCorQ$ |
| $MemQ = EditCorQ$ | $PolMemQ \; ? \; PolEditCorQ$ |

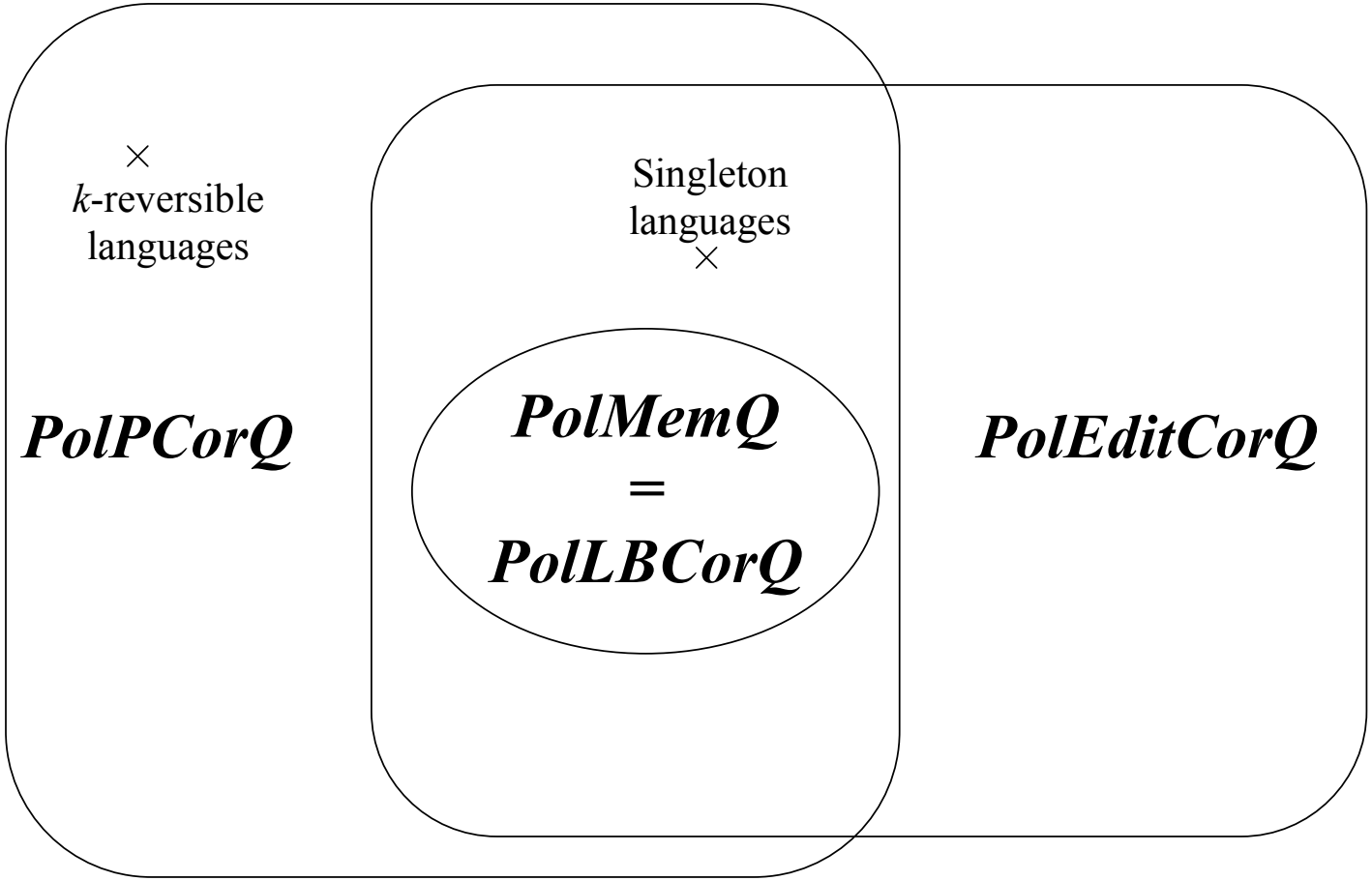
# Pol-Time Learning with CQs

| Unrestricted | Polynomial Time |
|---|---|
| $MemQ \subsetneq PCorQ$ | $PolMemQ \subsetneq PolPCorQ$ |
| $PCorQ \subsetneq EquQ$ | $PolPCorQ \not\subseteq PolEquQ$ |
| $MemQ = LBCorQ$ | $PolMemQ = PolLBCorQ$ |
| $MemQ = EditCorQ$ | $PolMemQ \subsetneq PolEditCorQ$ |

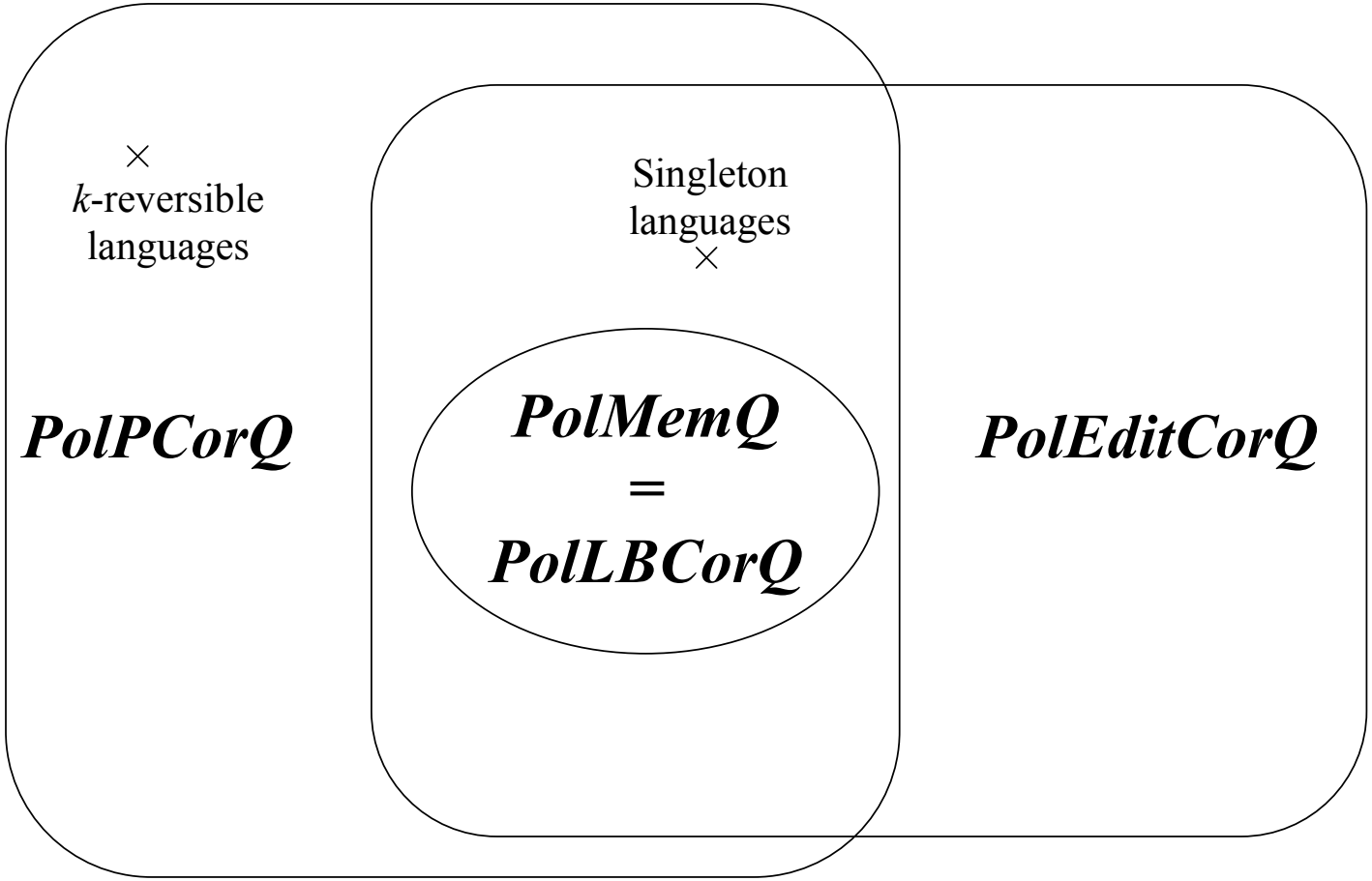
# The Global Picture



# The Global Picture

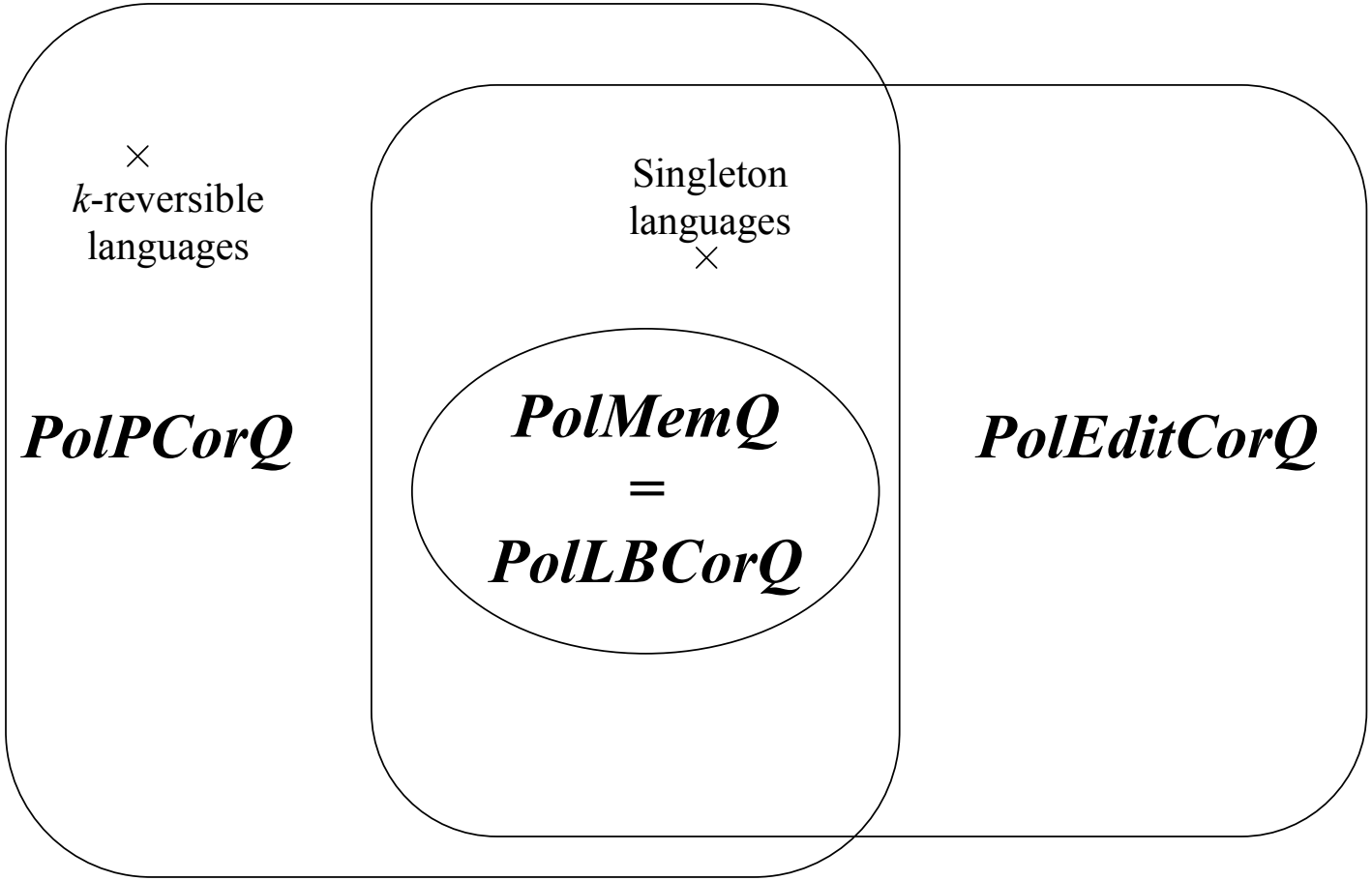


# The Global Picture



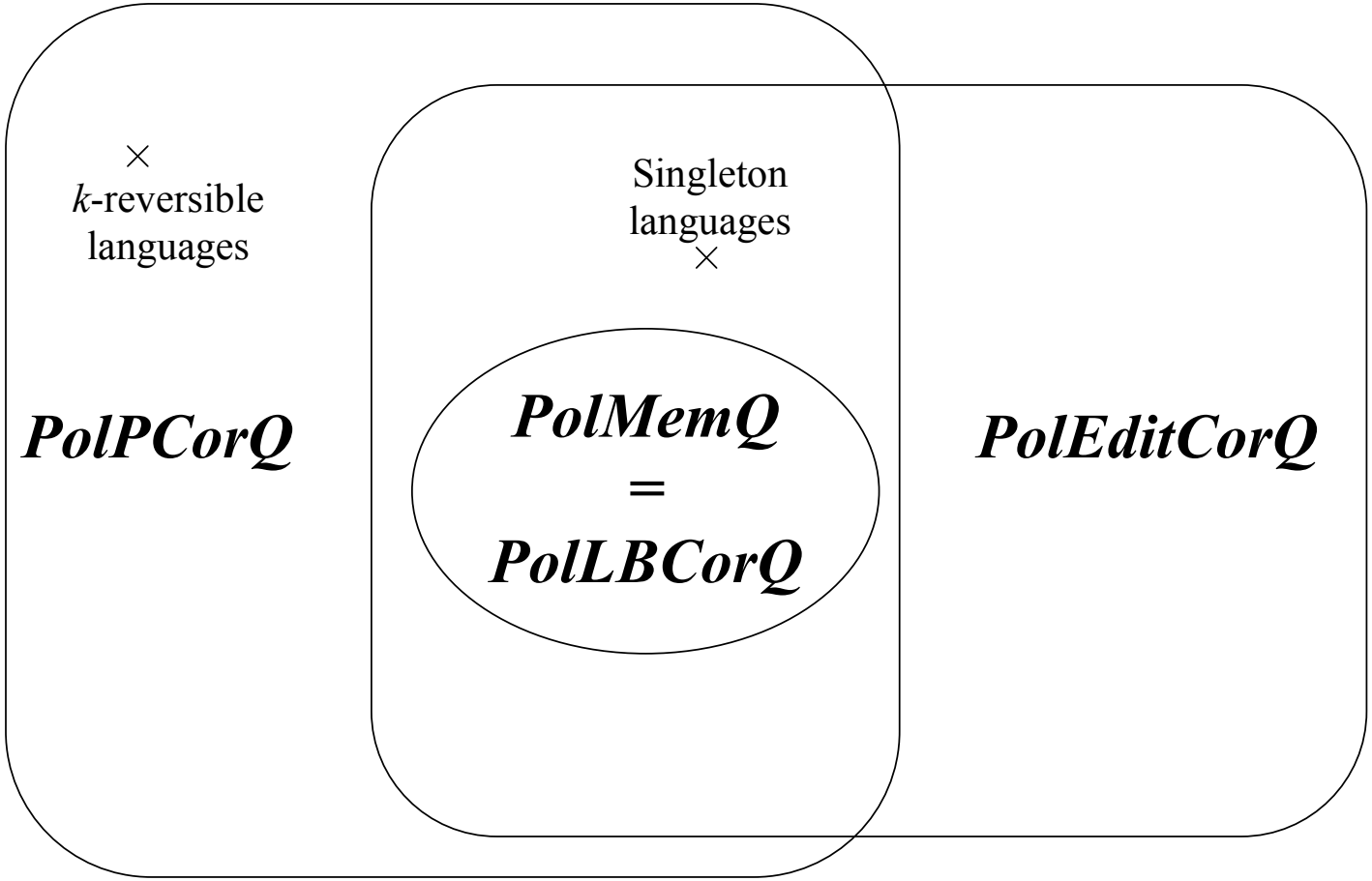

$k$-**Rev** $\notin$ *MemQ* [TirKnu07]

# The Global Picture



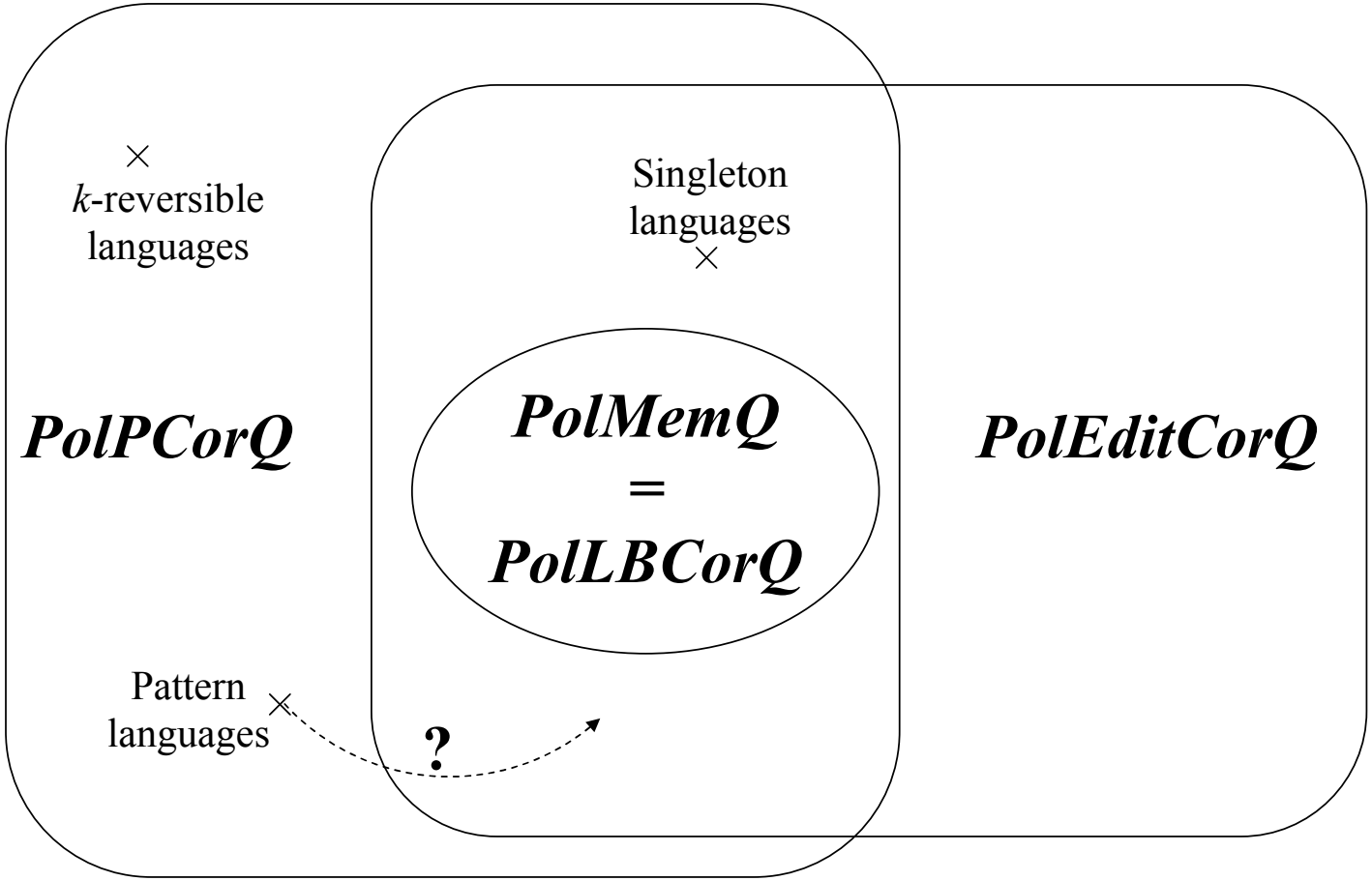

$$MemQ = EditCorQ$$

# The Global Picture

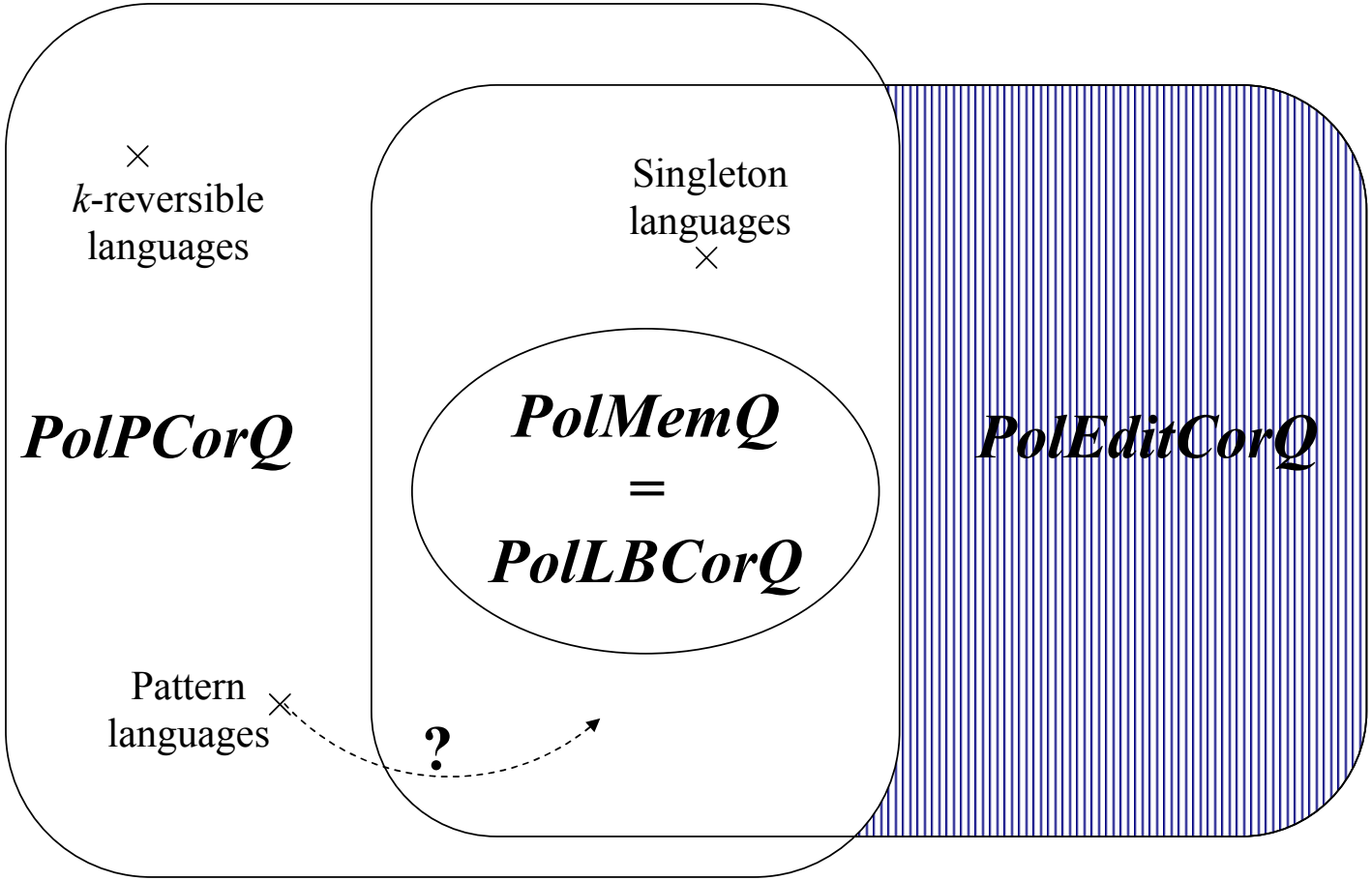



$k$-**Rev** $\notin$ *EditCorQ*

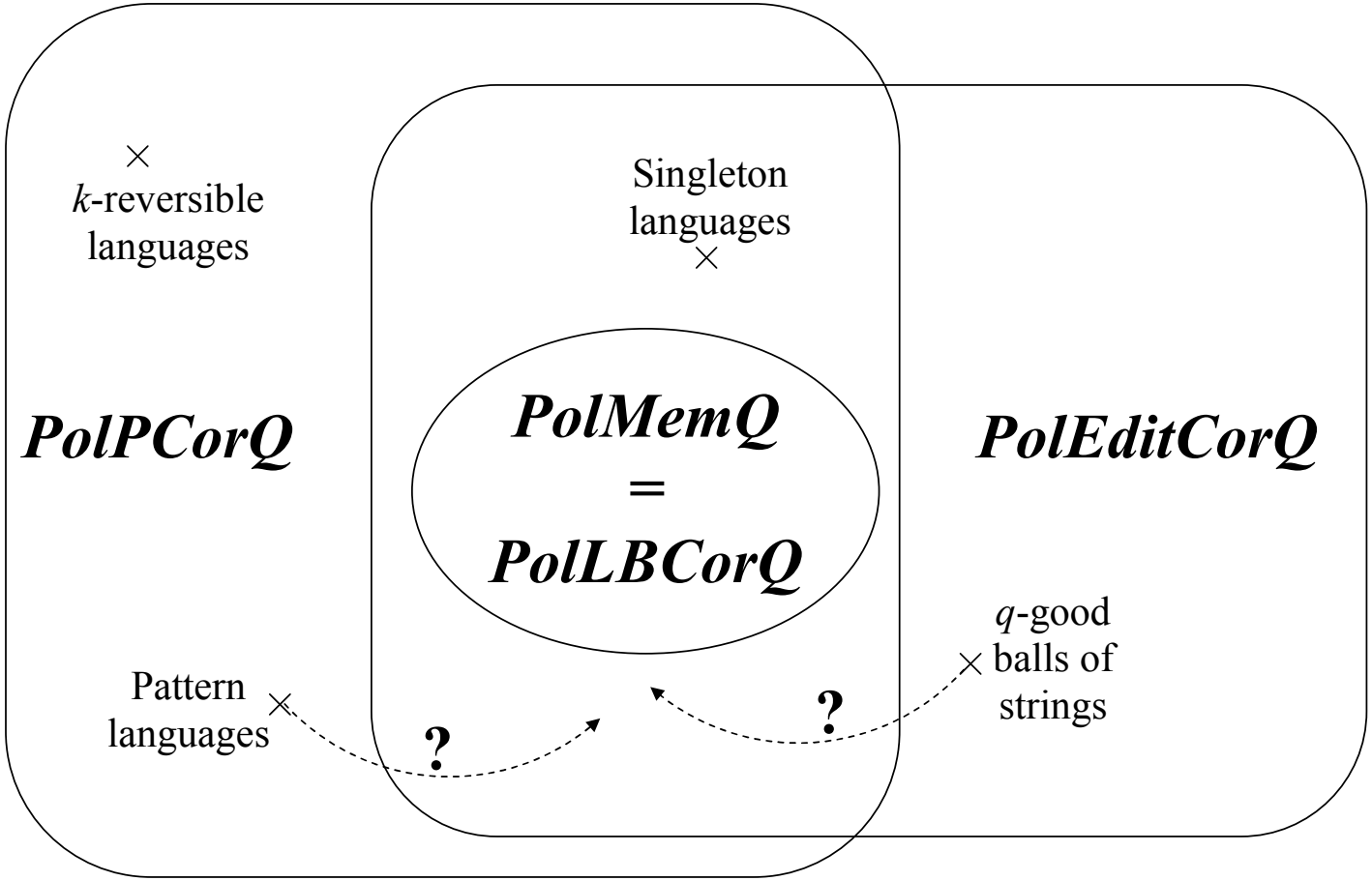
# The Global Picture



# The Global Picture



# The Global Picture



# References I


Dana Angluin:

Learning regular sets from queries and counterexamples.

Information and Computation **75**(2) (1987) 87–106

Dana Angluin:

Negative results for equivalence queries.

Machine Learning **5**(2) (1990) 121–150

Leonor Becerra-Bonache, Adrian Horia Dediu, Cristina Tîrnăucă:

Learning DFA from correction and equivalence queries.

Proc. of ICGI '06. Vol. 4201 of LNAI, Springer-Verlag (2006) 281–292

# References II


Leonor Becerra-Bonache, Takashi Yokomori:

Learning Mild Context-Sensitiveness: Toward Understanding Children's Language Learning

Proc. of ICGI '04. Vol. 3264 of LNAI, Springer-Verlag (2004) 53–64

Leonor Becerra-Bonache, Colin de la Higuera, Jean-Christophe Janodet, Frédéric Tantini:

Learning Balls of Strings with Correction Queries.

Proc. of ECML '07. Vol. 4701 of LNAI, Springer-Verlag (2007) 18–29

Steffen Lange, Sandra Zilles:

Formal language identification: query learning vs. Gold-style learning.

Information Processing Letters **91**(6) (2004) 285–292

# References III


Cristina Tîrnăucă:

Learning reversible languages from correction queries only.

http://grlmc-dfilrom.urv.cat/grlmc/PersonalPages/cris

Cristina Tîrnăucă, Satoshi Kobayashi:

A characterization of the language classes learnable with correction queries.

Proc. of TAMC '07. Vol. 4484 of LNCS, Springer-Verlag (2007) 398–407

Cristina Tîrnăucă, Timo Knuutila:

Polynomial time algorithms for learning $k$-reversible languages and pattern languages with correction queries.

Proc. of ALT '07. Vol. 4754 of LNCS, Springer-Verlag (2007) 272–284

Thank You!

Merci Beaucoup!

!Muchas Gracias!

Mulţumesc!